Visual Servoing of Fixed-Wing Unmanned Aerial Vehicle Using Command Filtered Backstepping

Fadjar Rahino Triputra[1,2], Bambang Riyanto Trilaksono[2], Trio Adiono[2],
and Rianto Adhy Sasongko[3]

[1]Center for Information and Communication Technology,
Agency for the Assessment and Application of Technology,
[2]School of Electrical Engineering and Informatics, Institut Teknologi Bandung,
[3]Faculty of Mechanical and Aerospace Engineering, Institut Teknologi Bandung,
INDONESIA
fadjar.rahino@bppt.go.id

Abstract: This paper presents a full dynamic visual servoing flight controller design using command filtered backstepping (CFBS) control law for a fixed-wing unmanned aerial vehicle (UAV). A full nonlinear dynamic model that involves feature point motion, pan-tilt gimbal mechanism movement, and UAV flight dynamic is adapted to the controller design of CFBS. The proposed design scheme can provide a system which is easy to implement in various fixed-wing UAV platforms, since it only needs physical characteristics such as mass, mass of inertia, center of gravity, geometry and propeller-engine to configure the system. Further, additional novel algorithms are developed and added to the original CFBS control law design to make longitudinal and lateral-directional maneuvers safer and smoother. The proposed algorithm is implemented and tested in both numerical simulation and hardware-in-the-loop simulation (HILS). HILS is conducted by implementing the algorithm on the real UAV on-board hardware that consist of an embedded PC for image extraction and a microcontroller for the flight controller. The numerical simulation and HILS results show that the developed system is able to perform target tracking and pursuing task effectively.

Keywords: command filtered backstepping; fixed-wing unmanned aerial vehicle; image based visual servoing; target tracking; visual based flight controller.

1. Introduction

Object pursuing and tracking mission is a task commonly carried out by an unmanned aerial vehicle (UAV) which requires a capability of directing the tracking camera and managing the UAV maneuver accordingly. Fixed wing UAVs, although they have advantages of longer endurance and range compared to rotorcraft UAV, have limited maneuver capability due to the flight characteristics such as stall speed, bank angle limit, etc. This limitation may reduce the capability of fixed wing UAV to conduct object pursuing and tracking mission, especially for moving objects. A gimballed camera mechanism, as a tracking payload, may be employed to overcome this limitation. However, to guarantee that pursue and track mission can be accomplished by such system arrangement, a suitable control system, which can manage the maneuver of the UAV and the movement of the camera gimbal synchronously must be available.

The object tracking capability is started while an operator in the ground station select a suspicious unknown object on the ground/sea at the monitor screen. An image feature of the object is assumed always be available by using a feature extraction for long-term tracking task [1]. The position of the selected feature on the camera view plane becomes a measure for controlling the direction of gimbal mechanism and the UAV maneuver in order to pursue the target. In addition, the ground distance between the UAV and the object can be calculated by

Received: October 3rd, 2015. Accepted: December 7th, 2015
DOI: 10.15676/ijeei.2015.7.4.4

the camera attitude and the UAV altitude. Depending on the minimum loitering distance for each UAV platforms, then the UAV performs the target pursuing task by loitering the object autonomously.

Managing the object tracking task, UAV dynamic, feature motion, and pan-tilt gimbal movement models are also needed as part of the visual based flight control design. Several studies about the capability of flight dynamic [2][3][4] has been conducted for a linear flight control design and then a nonlinear dynamic modeling of the fixed-wing UAV [5] is chosen for a nonlinear flight control design. Further, the feature point motion and pan-tilt gimbal mechanism model [6] are developed based on image based visual servoing (IBVS) methodology [7][8][9].

Several studies about UAV using IBVS have also been conducted especially for rotary wing UAV to control its position and attitude with a camera fixed to its body frame [10] and for fixed-wing UAV to control landing guidance [11] and tracking linear infrastructure [12] with camera fixed to its body frame. Implementation of IBVS on fixed wing UAV with gimbaled camera mechanism to loiter the target object [13] has also been conducted using integral backstepping (IBS) control law [15] that adopts a simplified model of fixed-wing UAV [14]. However, the requirement of the analytical time-differential derivation for each term of the nonlinear model formulation made the IBS control law design to be problematic.

This paper presents the development of a new visual servoing scheme on a fixed-wing UAV to control target tracking task with camera attached to gimbal mechanism using command filtered backstepping (CFBS) [16][17] control law that adopts the full nonlinear model of fixed-wing UAV dynamic [5] and camera gimbal mechanism IBVS [6]. The additional advantage of CFBS compared with IBS is the easiness of control law design to adapt the complicated nonlinear model formulation without the necessity for solving time-differential derivation analytically. Furthermore, additional novel algorithms are then introduced to the original CFBS control law design to make longitudinal and lateral-directional maneuvers safer and smoother. Since the CFBS control law design adopted the nonlinear UAV dynamic model, the controller configuration can be adjusted by inputting the UAV physical characteristics i.e. mass, mass of inertia, center of gravity, geometry and propeller-engine without complicated control gain setting.

The proposed algorithm is then implemented on real hardware that consist of an embedded PC for processing the image extraction and a microcontroller for calculating the proposed IBVS algorithm using CFBS control law. Hardware-in-the-loop simulation (HILS) is also developed to test the real hardware performances. The numerical simulation and HILS prove the effectiveness of the proposed algorithm to perform target tracking and pursuing task as expected by loitering the target object autonomously.

The outline of the paper is in the following. Section 2 describes a nonlinear modeling of UAV Dynamic and IBVS. Section 3 discusses the control design of UAV dynamic and IBVS using CFBS. Section 4 shows the additional novel algorithms. Section 5 presents the SILS/HILS simulation results. Section 6 summarizes the paper and ongoing research.

2. UAV Dynamic and IBVS Modeling

The UAV flight dynamic model which involves some aerodynamic coefficients [18] is computed using physical geometry and mechanical properties data of the fixed-wing UAV. Power forces of thrust T is calculated using power and thrust coefficient functions from propeller-engine characteristics [19]. There are 4 coordinate frames that are used for constructing UAV dynamic equations, i.e. vehicle, stability, wind, and inertia frames [5]. The UAV dynamic are formulated in vehicle frame as follows [3][4][5]:

$$\dot{\mathrm{V}} = \frac{\mathrm{F}_{a_v}+\mathrm{F}_{p_v}+\mathrm{F}_{g_v}}{m} - \Omega \times \mathrm{V} + \frac{\mathrm{C}_{\mathrm{F}_{a_\delta}}}{m}\begin{bmatrix}\delta_e\\ \delta_a\\ \delta_r\end{bmatrix} + \frac{C_{\mathrm{F}_{p_\delta}}}{m}\delta_t \tag{1}$$

$$\dot{\Omega} = \mathrm{J}^{-1}\left(\mathrm{M}_{a_v} + \mathrm{M}_{p_v} - \Omega \times \mathrm{J}\Omega\right) + \mathrm{J}^{-1}\mathrm{C}_{\mathrm{M}_{a_\delta}}\begin{bmatrix}\delta_e\\ \delta_a\\ \delta_r\end{bmatrix} + \mathrm{J}^{-1}C_{\mathrm{M}_{p_\delta}}\delta_t \tag{2}$$

$$\dot{\Xi} = \mathrm{R}_\Omega\Omega \tag{3}$$

$$\dot{h} = \mathrm{R}_h\Omega \tag{4}$$

$$\dot{\chi} = \frac{g}{|\mathrm{V}|}\tan\phi\cos(\chi - \psi) \tag{5}$$

where, $\mathrm{V} = [U \quad V \quad W]^\intercal$, $\Omega = [p \quad q \quad r]^\intercal$, $\Xi = [\phi \quad \theta \quad \psi]^\intercal$, and h respectively are UAV velocity vector, angular velocity vector, attitude vector and sea-level altitude in vehicle frame; χ is UAV heading in inertia frame and g is gravitational constant; δ_e, δ_a, δ_r, and δ_t respectively are surface and power controls of elevator, aileron, rudder, and thrust; F_{a_v}, F_{p_v}, and F_{g_v} respectively are aerodynamic, power and gravity forces in vehicle frame; $\mathrm{C}_{\mathrm{F}_{a_\delta}}$ and $C_{\mathrm{F}_{p_\delta}}$ respectively are aerodynamic and power coeffiecient functions; M_{a_v} and M_{p_v} respectively are aerodynamic and power moments; $\mathrm{C}_{\mathrm{M}_{a_\delta}}$ and $C_{\mathrm{M}_{p_\delta}}$ respectively are aerodynamic and power coefficient functions; m and J respectively are mass and mass of inertia; U, V, and W respectively are velocity vector components in vehicle frame; p, q, and r respectively are angular velocity vector components in vehicle frame; ϕ, θ, and ψ respectively are roll, pitch, and yaw angles in vehicle frame.

There are 6 coordinate frames that are used for calculating IBVS model, i.e. image plane, camera, pan, tilt, vehicle, and inertia frames [6]. The camera image feature model is then written in image plane and vehicle frames as follows [6]:

$$\dot{\mathrm{p}} = \mathrm{L}(\mathrm{p}, z)\left({}^c\mathrm{R}_v^\Omega\begin{bmatrix}\mathrm{V}\\ \Omega\end{bmatrix} + {}^c\mathrm{R}_v^\Theta\mathrm{u}_\Theta\right) \tag{6}$$

$$\dot{\Theta} = \mathrm{u}_\Theta \tag{7}$$

where, $\mathrm{p} = [p_x \quad p_y]^\intercal$ is feature position on image plane; z is distance between camera and target object; $\mathrm{u}_\Theta = [u_{\theta_t} \quad u_{\psi_p}]^\intercal$ is pan-tilt gimbal mechanism control input vector; $\Theta = [\theta_t \quad \psi_p]^\intercal$ is pan-tilt angle vector of gimbal mechanism; L is Jacobian matrix function that is defined as follows:

$$\mathrm{L}(\mathrm{p}, z) = \begin{bmatrix} -\frac{\lambda}{z} & 0 & \frac{p_x}{z} & \frac{p_x p_y}{\lambda} & -\frac{\lambda^2 + p_x^2}{\lambda} & p_y \\ 0 & -\frac{\lambda}{z} & \frac{p_y}{z} & \frac{\lambda^2 + p_y^2}{\lambda} & -\frac{p_x p_y}{\lambda} & -p_x \end{bmatrix} \tag{8}$$

with λ denoting the focal length of camera characteristic; ${}^c\mathrm{R}_v^\Omega$ and ${}^c\mathrm{R}_v^\Theta$ respectively are transformation matrixes for UAV dynamic and pan-tilt gimbal mechanism that are defined as follows:

