

Computer vision-based control of an autonomous blimp

Korkut BEKİROĞLU^{1,*}, Mario SZNAIER², Constantino LAGO¹,
Bahram SHAFAI²

¹Department of Electrical Engineering, Pennsylvania State University, University Park, PA, USA

²Department of Electrical and Computer Engineering, Northeastern University, Boston, MA, USA

Received: 22.01.2015

Accepted/Published Online: 02.07.2015

Final Version: 20.06.2016

Abstract: The objective of this study is twofold: to approximate a model of a blimp, and to use this model to develop a setup to track a target with the blimp that is outfitted with a wireless camera and radio-controlled propellers. This article presents a powerful method to track any moving or stationary target with an unmanned aerial vehicle by combining the advantages of the proportional derivative (PD) controller, continuously adaptive mean shift (Camshift) algorithm, and pulse width modulation (PWM) method. As a result, it is demonstrated that a decent approximation of blimp behavior is sufficient when using a mathematical model that contains saturation in velocity and actuation. Additionally, a code for the proposed algorithm is developed to capture every frame sample with a frame grabber as a sensor in real time. Once the previously chosen object is tracked, the coordinate data (location information) are transferred to the controller to apply required pulses to DC motors on the blimp. In this paper, the proposed controller is outlined in two steps. Initially, one calculates a PD controller that fulfills the specifications of the mathematical model without saturation. Secondly, PWM is utilized to address the impact of nonlinearities.

Key words: Unmanned air vehicle, PD controller, computer vision (Camshift), pulse width modulation, vision-based control

1. Introduction

A small unmanned air vehicle (UAV), equipped with radio-controlled propellers and a wireless camera, is designed to track stationary or moving objects. Initially, an approximate model of this framework is created and input/output relationships are utilized to validate this model. Once the model is available, the linear (PD) controller and a pulse width modulation (PWM) algorithm are effectively used to overcome saturation in the estimated model, the massive inertia of the UAV, and the speed control of ON/OFF motors.

Various UAV applications are available for military as well as regular civilian uses. Since most UAVs utilize low altitude and low speed for surveillance and observation, they might be utilized for ground activity control, watching and foreseeing rush hour traffic, and taking precautionary measures to ease accidental circumstances. In order to attain a high level of safety and detect a specific object, the elevation of UAVs can be balanced automatically by utilizing a visual sensor, and, for this purpose, this study employs a wireless camera. The UAVs should automatically perform their previously defined duties. This navigation process closely resembles how pilots control aircraft. However, remotely piloted UAVs are extremely hard to control due to their massive inertia and substantial structure. Due to their challenging dynamics, one specific control action may not be sufficient to control it.

*Correspondence: korkut.bekiroglu@gmail.com

The structure of the aircraft equipment is critical for robustness, because it specifically influences the system dynamics. In [1], the ideal aircraft hardware is chosen to have maximum performance at minimum cost. In [2,3], an autonomous blimp is created for an inspection system so that it could drift around a particular target. An alternate visual blimp tracking application was developed in [4,5] in order to gauge the blimp position. In these settings, the utilization of the fixed camera on the floor limits the mobility of the airship and subsequently influences the stability of the system. In our setup, the camera is seated on the gondola of the UAV, thus expanding the mobility of the blimp.

Some initial results and calculations of this study were presented in [6]. Differently than [6], this article exhibits more information about the algorithm, hardware (circuits, camera, etc.) design, calculations, and results. Each and every application in the literature requires a different ability to accomplish its mission, and thus different algorithms are available in the literature. For our case, which is an indoor application, a tracker algorithm and a controller for the UAV are developed by using Camshift and a linear controller, which together demonstrate an alternative method to the previously stated techniques. Furthermore, an input/output model of an UAV is approximated in this study. Finally, different than the previous works, a PWM algorithm is created to handle the nonlinearities in the system due to the ON/OFF nature of DC motors.

The system structure is briefly explained in Section 2, including hardware design. In Section 3, the impact of the aerial force on dynamics is briefly clarified and a suitable UAV behavior model is given. Section 4 explains how a graphical user interface (GUI) is developed to manually and/or automatically drive UAV. Finally, the experimental results and the conclusion are presented in Sections 5 and 6, respectively.