$${}^c\mathrm{R}_v^\Omega = \begin{bmatrix} {}^c\mathrm{R}_v & {}^c\mathrm{R}_v^\dagger \\ 0_{3\times3} & {}^c\mathrm{R}_v \end{bmatrix} \tag{9}$$

$${}^c\mathrm{R}_v^\Theta = \begin{bmatrix} {}^c\mathrm{R}_v^\ddagger \\ {}^c\mathrm{R}_v^\wedge \end{bmatrix} \tag{10}$$

$${}^c\mathrm{R}_v = {}^{\theta_t}\mathrm{R}_c{}^\intercal \, {}^{\psi_p}\mathrm{R}_{\theta_t}{}^\intercal \, {}^v\mathrm{R}_{\psi_p}{}^\intercal \tag{11}$$

$${}^c\mathrm{R}_v^\dagger = -\,{}^c\mathrm{R}_v\mathrm{S}({}^v\mathrm{T}_c) \tag{12}$$

$${}^c\mathrm{R}_v^\ddagger = \begin{bmatrix} 0 & -{}^{\theta_t}x_c & {}^{\theta_t}z_c \\ {}^{\psi_p}x_c & -{}^{\psi_p}y_c\sin\theta_t & -{}^{\psi_p}y_c\cos\theta_t \end{bmatrix}^\intercal \tag{13}$$

$${}^c\mathrm{R}_v^\wedge = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_t & -\sin\theta_t \end{bmatrix}^\intercal \tag{14}$$

$$ {}^{\theta_t}\mathrm{R}_c = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \tag{15}$$

$$ {}^{\psi_p}\mathrm{R}_{\theta_t} = \begin{bmatrix} \cos\theta_t & 0 & \sin\theta_t \\ 0 & 1 & 0 \\ -\sin\theta_t & 0 & \cos\theta_t \end{bmatrix} \tag{16}$$

$$ {}^{v}\mathrm{R}_{\psi_p} = \begin{bmatrix} \cos\psi_p & -\sin\psi_p & 0 \\ \sin\psi_p & \cos\psi_p & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{17}$$

where, $\mathrm{S}({}^{v}\mathrm{T}_c)$ is a skew-matrix of vector ${}^{v}\mathrm{T}_c$; ${}^{v}\mathrm{T}_c = {}^{v}\mathrm{T}_{\psi_p} + {}^{\psi_p}\mathrm{T}_{\theta_t} + {}^{\theta_t}\mathrm{T}_c$ is translation vector from camera frame to vehicle frame; ${}^{\psi_p}x_c = {}^{\psi_p}x_{\theta_t} + {}^{\theta_t}x_c$ and ${}^{\psi_p}y_c = {}^{\psi_p}y_{\theta_t} + {}^{\theta_t}y_c$ respectively are translation component of x and y axes from camera frame to pan frame.

CFBS Control Design

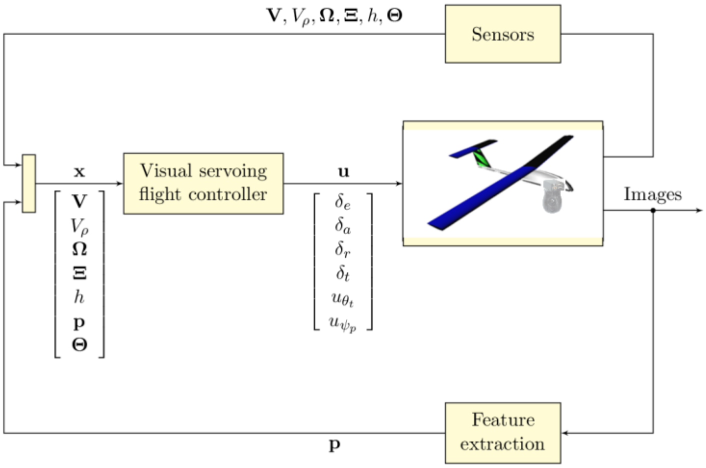

$\mathbf{V}, V_\rho, \mathbf{\Omega}, \mathbf{\Xi}, h, \mathbf{\Theta}$
Sensors
x
Visual servoing flight controller
u
Images
$\begin{bmatrix} \mathbf{V} \\ V_\rho \\ \mathbf{\Omega} \\ \mathbf{\Xi} \\ h \\ \mathbf{p} \\ \mathbf{\Theta} \end{bmatrix}$
$\begin{bmatrix} \delta_e \\ \delta_a \\ \delta_r \\ \delta_t \\ u_{\theta_t} \\ u_{\psi_p} \end{bmatrix}$
Feature extraction
p

Figure 1. Visual based autonomous flight control system design diagram.

Figure 1 shows a design diagram of visual based autonomous flight control system. The pan-tilt gimbal mechanism is installed between the camera and the body of UAV that allow the camera to move in both horizontal and vertical relative to the UAV fuselage. The UAV is assumed to not perform extreme maneuvers such that the bank and the pitch angles exceeds about 25 degrees. Sensors that are attached to UAV provide required data such as velocity V, pitot velocity V_ρ, angular velocity Ω, attitude Ξ, altitude h, and gimbal pan-tilt angles Θ. Camera is assumed to always look down capturing a selected target object on ground/sea. Feature extraction algorithm is also assumed to always produce an extracted feature point p from the selected target object during the operation of this system. Finally, a visual servoing flight controller calculates a control law algorithm based on all of UAV data and image feature position, denoted by $\mathrm{x} = \begin{bmatrix} \mathrm{V}^\intercal & V_\rho & \Omega^\intercal & \Xi^\intercal & h & \mathrm{p}^\intercal & \Theta^\intercal \end{bmatrix}^\intercal$, to produce UAV control surface deflection engine thrust variation, and gimbal mechanism pan-tilt movement, denoted by $\mathrm{u} = \begin{bmatrix} \delta_e & \delta_a & \delta_r & \delta_t & u_{\theta_t} & u_{\psi_p} \end{bmatrix}^\intercal$, for maintaining the feature point p at the center of camera image plane, and altitude h constant at h_c.

Figure 2 shows the detailed design diagram of visual servoing flight controller. There are 3 main controls, i.e. gimbal mechanism control, lateral flight control and longitudinal flight control. The gimbal mechanism control requires the data of V, Ω, Ξ, h, p, Θ to produce appropriate gimbal mechanism pan-tilt controls u_{θ_t}, u_{ψ_p} and a commanded heading χ_c for maintaining the feature point p to be at the center of camera image plane. Further, the lateral flight control needs V, V_ρ, Ω, Ξ, χ_c to produce appropriate UAV control surfaces deflections δ_a, δ_r for making the UAV heading to be at the commanded heading χ_c. Finally, the longitudinal flight control needs V_ρ, Ω, Ξ, h to produce appropriate UAV control surface and engine thrust variation δ_e, δ_t for maintaining the altitude h to be at the commanded altitude h_c.

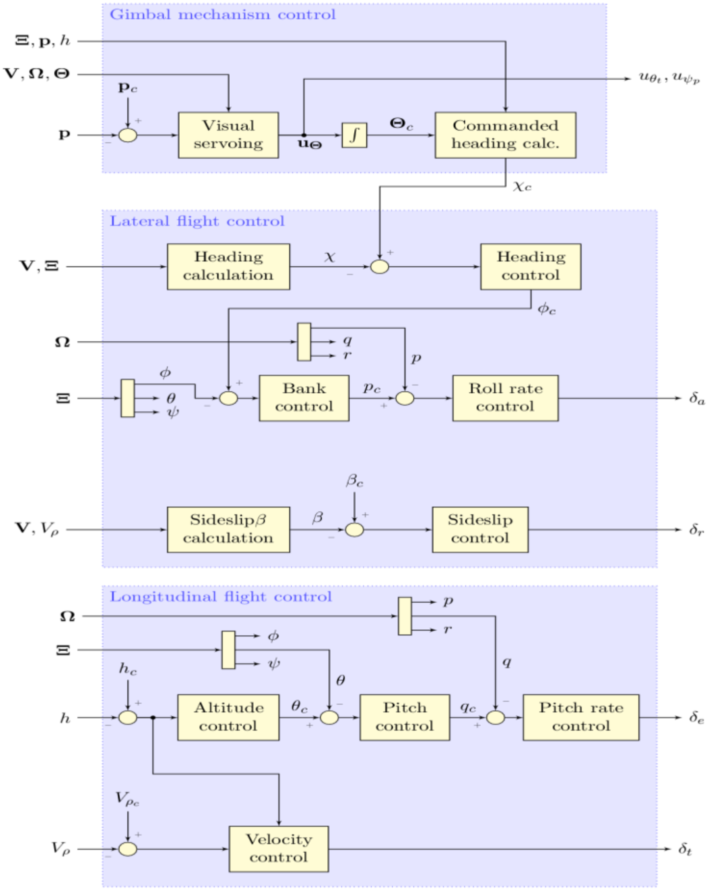
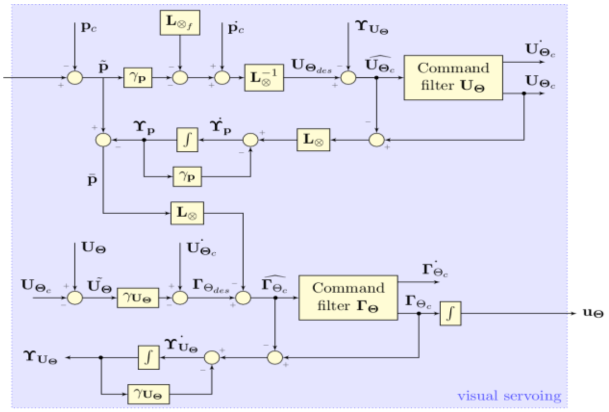
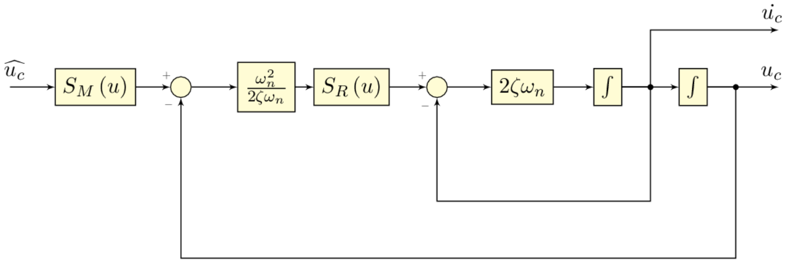


Figure 2. Visual servoing flight controller design diagram.

A. Gimbal Mechanism Control


$\mathbf{p}_c$
$\mathbf{L}_{\otimes_f}$
$\dot{\mathbf{p}}_c$
$\Upsilon_{\mathbf{U}_\Theta}$
$\dot{\mathbf{U}}_{\Theta_c}$
$\mathbf{p}$
$\tilde{\mathbf{p}}$
$\gamma_\mathbf{p}$
$\mathbf{L}_\otimes^{-1}$
$\mathbf{U}_{\Theta_{des}}$
$\widehat{\mathbf{U}_{\Theta_c}}$
Command filter $\mathbf{U}_\Theta$
$\mathbf{U}_{\Theta_c}$
$\Upsilon_\mathbf{p}$
$\int$
$\dot{\Upsilon}_\mathbf{p}$
$\mathbf{L}_\otimes$
$\gamma_\mathbf{p}$
$\bar{\mathbf{p}}$
$\mathbf{L}_\otimes$
$\mathbf{U}_\Theta$
$\dot{\mathbf{U}}_{\Theta_c}$
$\dot{\Gamma}_{\Theta_c}$
$\mathbf{U}_{\Theta_c}$
$\tilde{\mathbf{U}}_\Theta$
$\gamma_{\mathbf{U}_\Theta}$
$\Gamma_{\Theta_{des}}$
$\widehat{\Gamma_{\Theta_c}}$
Command filter Γ_Θ
Γ_{Θ_c}
$\int$
$\mathbf{u}_\Theta$
$\Upsilon_{\mathbf{U}_\Theta}$
$\int$
$\dot{\Upsilon}_{\mathbf{U}_\Theta}$
$\gamma_{\mathbf{U}_\Theta}$
visual servoing

(a) Two stages of CFBS control law.



(b) Command filter block in general.

Figure 3. Visual servoing design using CFBS control law.

Figure 3 shows a CFBS control law design for visual servoing block in gimbal mechanism control. Adapting to the CFBS design, IBVS model then is prepared from Equation (6) as follows:

$$\dot{\mathrm{p}} = \mathrm{L}_{\otimes_f} + \mathrm{L}_{\otimes}\mathrm{U}_{\Theta} \tag{18}$$

$$\dot{\mathrm{U}_{\Theta}} = \Gamma_{\Theta} \tag{19}$$

where, $\mathrm{U}_{\Theta} = \begin{bmatrix} U_{\theta_t} & U_{\psi_p} \end{bmatrix}^{\intercal}$ is defined as virtual control vector of gimbal mechanism pan-tilt; $\Gamma_{\Theta} = \begin{bmatrix} \zeta_{\theta_t} & \zeta_{\psi_p} \end{bmatrix}^{\intercal}$ is defined as output vector of backstepping control; $\mathrm{L}_{\otimes_f}$ and $\mathrm{L}_{\otimes}$ respectively are UAV dynamics variables depended transformation matrix and the gimbal pan-tilt variables depended transformation matrix that are defined as follows:

$$\mathrm{L}_{\otimes_f} = \mathrm{L}(\mathrm{p}, z)\ {}^{c}\mathrm{R}_{v}^{\Omega} \begin{bmatrix} \mathrm{V} \\ \Omega \end{bmatrix} \tag{20}$$

$$\mathrm{L}_{\otimes} = \mathrm{L}(\mathrm{p}, z)\ {}^{c}\mathrm{R}_{v}^{\Theta} \tag{21}$$

Residuals of image feature point and virtual control respectively are defined as follows:

$$\tilde{\mathrm{p}} = \mathrm{p} - \mathrm{p}_c \tag{22}$$

$$\tilde{\mathrm{U}}_{\Theta} = \mathrm{U}_{\Theta} - \mathrm{U}_{\Theta_c} \tag{23}$$

where, p_c is command feature point vector that is defined as the center position of camera image plane for aiming the target object; and U_{Θ_c} is commanded virtual control vector of pan-tilt gimbal mechanism. Based on the number of the residuals, two stages of CFBS control law are built for IBVS model as shown in figure 3(a). The first stage produces a commanded virtual control vector U_{Θ_c} and the second stage produces a commanded backstepping output Γ_{Θ_c}. Afterwards, the appropriate pan-tilt gimbal mechanism control input vector u_{Θ} then can be obtained from the commanded backstepping output Γ_{Θ_c} using time-integral transfer function.

In the first stage, a destination virtual control vector is chosen as a stabilization function vector to converge the feature point residual vector as follows:

$$\mathrm{U}_{\Theta_{des}} = \mathrm{L}_{\otimes}^{-1}\left(-\gamma_{\mathrm{p}}\tilde{\mathrm{p}} + \dot{\mathrm{p}_c} - \mathrm{L}_{\otimes_f}\right) \tag{24}$$

where, γ_{p} is a positive scalar gain for reducing feature point residual vector; and $\dot{\mathrm{p}_c}$ is commanded feature motion vector that is usually defined as zero vector in order to make the feature point stationary.

Furthermore, an unfiltered commanded virtual control vector $\widehat{\mathrm{U}_{\Theta_c}}$ is defined along with a low pass filter (LPF) transfer function as follows:

$$\widehat{\mathrm{U}_{\Theta_c}} = \mathrm{U}_{\Theta_{des}} - \Upsilon_{\mathrm{U}_{\Theta}} \tag{25}$$

$$\dot{\Upsilon}_{\mathrm{p}} = -\gamma_{\mathrm{p}}\Upsilon_{\mathrm{p}} + \mathrm{L}_{\otimes}\left(\mathrm{U}_{\Theta_c} - \widehat{\mathrm{U}_{\Theta_c}}\right) \tag{26}$$

where, $\Upsilon_{\mathrm{U}_\Theta}$ is a LPF output vector of a residual between filtered and unfiltered backstepping output vector in the second stage; U_{Θ_c} is a filtered commanded virtual control vector that is obtained from the transfer function of command filter U_Θ; and Υ_{p} is a LPF output vector of a residual between filtered and unfiltered commanded virtual control.
In addition, a compensated tracking error of the feature point residual is defined as follows:

$$\bar{\mathrm{p}} = \tilde{\mathrm{p}} - \Upsilon_{\mathrm{p}} \tag{27}$$

In the second stage, a destination backstepping output is chosen as a stabilization function vector to converge the virtual control residual vector as follows:

$$\Gamma_{\Theta_{des}} = -\gamma_{\mathrm{U}_\Theta}\widetilde{\mathrm{U}}_\Theta + \dot{\mathrm{U}}_{\Theta_c} \tag{28}$$

where, $\gamma_{\mathrm{U}_\Theta}$ is a positive scalar gain for reducing virtual control residual vector; and $\dot{\mathrm{U}}_{\Theta_c}$ is virtual control rate that is obtained from the transfer function of command filter U_Θ in the first stage.

An unfiltered commanded backstepping output vector $\widehat{\Gamma_{\Theta_c}}$ is then defined along with a LPF transfer function as follows:

$$\widehat{\Gamma_{\Theta_c}} = \Gamma_{\Theta_{des}} - \mathrm{L}_\otimes\bar{\mathrm{p}} \tag{29}$$

$$\dot{\Upsilon}_{\mathrm{U}_\Theta} = -\gamma_{\mathrm{U}_\Theta}\Upsilon_{\mathrm{U}_\Theta} + \Gamma_{\Theta_c} - \widehat{\Gamma_{\Theta_c}} \tag{30}$$

where, Γ_{Θ_c} is a filtered commanded backstepping output vector that is obtained from the transfer function of command filter Γ_Θ.

Figure 3(b) shows the transfer function of command filter block in general. Each component of unfiltered vector $\widehat{\mathrm{U}_{\Theta_c}}$ and $\widehat{\Gamma_{\Theta_c}}$ replaces the unfiltered signal input $\widehat{u_c}$ of command filter block. The command filter block produces both output u_c and output rate $\dot{u}_c$ of the filtered signal that is corresponded to each component of filtered vector U_{Θ_c}, Γ_{Θ_c} and filtered vector rate $\dot{\mathrm{U}}_{\Theta_c}$, $\dot{\Gamma}_{\Theta_c}$. The command filter block is defined as follows [16][17]:

$$\ddot{u}_c = 2\zeta\omega_n\left\{S_R\left(\frac{\omega_n}{2\zeta}\{S_M(\widehat{u_c}) - u_c\}\right) - \dot{u}_c\right\} \tag{31}$$

where, ω_n and ζ respectively are natural frequency and damping ratio of the flight control system; S_M and S_R respectively are magnitude and rate limit function that are defined as follows:

$$S_M(u) = \text{if}\begin{Bmatrix} u > M \\ -M \le u \le M \\ u < -M \end{Bmatrix}\text{then}\begin{Bmatrix} M \\ u \\ -M \end{Bmatrix} \tag{32}$$

$$S_R(u) = \text{if}\begin{Bmatrix} u > R \\ -R \le u \le R \\ u < -R \end{Bmatrix}\text{then}\begin{Bmatrix} R \\ u \\ -R \end{Bmatrix} \tag{33}$$

where, M and R respectively are the maximum value of the signal u and its rate.

Finally, the commanded gimbal mechanism pan-tilt angles Θ can be obtained from pan-tilt gimbal mechanism control input vector u_Θ using time-integral transfer function. Afterwards, the commanded UAV heading χ_c is calculated using the UAV attitude/altitude, the current feature position, and also the gimbal pan-tilt angles with the following algorithm:
Calculate distance and heading from camera to target object as follows:

$$d = \frac{h_{agl}}{\tan\{\sin^{-1}(-\mathrm{s}\,\theta\,\mathrm{c}\,\widehat{\theta_t}\,\mathrm{c}\,\widehat{\psi_p} + \mathrm{s}\,\phi\,\mathrm{c}\,\theta\,\mathrm{c}\,\widehat{\theta_t}\,\mathrm{s}\,\widehat{\psi_p} - \mathrm{c}\,\phi\,\mathrm{c}\,\theta\,\mathrm{s}\,\widehat{\theta_t})\}} \tag{34}$$

$$\chi_{cam} = \tan^{-1}\left\{\frac{(\mathrm{c}\,\theta\,\mathrm{s}\,\psi\,\mathrm{c}\,\widehat{\psi_p} + (\mathrm{s}\,\phi\,\mathrm{s}\,\theta\,\mathrm{s}\,\psi + \mathrm{c}\,\phi\,\mathrm{c}\,\psi)\,\mathrm{s}\,\widehat{\psi_p})\,\mathrm{c}\,\widehat{\theta_t} - (\mathrm{c}\,\phi\,\mathrm{s}\,\theta\,\mathrm{s}\,\psi - \mathrm{s}\,\phi\,\mathrm{c}\,\psi)\,\mathrm{s}\,\widehat{\widehat{\theta_t}}}{(\mathrm{c}\,\theta\,\mathrm{c}\,\psi\,\mathrm{c}\,\widehat{\psi_p} + (\mathrm{s}\,\phi\,\mathrm{s}\,\theta\,\mathrm{c}\,\psi - \mathrm{c}\,\phi\,\mathrm{s}\,\psi)\,\mathrm{s}\,\widehat{\psi_p})\,\mathrm{c}\,\widehat{\theta_t} - (\mathrm{c}\,\phi\,\mathrm{s}\,\theta\,\mathrm{c}\,\psi + \mathrm{s}\,\phi\,\mathrm{s}\,\psi)\,\mathrm{s}\,\widehat{\theta_t})}\right\} \tag{35}$$

where, s and c respectively are the abbreviations of sin and cos functions; h_{agl} is altitude from ground/sea; $\widehat{\theta_t}$ and $\widehat{\psi_p}$ respectively are compensated tilt and pan angles due to the current feature point position that are defined as follows:

$$\widehat{\theta_t} = \theta_t - \tan^{-1}\left(\frac{p_y}{\lambda}\right) \tag{36}$$

$$\widehat{\psi_p} = \psi_p + \tan^{-1}\left(\frac{p_x}{p_y \sin\theta_t + \lambda\cos\theta_t}\right) \tag{37}$$