2. System structure and hardware design

During the mechanical design of an UAV, different issues are encountered. As a first problem in any blimp framework, the larger the envelope volume (the balloon containing the lifting gas) that carries the required total weight on the gondola of the UAV, the higher the force required to drive the UAV. Therefore, the envelope size needs to be big enough to let us convey the fundamental hardware. In any case, a blimp utilized indoors needs to be sufficiently small. The envelope size exhibits difficulty in terms of robust tracking and control. The literature provides more information about the envelope size, batteries, and camera of this kind of setup [1,7].

Including the physical structure of the blimp, the system mainly has two wireless connections, which have an effect on robustness. The first one is between the video camera and its receiver (frame grabber), and the second is between the DC motors and their remote control. These wireless paths need to have enough ranges. If the video camera connection has any issues, it straightforwardly influences the Camshift tracker because of the missing and/or distorted frame. It subsequently influences the controller algorithm and results in a loss of system robustness. Moreover, if the second path has issues, the motors cannot obtain control actions from the controller, which also straightforwardly influences the control algorithm. Figure 1 and the following list portray the general system structure.

- First, the camera gathers live images and transmits them to the receiver.
- The captured image data are obtained by the frame grabber.
- Once the code obtains the image data, the Camshift algorithm uses these data and gives the current location information of the object on the frame. The tracker algorithm output and the reference input (the center of the frame) are compared and an error signal is generated. Then this error signal is delivered

to the controller. Next, the controller output is sent to the PWM algorithm that dynamically changes the time length of the ON and OFF cycles of the DC motors.

- Once the transmitter circuit acquires the information from the PWM, it transmits the pulses to the motor receiver. Then the DC motors drive the UAV according to the indicated direction at a modulated speed.
- The camera collects another image frame and the same loop is performed.

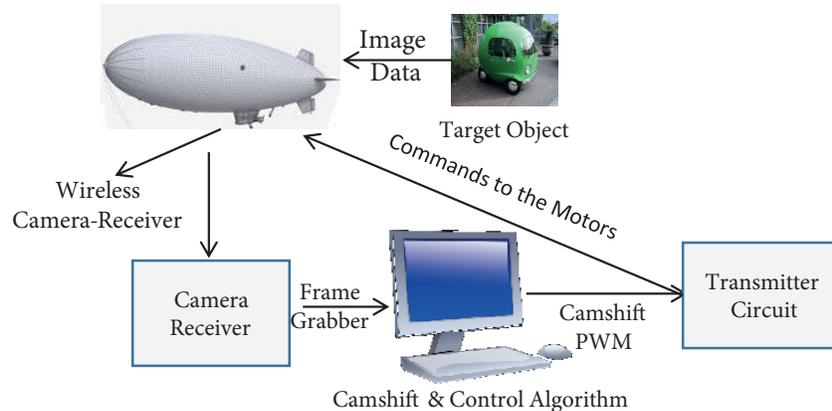


Figure 1. General algorithm.

The hardware in this setup needs to receive, interpret, and evaluate the available video data as well as drive the UAV. The primary system hardware consists of radio-controlled propellers with transmitter–receiver circuits on the gondola, a computer, a wireless video camera, two DC motors (move blimp right/left, forward/backward), and the blimp itself. A remote controller is linked to the computer’s parallel port and utilized to transmit control commands to the propellers.

3. Problem statement

3.1. Modeling

Since the payload of the UAV is restricted and the envelope volume is quite large, the controller configuration and the modeling of the UAV are fairly difficult. Some of the issues of controlling the UAV are its inertia, shape, unstable dynamics relevant to aerodynamics, and erratic disturbances. Accordingly, PWM and the linear controller algorithms should be sufficiently robust to overcome these disturbances and unpredictable dynamics [7].

Particularly, the Newtonian laws of motion are commonly utilized to depict the mathematical model of UAVs. The motion equations contain drag, thrust/buoyancy, and gravitational forces on the UAV. The drag force is experienced by an item due to motion through the air. The drag force equation is computed by using the general aerodynamic drag equation. The gravitational forces are also adjusted by the buoyancy forces that are created by the lighter-than-air vehicle itself. The UAV does not require energy to keep the altitude of the blimp at a definite level, and thus one does not need to tackle the gravitational forces and buoyancy forces [7] (assume the blimp is balanced vertically), since one just deals with controlling the UAV horizontally. The density of fluid ρ is proportional to the drag force that is presumed to be quite small for this setup. The drag force is proportional to A , the cross-sectional area in the motion direction. Furthermore, other factors influence

the drag force $F_{drag} = \frac{1}{2} \rho ACv^2$, such as viscosity, shape, compressibility, texture, boundary layer separation, and lift, which are expressed by drag coefficient C .