Decide a loiter radius using an appropriate UAV bank angle ϕ_{loiter} as follows:

$$R_{loiter} = \frac{{V_{\rho_c}}^2}{g\tan\phi_{loiter}} \tag{38}$$

where, V_{ρ_c} is a commanded UAV wind velocity.
Calculate the commanded heading as follows:

$$\chi_c = \chi_{cam} \pm \tan^{-1}\left(\frac{\pi/_{18}}{\frac{d}{R_{loiter}}-1}\right) \tag{39}$$

where, operator $\pm$ must be decided whether using minus (-) operator for clock-wise loitering or using plus (+) operator for counter-clock-wise loitering.

B. Lateral Flight Control

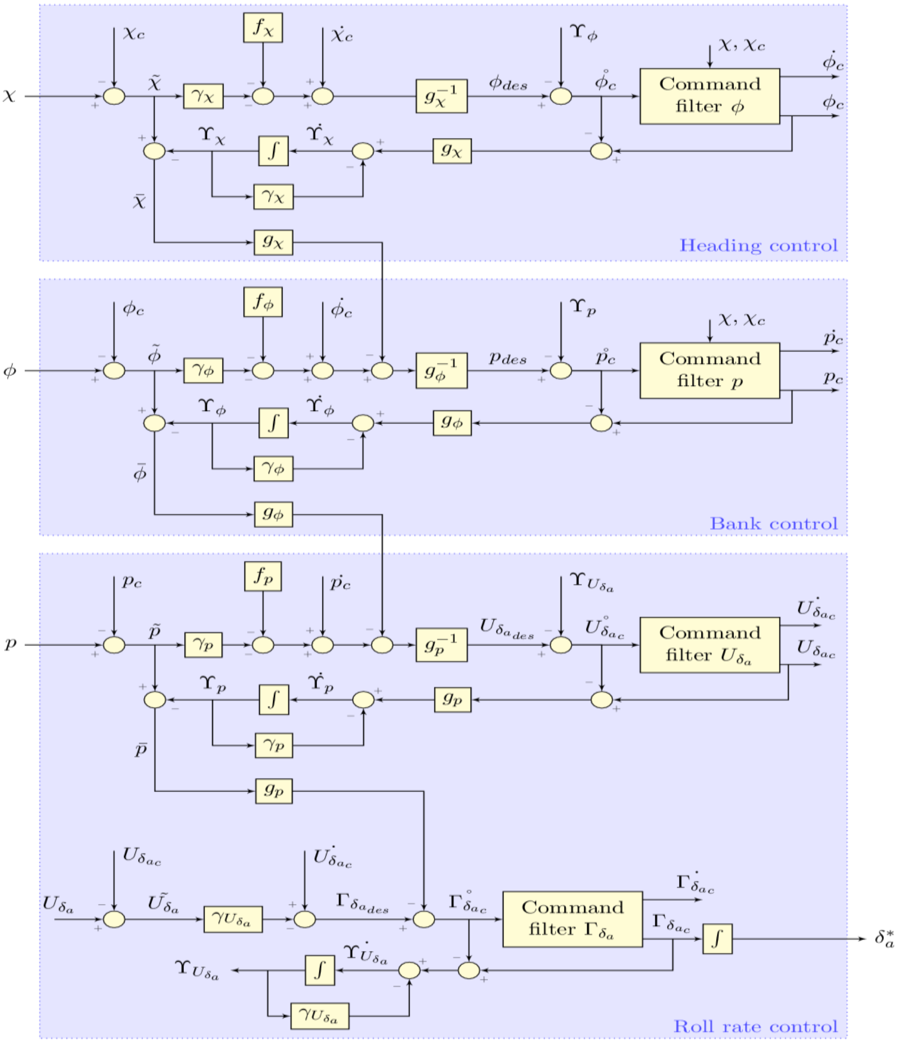


Figure 4. Lateral flight control design using CFBS control law.

Figure 4 shows a CFBS control law design for lateral flight control. Adapting to CFBS design, lateral dynamic model then is prepared from Equations (1) - (3), (5) as follows:

$$\dot{\chi} = f_\chi + g_\chi \phi \tag{40}$$

$$\dot{\phi} = f_\phi + g_\phi p \tag{41}$$

$$\dot{p} = f_p + g_p U_{\delta_a} \tag{42}$$

$$\dot{U}_{\delta_a} = f_{U_{\delta_a}} + g_{U_{\delta_a}} \Gamma_{\delta_a} \tag{43}$$

where, U_{δ_a} and Γ_{δ_a} respectively are virtual control and backstepping output of aileron; f_χ, g_χ, f_ϕ, g_ϕ, f_p, g_p, $f_{U_{\delta_a}}$, and $g_{U_{\delta_a}}$ respectively are nonlinear parts of heading, bank, roll rate and aileron virtual control models that are defined as follows:

$$f_\chi = 0 \tag{44}$$

$$g_\chi = \frac{g}{|\mathrm{V}|} \cos(\chi - \psi) \tag{45}$$

$$f_\phi = (q \sin\phi + r \cos\phi) \tan\theta \tag{46}$$

$$g_\phi = 1 \tag{47}$$

$$f_p = f_{\Omega_{lat,1}} \tag{48}$$

$$g_p = g_{\Omega_{lat,1}} \tag{49}$$

$$f_{U_{\delta_a}} = 0 \tag{50}$$

$$g_{U_{\delta_a}} = 1 \tag{51}$$

where, $f_{\Omega_{lat,1}}$ and $g_{\Omega_{lat,1}}$ respectively are angular velocity vector components of $\mathrm{F}_{\Omega_{lat}} = \left[f_{\Omega_{lat,1}} \quad f_{\Omega_{lat,2}} \quad f_{\Omega_{lat,3}}\right]^\intercal$ and $\mathrm{G}_{\Omega_{lat}} = [g_{\Omega_{lat,1}} \quad g_{\Omega_{lat,2}} \quad g_{\Omega_{lat,3}}]^\intercal$ that are defined from Equation (2) as follows:

$$\mathrm{F}_{\Omega_{lat}} = \mathrm{J}^{-1}\left(\mathrm{M}_{a_v} + \mathrm{M}_{p_v} - \Omega \times \mathrm{J}\Omega\right) + \mathrm{J}^{-1}\mathrm{C}_{\mathrm{M}_{a_\delta}} \begin{bmatrix} \delta_e \\ 0 \\ \delta_r \end{bmatrix} + \mathrm{J}^{-1} C_{\mathrm{M}_{p_\delta}} \delta_t \tag{52}$$

$$\mathrm{G}_{\Omega_{lat}} = \mathrm{J}^{-1}\mathrm{C}_{\mathrm{M}_{a_\delta}} \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \tag{53}$$

Based on four residuals of heading, bank, roll rate, and aileron virtual control, four stages of CFBS control law are built for lateral flight controller shown in Figure 4. The design procedures are the same as described by Equations (19) – (27) from defining residuals, choosing stabilization function, calculating unfiltered and filtered signals until defining LPF output for the previous and next stages. The variables γ_χ, γ_ϕ, γ_p, and $\gamma_{U_{\delta_a}}$ respectively are positive scalar gains for reducing residuals of heading, bank, roll rate, and aileron virtual control. The required deflection of aileron can be obtained from aileron backstepping output using time-integral transfer function. The aileron signal will be processed further using our proposed algorithm to make lateral-directional maneuvers safer. In addition, the maximum value for command filter ϕ and p will also be calculated to provide lateral-directional maneuvers smoother. This paper assumes no windy circumstance, hence the surface control of rudder is set to zero.

C. Longitudinal Flight Control

Figure 5 shows a CFBS control law design for longitudinal flight control. Adapting to CFBS design, longitudinal dynamic model then is prepared from Equations (1) - (4) as follows:

$$\dot{h} = f_h + g_h \theta \tag{54}$$

$$\dot{\theta} = f_\theta + g_\theta q \tag{55}$$

$$\dot{q} = f_q + g_q U_{\delta_e} \tag{56}$$

$$\dot{U}_{\delta_e} = f_{U_{\delta_e}} + g_{U_{\delta_e}} \Gamma_{\delta_e} \tag{57}$$

where, U_{δ_e} and Γ_{δ_e} respectively are virtual control and backstepping output of elevator; f_h, g_h, f_θ, g_θ, f_q, g_q, $f_{U_{\delta_e}}$, and $g_{U_{\delta_e}}$ respectively are nonlinear parts of altitude, pitch, pitch rate and elevator virtual control models that are defined as follows:

$$f_h = -(V \sin\phi + W \cos\phi)\cos\theta \tag{58}$$

$$g_h = U \tag{59}$$

$$f_\theta = -r\sin\phi \tag{60}$$

$$g_\theta = \cos\phi \tag{61}$$

$$f_q = f_{\Omega_{lon,2}} \tag{62}$$

$$g_q = g_{\Omega_{lon,2}} \tag{63}$$

$$f_{U_{\delta_e}} = 0 \tag{64}$$

$$g_{U_{\delta_e}} = 1 \tag{65}$$

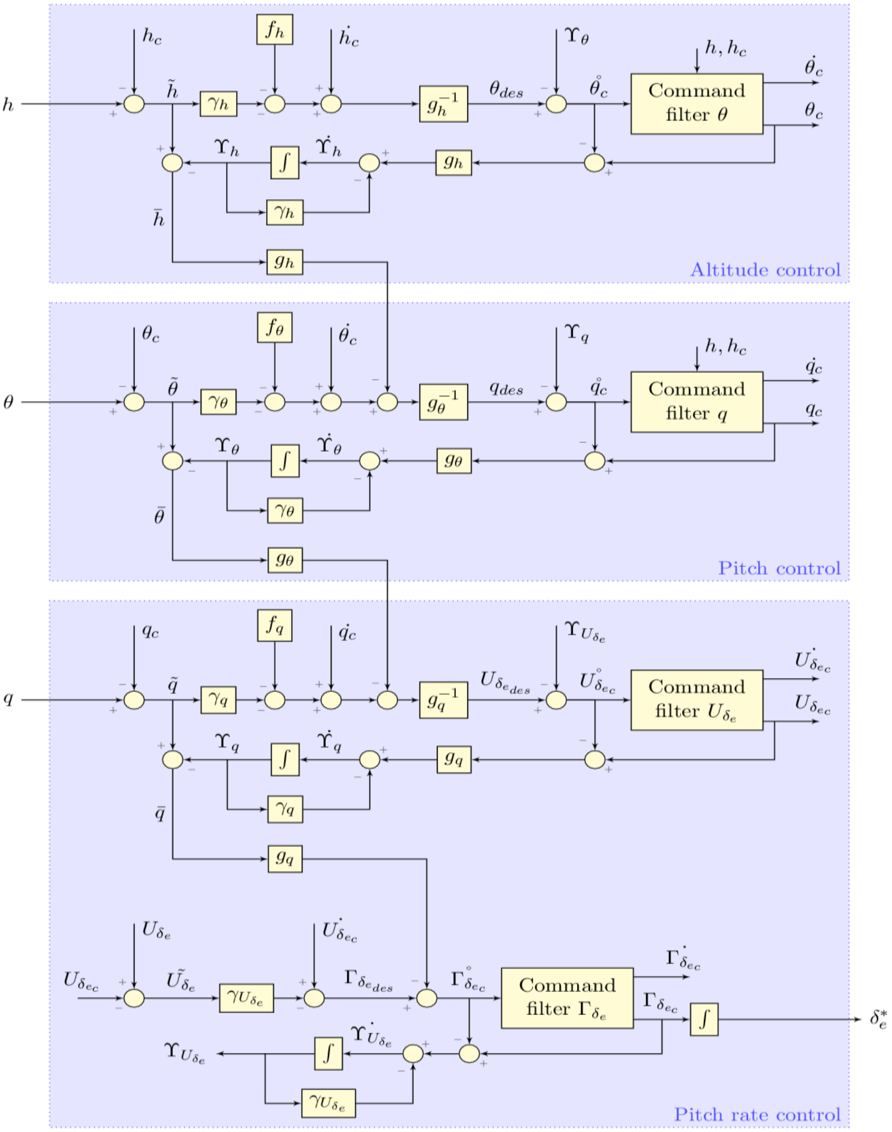


Figure 5. Longitudinal flight control design using CFBS control law.

where, $f_{\Omega_{lon,2}}$ and $g_{\Omega_{lon,2}}$ respectively are angular velocity vector components of $\mathrm{F}_{\Omega_{lon}} = \left[f_{\Omega_{lon,1}} \quad f_{\Omega_{lon,2}} \quad f_{\Omega_{lon,3}}\right]^{\intercal}$ and $\mathrm{G}_{\Omega_{lon}} = [g_{\Omega_{lon,1}} \quad g_{\Omega_{lon,2}} \quad g_{\Omega_{lon,3}}]^{\intercal}$ that are defined from Equation (2) as follows:

$$\mathrm{F}_{\Omega_{lon}} = \mathrm{J}^{-1}\left(\mathrm{M}_{a_v} + \mathrm{M}_{p_v} - \Omega \times \mathrm{J}\Omega\right) + \mathrm{J}^{-1}\mathrm{C}_{\mathrm{M}_{a_\delta}}\begin{bmatrix} 0 \\ \delta_a \\ \delta_r \end{bmatrix} + \mathrm{J}^{-1}C_{\mathrm{M}_{p_\delta}}\delta_t \tag{66}$$

$$\mathrm{G}_{\Omega_{lon}} = \mathrm{J}^{-1}\mathrm{C}_{\mathrm{M}_{a_\delta}}\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{67}$$

Based on four residuals of altitude, pitch, pitch rate, and elevator virtual control, four stages of CFBS control law are built for longitudinal flight controller shown in Figure 5. The design procedures are the same as described by Equation (19) – (27) from defining residuals, choosing stabilization function, calculating unfiltered and filtered signals until defining LPF output for the previous and next stages. The variables γ_h, γ_θ, γ_q, and $\gamma_{U_{\delta_e}}$ respectively are positive scalar gains for reducing residuals of altitude, pitch, pitch rate, and elevator virtual control.

The required deflection of elevator can be obtained from elevator backstepping output using time-integral transfer function. The elevator signal will be processed further using our proposed algorithm to make longitudinal maneuvers safer. In addition, the maximum values for command filter θ and q will be also calculated to provide longitudinal maneuvers smoother. This paper assumes no windy circumstance, hence the thrust control of throttle is set to fix while maintaining the altitude. The thrust control variation δ_t is only depended on the UAV altitude using algorithm as follows:

$$\delta_t = \text{if}\begin{Bmatrix} h > h_c + \Delta h \\ h_c - \Delta h \leq h \leq h_c + \Delta h \\ h < h_c - \Delta h \end{Bmatrix}\text{then}\begin{cases} \text{max} \\ \text{normal} \\ \text{min} \end{cases} \tag{68}$$

where, Δh is a margin altitude for maintaining the UAV altitude.

4. Proposed Algorithm

Maximum values of magnitude and rate for each command filter in the original CFBS design need to be set properly. However, there is no detailed description [16][17] about how these variables should be set especially for the CFBS design with many stages. Sometimes these values are set intuitively according to reasonable physical values, i.e. the maximum of bank, roll rate, pitch, or pith rate. However, improper adjustments of these maximum values in the following stages will make the residual in the first stage will not be converged because the backstepping output in the last stage may induces an excessive signal for the first stage although the value of backstepping output in the last stage is still within its maximum value. To avoid this problem, two novel algorithms are introduced to the original CFBS design of lateral and longitudinal flight control in this paper.

A. Safe maneuver algorithm

An algorithm to make lateral maneuver safer for CFBS design of lateral flight control is developed as follows:

1. Limit the value of the commanded aileron deflection from original CFBS control law
   $$\breve{\delta_a} = S_M({\delta_a}^*), \text{with } M = M_{\delta_a} \tag{69}$$
   where, M_{δ_a} is maximum safe value of aileron δ_a that is set in command filter U_{δ_a}.
2. Limit the roll rate due to the commanded aileron deflection
   $$\breve{p} = S_M\left(p + \left(f_p + g_p\breve{\delta_a}\right)\Delta T\right), \text{with } M = M_p \tag{70}$$
   where, M_p is maximum safe value of roll rate p that is set in command filter p; ΔT is time sampling for digital calculation.
3. Limit the bank due to the commanded roll rate

$$\breve{\phi} = S_M\big(\phi + \big(f_\phi + g_\phi \breve{p}\big)\Delta T\big), \text{with } \; M = M_\phi \tag{71}$$

where, M_ϕ is maximum safe value of bank ϕ that is set in command filter ϕ.

4. The safe commanded aileron deflection can be obtained as follows:

$$\delta_a = S_M \left(\frac{\frac{\frac{\frac{\breve{\phi}-\phi}{\Delta T} - f_\phi}{g_\phi}}{\Delta T} - f_p}{g_p} \right), \text{with } M = M_{\delta_a} \tag{72}$$

An algorithm to make longitudinal maneuver safer for CFBS design of longitudinal flight control is developed as follows:

1. Limit the value of the commanded elevator deflection from original CFBS control law

$$\breve{\delta_e} = S_M(\delta_e{}^*), \text{with } \; M = M_{\delta_e} \tag{73}$$

where, M_{δ_e} is maximum safe value of elevator δ_e that is set in command filter U_{δ_e}.

2. Limit the pitch rate due to the commanded elevator deflection

$$\breve{q} = S_M\big(q + \big(f_q + g_q \breve{\delta_e}\big)\Delta T\big), \text{with } \; M = M_q \tag{74}$$

where, M_q is maximum safe value of pitch rate q that is set in command filter q.

3. Limit the pitch due to the commanded pitch rate

$$\breve{\theta} = S_M(\theta + (f_\theta + g_\theta \breve{q})\Delta T), \text{with } \; M = M_\theta \tag{75}$$

where, M_θ is maximum safe value of pitch θ that is set in command filter θ.

4. The safe commanded elevator deflection can be obtained as follows:

$$\delta_e = S_M \left(\frac{\frac{\frac{\frac{\breve{\theta}-\theta}{\Delta T} - f_\theta}{g_\theta}}{\Delta T} - f_q}{g_q} \right), \; \text{with } M = M_{\delta_e} \tag{76}$$

B. Smooth maneuver algorithm

In the original CFBS design, the maximum safe values of bank and roll rate for lateral flight controller are usually fixed. However, improper set values induce unexpected responses of the lateral dynamic of the UAV. Larger set values inflict quick responses but the control has difficulty in converging the heading residual. Smaller set values inflict slow responses but the control can converge the heading residual smoothly. However both quicker responses and smoother convergence are needed to guide the UAV to follow the commanded heading as soon as possible. The same problem also happens for longitudinal dynamic of the UAV while setting the maximum safe values of pitch and pitch roll.

An algorithm to make lateral maneuver smoother for CFBS design of lateral flight control is then developed as follows:

1. After the calculation of the commanded heading χ_c and current heading χ, the maximum safe value of bank M_ϕ in command filter ϕ is set as follows:

$$M_\phi = \max \; \left(\min \; \left(M_{\phi_{max}} \frac{|\chi-\chi_c|}{\Delta\chi}, M_{\phi_{max}}\right), M_{\phi_{min}}\right) \tag{77}$$

where, $\Delta\chi$ is a margin heading for maintaining the UAV heading; $M_{\phi_{max}}$ and $M_{\phi_{min}}$ respectively are largest and smallest maximum safe value of bank to converge the heading residual smoothly.

2. The maximum safe rate of bank in command filter ϕ and the maximum safe value of roll rate in command filter p are then set as follows:

$$R_\phi = M_p = \max \; \left(\min \; \left(M_{p_{max}} \frac{|\chi-\chi_c|}{\Delta\chi}, M_{p_{max}}\right), M_{p_{min}}\right) \tag{78}$$

where, $M_{p_{max}}$ and $M_{p_{min}}$ respectively are largest and smallest maximum safe value of roll rate to converge the heading residual smoothly.