It is clear that we have limitations in actuation and, subsequently, one cannot compensate the substantial drag force. This force is relatively proportional to the velocity square. This implies that there is a greater velocity that might be attained by the UAV. Given the huge volume and low mass of the UAV, the velocity limit is hit ‘rapidly’ and, subsequently, a saturation nonlinearity in velocity gives a decent rough guess of UAV behavior. Correspondingly, the velocity saturation is added to the approximate model with actuator saturation. Additionally, the drag coefficient C and weight ρ are thought to be low. Therefore, these coefficients have no significant impact on the UAV dynamics. However, despite the fact that these coefficients have little impact on the UAV dynamics, they marginally change the upper and lower cutoff points of velocity saturation.

The actuation is thrust for the UAV. Moreover, $F = m\ddot{x}$ is the most general equation for Newtonian systems. Although it has moderately little mass, because of its volume, its inertia is huge compared to the force generated by the propellers. Therefore, the UAV transfer function can be approximated as $F = m\ddot{x}$. However, when we are characterizing the differential equation, we have to remember that this is just the effective piece of the UAV dynamics. The effective piece of UAV dynamics originates from the mass, although the general system transfer function also has dynamics from the camera, DC motors, and other tools. Along these lines, the important part of the dynamics is characterized as $\frac{X(s)}{F(s)} = \frac{1}{s^2}$. This linear model gives a decent estimate for a part of the scope of the blimp operation. Finally, these dynamics are utilized while outlining the linear controller. As a result, in addition to actuator saturation, which is general in these kinds of systems, a velocity saturation block should be included between the two integrators. The cutoff of the velocity saturation block can change according to the changing aerodynamic force. The result in Figure 2 summarizes the system dynamics approximately.

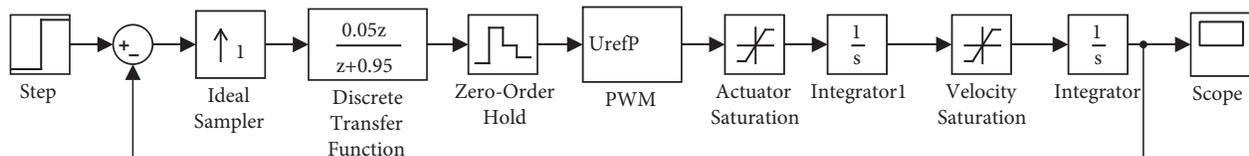


Figure 2. Controller implementation.

The blimp output data plotted in Figure 3 might be decently estimated with a second-order polynomial (blue line) in the time interval of $\{3-11\}$ s and with a linear function (yellow line) after 11 s. The data are approximated with a linear function after 11 s, since the velocity state hits the cutoff of velocity saturation. In another words, we can say that the blimp velocity in the time interval of 0 to 11 is in linear region of saturation. This is exceptionally critical information with the controller sampling rate while calculating the average power that is controlled by the PWM block in Figure 2.

3.2. Problem formulation

In the previous modeling section, it was demonstrated that the approximate model $(1/s^2)$ in a certain time interval dominates the system dynamics, which relates to the linear region of displacement in Figure 3. The system acts nonlinearly on the outside of this linear region or time interval because of the saturations caused by the actuator and drag force. Various methods are attempted to overcome the nonlinearities, yet the velocity saturation block confuses the controller design procedure. Therefore, a PWM block is utilized to control the speed of the propellers. If the propellers generate a reasonably small thrust to keep the displacement of the

UAV in the linear region of the actuator and velocity saturation blocks, we could use a linear controller design technique. We know that actuators work in just one of two states, such as full thrust forwards and backwards. Therefore, in order to apply the controller output to the actuators, the output information of the controller needs to be postprepared with a PWM block to estimate the overall impact of speed-controllable DC motors. Note that this PWM method makes sense, since we have an integrator immediately after the controller in which the impact smooths out the progressions in PWM to produce a continuous speed change to the overall UAV system. Consequently, the blimp is steady with a linear controller and moves slowly towards the target. The structure of the closed loop system is demonstrated in Figure 2. The whole closed loop system, including its continuous approximate model, discrete controller, saturation blocks, sampling element (zero order hold, ZOH), and PWM block are shown in Figure 2.