An algorithm to make longitudinal maneuver smoother for CFBS design of longitudinal flight control is then developed as follows:

1. The maximum safe value of pitch M_θ in command filter θ is set as follows:
   $M_\theta = \max\ \left(\min\ \left(M_\theta \frac{|h-h_c|}{\Delta h}, M_\theta\right), M_{\theta_{min}}\right)$ (79)
   where, $M_{\theta_{min}}$ is smallest maximum safe value of pitch to converge the altitude residual smoothly.
2. The maximum safe rate of pitch in command filter θ and the maximum safe value of pitch rate in command filter q are then set as follows:
   $R_\theta = M_q = \max\ \left(\min\ \left(M_q \frac{|h-h_c|}{\Delta h}, M_q\right), M_{q_{min}}\right)$ (80)
   where, $M_{q_{min}}$ is smallest maximum safe value of pitch rate to converge the altitude residual smoothly.

5. Simulation Results

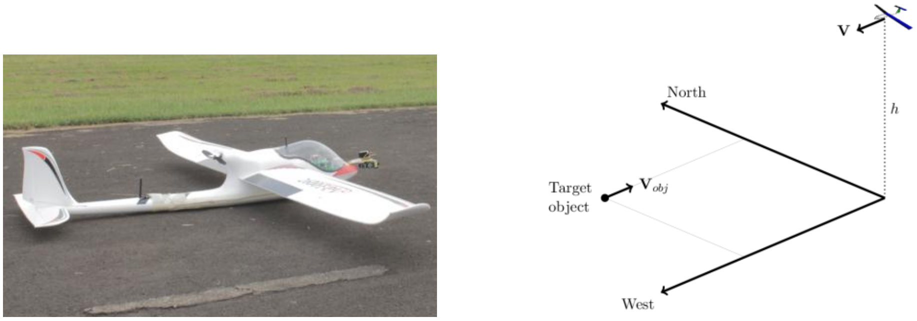
Table 1. FPV PowerZone 2600 characteristics

Parameter	Symbol	Values
MTOW	m	3 kg
c.g. position		28% MAC
MAC		25.187 cm
Main wing	S_w	0.65433 m^2
	b_w	2.6 m
Horizontal tail plane	S_h	0.091 m^2
	b_h	0.7 m
Vertical tail plane	S_h	0.03675 m^2
	b_h	0.245 m
Mass of inertia	J_{xx}	0.790523 kg m^2
	J_{yy}	0.47382 kg m^2
	J_{zz}	1.009734 kg m^2
	J_{xz}	0.015856 kg m^2
Propeller-Engine	P_{max}	2.28 kW
	D_{prop}	25.6 cm

Small UAV platform characteristics of FPV PowerZone 2600 shown in Figure 6(a) and Table 1 is used for numerical simulation. A simulation scenario shown in Figure 6(b) is conducted with initial parameters written in Table 2. Initially, the UAV cruises in steady state westward and the camera catch a target object feature at the top-right of the camera image plane when the simulation is started.

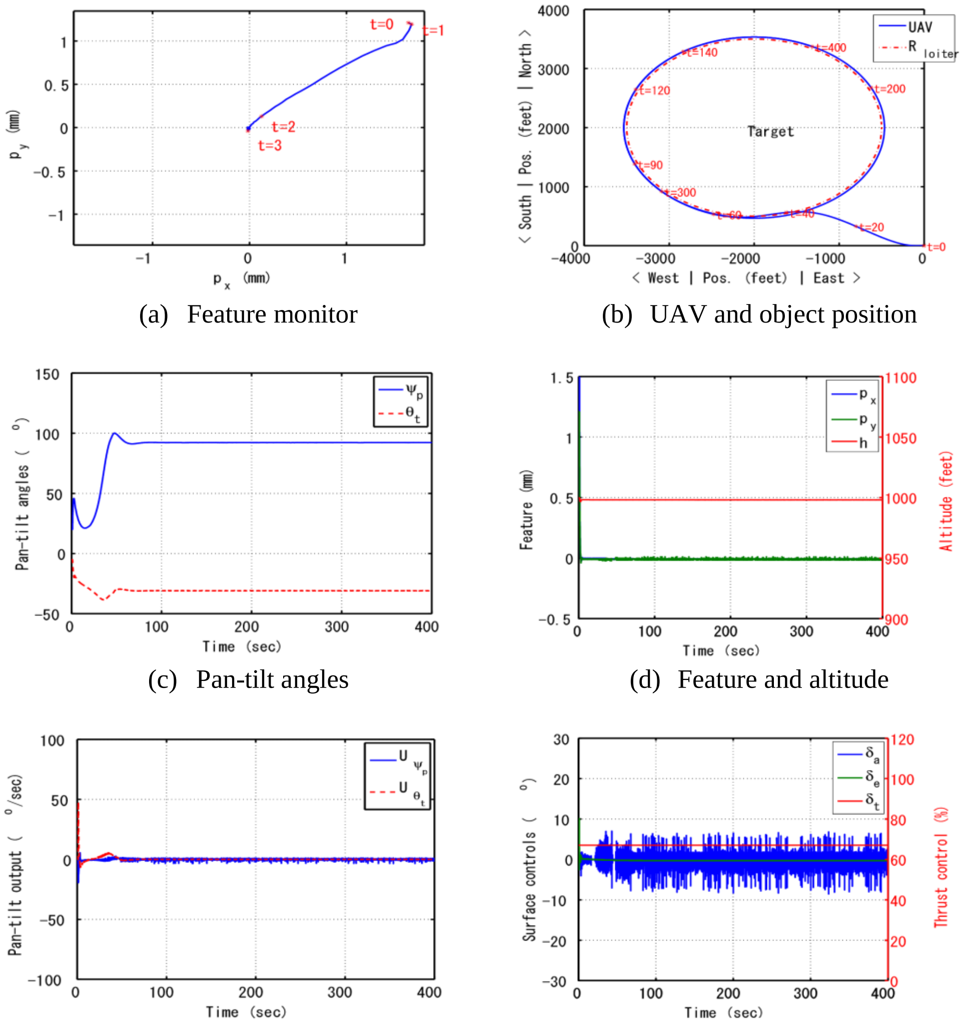
A. Numerical Simulation

Two simulation scenarios are conducted with stationary and moving target object as shown in Figure 6(b). Figure 7 shows the simulation results with stationary target object, while Figure 8 shows the simulation results with moving target object. Both simulations converge the feature and altitude residuals as shown in Subfigures (a) and (d) which the responded feature position converge to the commanded feature position of image plane center and the responded altitude converge to the commanded altitude. They show the effectiveness of CFBS control law to converge the residuals that adapt both UAV dynamic and IBVS models simultaneously.



(a) Platform of FPV PowerZone 2600. (b) Simulation scenario

Figure 6. UAV platform and the simulation scenario.



(a) Feature monitor (b) UAV and object position

(c) Pan-tilt angles (d) Feature and altitude

(e) Pan-tilt controls. (f) UAV controls.

Figure 7. Numerical simulation results for loitering the stationary target object

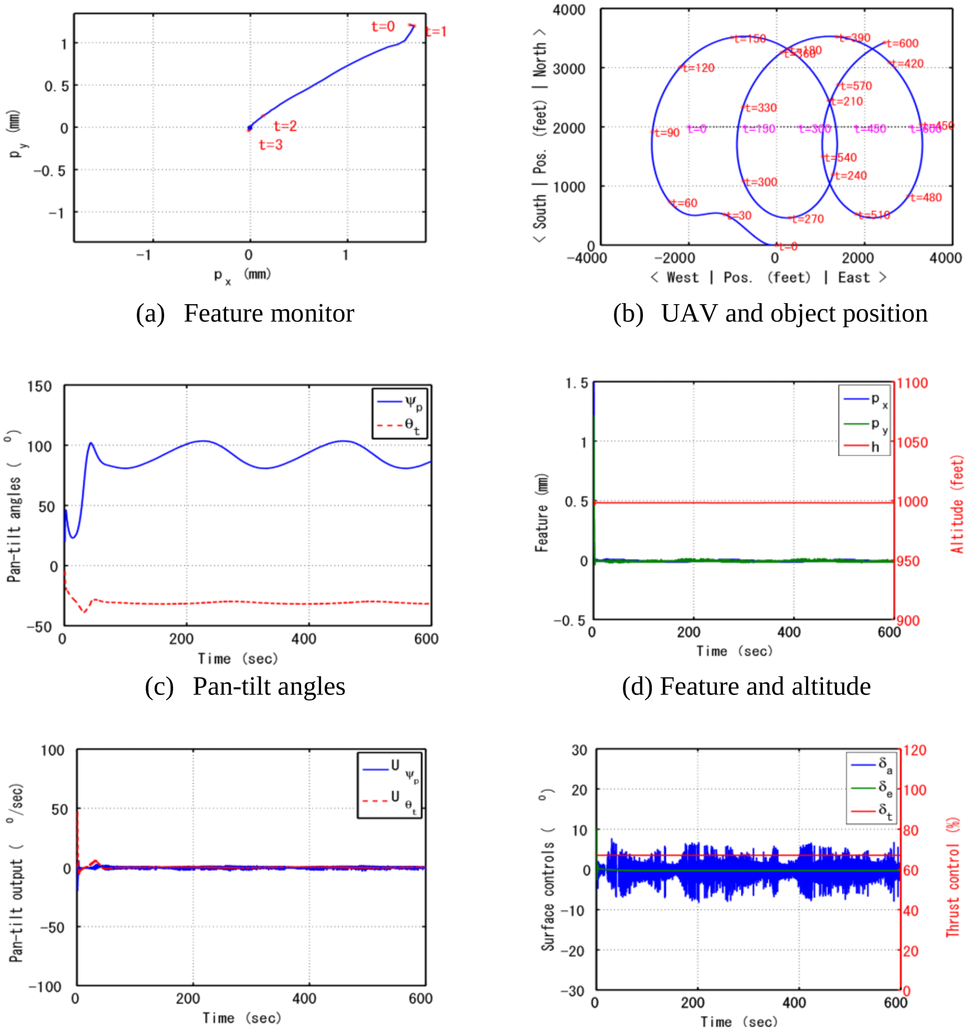
As a result of the proposed algorithm, the target object can be tracked autonomously by the fixed-wing UAV as it has been instructed for clock-wise loitering as shown in Figure 7(b) for the stationary target object and Figure 8(b) for the moving target object. The gimbal mechanism pan-tilt angles have also good responses to direct the camera always pointing to the target object as shown in Subfigure (c) for both simulations. When entering the loiter track, the pan angle consistently is at 90^{o} for the stationary target object and fluctuated around 90^{o} for the moving target object. The commanded controls as shown in Subfigures (e) and (f) also work great to drive the camera always pointing to the target object and to make the longitudinal and lateral-directional of the fixed-wing UAV maneuvers safer and smoother for following the commanded heading and altitude.

Table 2. Simulation parameters.

Parameter	Symbol	Values
Initial UAV position	$north$	0 feet
	$east$	0 feet
Initial UAV altitude	h	1000 feet
Initial object position	$north$	2000 feet
	$east$	-2000 feet
Object altitude	h_{obj}	0 feet
Object velocity (while moving)	V_{obj}	5 knot
Loiter radius	R_{loiter}	1500 feet
Commanded UAV altitude	h_c	1000 feet
Initial gimbal mechanism pan-tilt	θ_t	-5 ^{o}
	ψ_p	20 ^{o}
Throttle	$\delta_{t_{max}}$	100 %
	δ_t	67 %
	$\delta_{t_{min}}$	20 %
Initial elevator and aileron	δ_e, δ_a	0 ^{o}
Frequency	$\omega_n/2\pi$	25 Hz
Damping ratio	ζ	3
Loiter direction		Clock-wise
Lateral bank limit	$M_{\phi_{max}}$	25 ^{o}
	$M_{\phi_{min}}$	1 ^{o}
Lateral roll rate limit	$M_{p_{max}}$	10 ^{o}/sec
	$M_{p_{min}}$	0.1 ^{o}/sec
Longitudinal pitch limit	$M_{\theta_{max}}$	10 ^{o}
	$M_{\theta_{min}}$	4 ^{o}
Longitudinal pitch rate limit	$M_{q_{max}}$	10 ^{o}/sec
	$M_{q_{min}}$	0.1 ^{o}/sec
Aileron limit	$M_{U_{\delta_a}}$	20 ^{o}
Aileron rate limit	$M_{\Gamma_{\delta_a}}$	25 ^{o}/sec
Elevator limit	$M_{U_{\delta_e}}$	20 ^{o}
Elevator rate limit	$M_{\Gamma_{\delta_e}}$	25 ^{o}/sec
Margin heading	$\Delta\chi$	90 ^{o}
Margin altitude	Δh	5 feet

As a result of the proposed algorithm, the target object can be tracked autonomously by the fixed-wing UAV as it has been instructed for clock-wise loitering as shown in Figure 7(b) for the stationary target object and Figure 8(b) for the moving target object. The gimbal mechanism pan-tilt angles have also good responses to direct the camera always pointing to the target object as shown in Subfigure (c) for both simulations. When entering the loiter track, the pan angle consistently is at 90^{o} for the stationary target object and fluctuated around 90^{o} for the

moving target object. The commanded controls as shown in Subfigures (e) and (f) also work great to drive the camera always pointing to the target object and to make the longitudinal and lateral-directional of the fixed-wing UAV maneuvers safer and smoother for following the commanded heading and altitude.



(a) Feature monitor

(b) UAV and object position

(c) Pan-tilt angles

(d) Feature and altitude

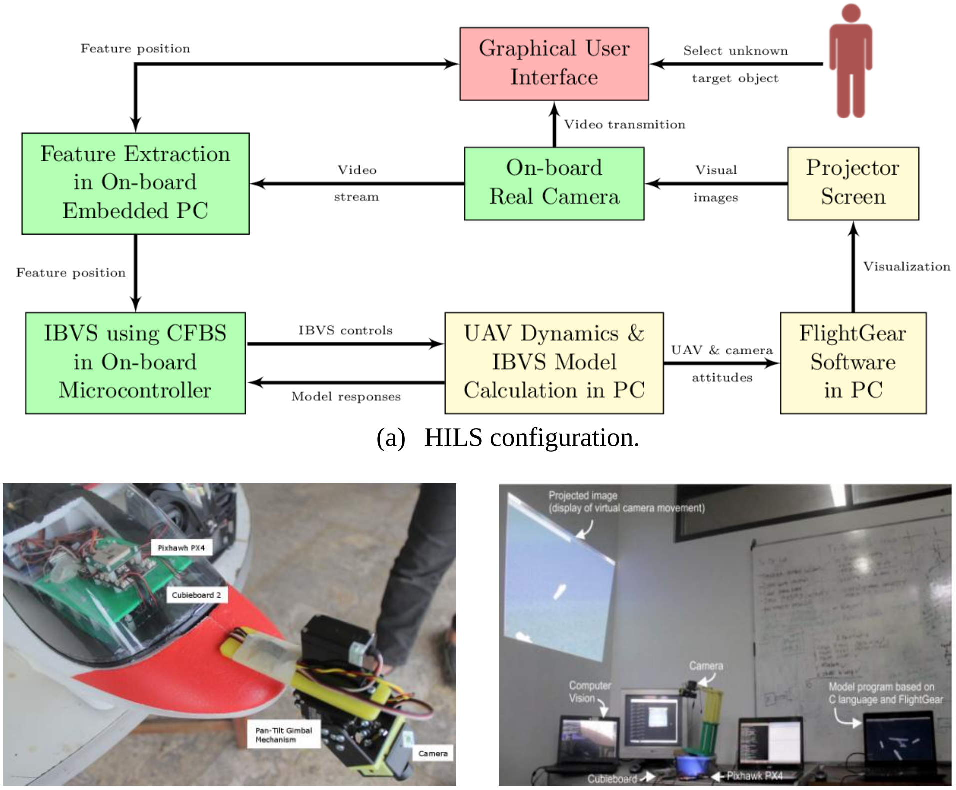
(e) Pan-tilt controls

(f) UAV controls

Figure 8. Numerical simulation results for loitering the moving target object.

B. Hardware-In-the-Loop Simulation

Figure 9 shows the development of hardware-in-the-loop simulation (HILS) for testing the implementation of the developed scheme on the on-board hardware before the real flight test. Figure 9(b) shows the on-board hardware whose total weight is less than 500g including an additional battery. Pixhawk PX4 flight controller board is employed for implementing the proposed IBVS algorithm using CBFS, while Cubieboard 2 embedded PC is employed for implementing the feature extraction of TLD algorithm. On-board camera is attached to pan-tilt gimbal mechanism. The captured images is transmitted to on-ground GUI PC and is also streamed to Cubieboard 2 embedded PC.


Graphical User Interface
Select unknown target object
Feature position
Video transmition
Feature Extraction in On-board Embedded PC
Video stream
On-board Real Camera
Visual images
Projector Screen
Feature position
Visualization
IBVS using CFBS in On-board Microcontroller
IBVS controls
Model responses
UAV Dynamics & IBVS Model Calculation in PC
UAV & camera attitudes
FlightGear Software in PC
Pixhawh PX4
Cubieboard 2
Pan-Tilt Gimbal Mechanism
Camera
Projected image (display of virtual camera movement)
Camera
Computer Vision
Model program based on C language and FlightGear
Cubieboard
Pixhawk PX4

(a) HILS configuration.

(b) On-board hardware.

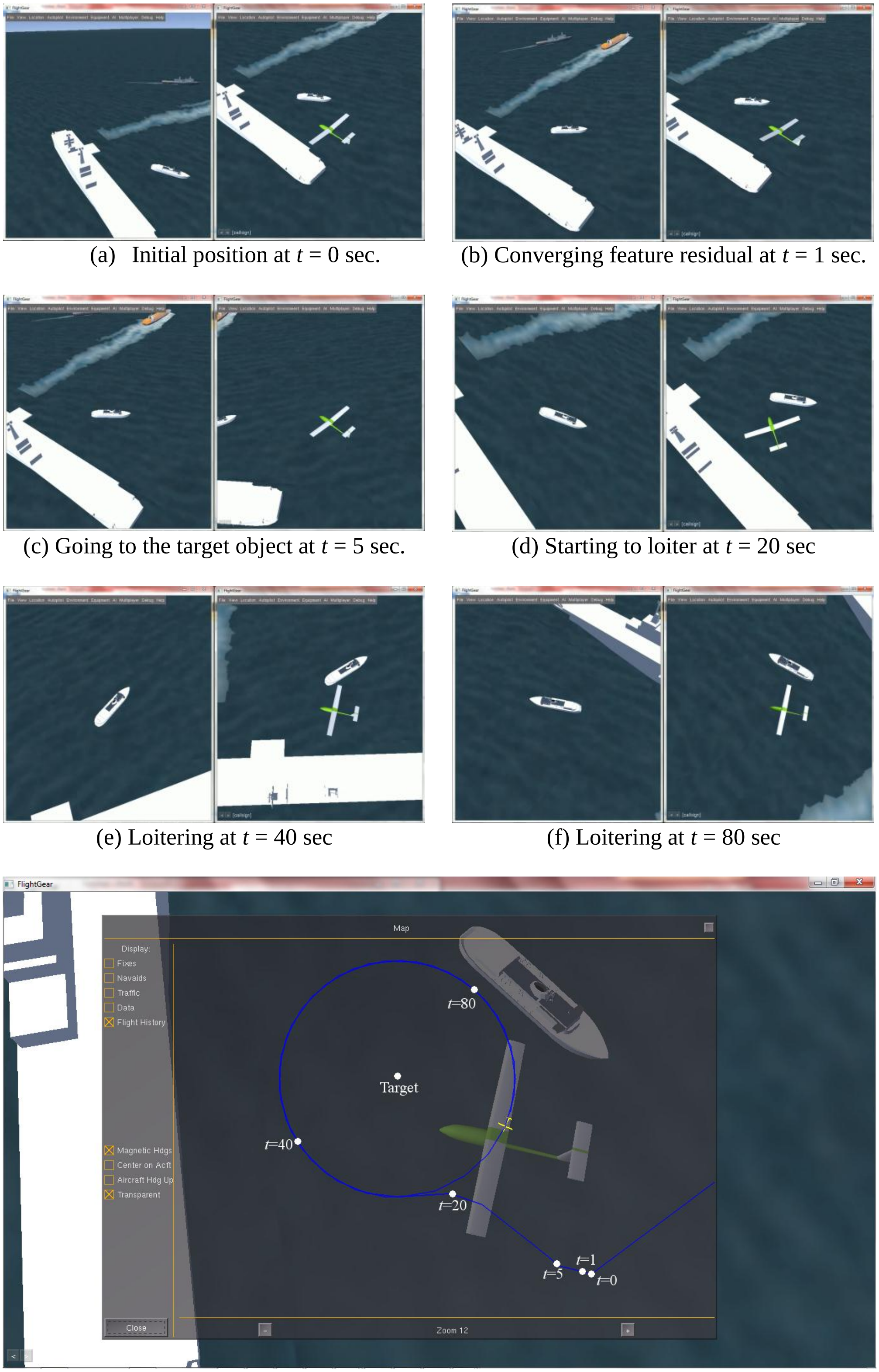
(c) Laboratory experiments.

Figure 9. Development of HILS.

Figure 9(a) shows the HILS configuration; the green blocks denote on-board system; the red blocks denote on-ground system; and the yellow blocks denote simulation system. Figure 9(c) shows the laboratory experiments that are conducted for testing the on-board hardware. Figures 10(a)-(f) shows the HILS results where the left side pictures show the virtual camera view and the right side pictures show the UAV attitude view. It shows that initially the IBVS changes the camera attitude using gimbal mechanism to aim a target object of small white ship as soon as possible. Afterwards, the UAV goes towards the target object and then loiters on it as expected. Figure 10(g) shows the track of the UAV while simulation runs. The simulation results show the good performance and the effectiveness of the proposed IBVS algorithm that is implemented on the real microcontroller hardware.

6. Conclusion

This work presented the development of a visual servoing scheme on a fixed-wing UAV using CFBS control law that adopted the nonlinear models of UAV dynamic and IBVS in an integrated manner. In addition, novel algorithms was introduced to make the longitudinal and lateral-directional maneuvers of fixed-wing UAV safer and smoother. The proposed algorithm was implemented on real hardware and tested using HILS to find out the hardware performances. The numerical simulation and HILS results proved the proposed algorithm and CFBS control law design effectiveness to drive the camera always pointing to the target object and to guide the UAV always chasing and loitering the target object autonomously with safe and smooth flight.



Display:

☐ Fixes
☐ Navaids
☐ Traffic
☐ Data
☒ Flight History

☒ Magnetic Hdgs
☐ Center on Acft
☐ Aircraft Hdg Up
☒ Transparent

(a) Initial position at *t* = 0 sec.

(b) Converging feature residual at *t* = 1 sec.

(c) Going to the target object at *t* = 5 sec.

(d) Starting to loiter at *t* = 20 sec

(e) Loitering at *t* = 40 sec

(f) Loitering at *t* = 80 sec

(g) UAV and object position

Figure 10. HILS results for loitering the stationary target object.

7. References

[1]. Z. Kalal, K. Mikolajczyk, and J. Matras, “Tracking-learning-detection,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 34(7), pp. 1409-1422, 2012.

[2]. M. V. Cook, *Flight Dynamics Principles: A Linear Systems Approach to Aircraft Stability and Control*, 2nd ed., Massachusetts: Elseiver, 2007.

[3]. R. W. Beard and T. W. McLain, *Small Unmanned Aircraft: Theory and Practice*, United Kingdom: Princeton University Press, 2012.

[4]. K. Schmidt, *Modern Flight Dynamics*, New York: McGraw-Hill, 2012.

[5]. F. R. Triputra, B. R. Trilaksono, T. Adiono, R. A. Sasongko, and M. Dahsyat, “Nonlinear dynamic modeling of a fixed-wing unmanned aerial vehicle: A case study of Wulung,” *Journal of Mechatronics, Electrical Power, and Vehicular Technology*, 6(1), pp. 19-30, 2015.

[6]. F. R. Triputra, B. R. Trilaksono, T. Adiono, R. A. Sasongko, “Nonlinear camera gimbal visual servoing using command filtered backstepping,” *Proc. of Int. Conf. on Intelligent Unmanned System*, 2015.

[7]. S. Hutchinson, G. Hager, and P. Cork, “A tutorial on visual servo control,” *IEEE Trans. on Robotics Automation*, 12(5), pp. 651-670, 1996.

[8]. F. Chaumette and S. Hutchinson, “Visual servo control, part I: basic approaches,” *IEEE Robotics and Automation Magazine*, 13(4), pp. 82-90, 2006.

[9]. F. Chaumette and S. Hutchinson, “Visual servo control, part II: advanced approaches,” *IEEE Robotics and Automation Magazine*, 14(1), pp. 109-118, 2007.

[10]. R. Mahony and T. Hamel, “Visual servoing of an under-actuated dynamic rigid-body system: An image-based approach,” *IEEE Trans. on Robotics Automation*, 18(2), pp. 187-198, 2002.

[11]. F. LeBras, T. Hamel, C. Barat, and R. Mahony, “Image-based visual servo controller for circular trajectories for a fixed-wing aircraft,” *Proc. of IEEE Conf. on Decision and Control*, 2009

[12]. S. Mills, N. Aouf, and L. Mejias, “Image based visual servo control for fixed wing UAVs tracking linear infrastructure in wind,” *Proc. of IEEE Int. Conf. on Robotics and Automation*, pp. 5769-5774, 2013.

[13]. P. Peliti, L. Rosa, G. Oriolo, and M. Vendittelli, “Vision-based loitering over a target for a fixed-wing UAV,” *Proc. of IFAC Sym. on Robot Control*, pp. 51-57, 2012.

[14]. N. Regina and M. Zanzi, “UAV guidance law for ground-based target trajectory tracking and loitering,” *Proc. of IEEE Aerospace Conf.*, pp. 1-9, 2011.

[15]. M. Krstic, I. Kanellakopoulos, and P. Kokotovic, *Nonlinear and Adaptive Control Design*, New York: John Willey & Sons, Inc, 1995.

[16]. J. A. Farrell and M. M. Polycarpou, *Adaptive Approximation Based Control*, New York: John Willey & Sons, Inc, 2006.

[17]. J. A. Farrell, M. M. Polycarpou, and M. Sharma, “Command Filtered Backstepping,” *IEEE Trans. on Automatic Control*, 54(6), pp. 1391–1395, 2009.

[18]. R. D. Finck, *USAF Stability and Control DATCOM*, McDonnell Douglas Corp. Wright-Patterson AFB, Ohio, Final Report AFWAL-TR-83-3048, April, 1978.

[19]. R. Eppler and M. Hepperle, “A procedure for propeller design by inverse method,” *Proc. of Int. Conf. on Inverse Design Concepts in Engineering Sciences*, pp. 445-460, 1984.

[20]. P. Corke, *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*, Berlin Heidelberg: Springer Publishing, 2013.

Fadjar Rahino Triputra is a research engineer at technology center for informatics and communication, BPPT, Indonesia. He was graduated from Electronics Engineering Department, Oita University, Japan, in 1994. He obtained his Master Degree from Electronics Engineering Department, Kyushu University, Japan, in 1996. His research interests include avionic systems, electronic and VLSI circuit design, automation system, and control system. From 2010, he involved in the development of avionic systems for unmanned aerial vehicle of BPPT. He is a member of IEEE.

Bambang Riyanto Trilaksono was born in Banyuwangi, Indonesia, on November 15, 1962. He was graduated from Electrical Engineering Department, Institut Teknologi Bandung (ITB), Indonesia, in 1986. He obtained his Master and Doctoral Degree both from Electrical Engineering Department, Waseda University, Japan, in 1991 and 1994, respectively. He is a lecturer at School of Electrical Engineering and Informatics, ITB. His research interests include robust and intelligent control & signal processing, multi-agent systems and robotics. From 1995 until 1998 he involved in the development of total aircraft simulator and in ground flight control test for N250 aircraft at Indonesian Aircraft Industry. He published over 300 papers in journal and conferences, among which 35 papers were published at international journals. He received several awards including Toray Science Foundation Research Award, in 2004. He is currently the Director of Advanced Robotics Research Laboratory, ITB, and the Head of Control and Computer Systems Research Group. He is a member of IEEE (Control Systems Society, Robotics and Automation Society, Signal Processing Society, Computational Intelligence Society, and Computer Society). He is an advisory committee member of Asian Control Association. He was and is servoing as reviewer of a number of journals including IEEE Trans. Automatic Control, International Journal of Control, Automatica, Applied Soft Computing, International Journal of Robust and Nonlinear Control. He served as Vice Chair of Asian Control Conference in 2006, and of International Conference on Intelligent Unmanned Systems in 2007 and 2010. He served as member of Program Committee of a number of conferences in control and intelligent systems. He was former chief editor of ITB Journal of Science, ITB Journal of Engineering Science, and ITB Journal of Information and Communication Technology. He is servoing as editorial board member for International Journal of Electrical Engineering and Informatics, Emerald International Journal of Intelligent Unmanned Systems, Journal of Unmanned Systems Technology, and Internetworking Indonesia Journal. He is editor of a book entitled “Intelligent Unmanned Systems: Theory and Applications”, Studies in Computational Intelligence, Vol. 192, Springer, 2009. He is a research fellow of University of New South Wales, Australia.

Trio Adiono received the B.Eng. degree in electrical engineering and M.Eng. degree in microelectronics from Institut Teknologi Bandung, Indonesia, in 1994 and 1996, respectively. He obtained his Ph.D. degree in VLSI Design from Tokyo Institute of Technology, Japan, in 2002. From 2002 to 2004, he was a research fellow of the Japan Society for the Promotion of Science (JSPS) in Tokyo Institute of Technology. In 2005, he was a visiting scholar at MESA+, Twente University, the Netherlands. In 2006, he was a guest lecturer at Malaysian Institute of Aviation Technology, University of Malaysia. He received the "Second Japan Intellectual Property (IP) Award" in 2000 from Nikkei BP for his research on "Low Bit-rate Video Communication LSI Design". He also holds a Japanese Patent on "High Quality Video Compression System". Currently, he is a lecturer at the School of Electrical Engineering and Informatics, and also serves as a Head of the Microelectronics Center and IC Design Laboratory, Institut Teknologi Bandung. He has co-founded several start-up companies such as Beyond MPEG Inc. (Japan) and Versatile Silicon Tech (Indonesia). He had several key committee positions and keynote speaker in national and international conferences, workshops, and training in microelectronics area. He currently serves as a chair of the IEEE SSCS Indonesia Chapter. His research interests include VLSI design, signal and image processing, smart card, electronics solution design and integration.

Rianto Adhy Sasongko was born in Kudus, Indonesia, on February 15, 1974. He was graduated from Aeronautics and Astronautics Department, Institut Teknologi Bandung (ITB), Indonesia, in 1997. He obtained his Master degree from Automatic Control and Systems Engineering Department, The University of Sheffield, United Kingdom, in 2000. He received his Doctoral degree from Electrical and Electronic Engineering Department, Imperial College, United Kingdom, in 2008. He is a lecturer at Faculty of Mechanical and Aerospace Engineering, ITB. His research interests include control and simulation of aerospace systems, flight dynamics, autonomous algorithm for UAV, aeroelasticity and fluid induced vibration on flexible structure, structure dynamic and vibration analysis, and flight dynamics modelling of flapping wing system.