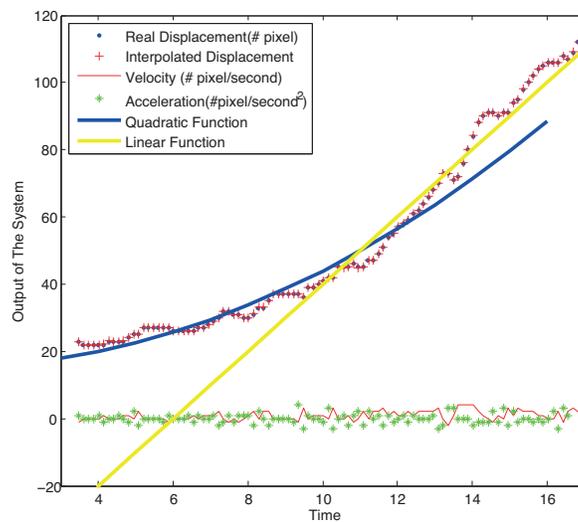


Figure 3. Output of the system-acceleration-velocity approximations.

4. Tracker and controller design

4.1. Controller design strategy

It is critical to indicate that the saturation blocks in actuators and between the two states render the UAV system nonlinear. In the literature, we have different controller design techniques for these types of nonlinear systems [8–12]. On the other hand, these works demonstrate feedback controller design methods once the system has input, actuator, or sensor saturation. Regrettably, our UAV system has none of these sorts of nonlinearity structures. Therefore, the previously stated methods are not specifically applicable to our system, because the UAV has state saturation due to the physics of the UAV structure. Additionally, the states of the approximate model in this framework are not accessible. Since we cannot use these methods, the displacement of the UAV in Figure 3 is kept in a linear region of state saturation, and then linear controller design techniques are utilized to control the system.

In Eq. (1), we give the approximate model with ZOH and the structure of the proposed controller. Any kind of linear digital controller could be utilized for the double integrator (approximate model) with a sampling rate of $T = 0.6$ s in Eq. (1) to track the reference input and stabilize the system. Using the fundamental

linear control system methods, a type-two stable system can track ramp and step inputs with zero steady-state error. Therefore, any linear stabilizing controller could achieve the tracking goal. Clearly, the proportional (P) controller cannot stabilize the UAV system. Furthermore, attempting a proportional integrator (PI) controller would not influence the system, since it increases the system type yet does not add any extra control action. Another two basic types of linear controllers are the proportional derivative (PD) and proportional integrator derivative (PID) controllers. The PID that brings an additional integrator to the overall system could stabilize the UAV system; however, it corrupts the tracker performance by causing unnecessary overshoot. Note that, since the field of view (FOV) of the on-board camera is restricted, excessive overshoot could result in losing the targeted object. Therefore, the blimp should be moved slowly in order to overcome the problems related to the camera's FOV size and the saturation nonlinearity. However, the speed of the UAV should be faster than the moving objects in order to consistently track them. Finally, the PD controller is preferred to obtain zero steady-state error and limited overshoot. A PD controller rather than a PID is considered for this setup.

$$\begin{aligned} G(z) &= ZOH \frac{1}{s^2} = \frac{z-1}{z} Z^{-1} \left(\frac{1}{s^3} \right) = \frac{z-1}{z} \left(\frac{T^2 z(z+1)}{(z-1)^3} \right) = \left(\frac{T^2 z(z+1)}{2(z-1)^2} \right) = \frac{0.18z-0.18}{(z-1)^2} \\ D(z) &= K \left(\frac{K_d(z-1)}{Tz} + K_p \right) \end{aligned} \quad (1)$$

We use bilinear transformation to test the controller and an open loop transfer function and the characteristic equation of Eq. (2) with the Routh–Hurwitz test. Moreover, we should note that the sampling rate is 0.6 s, and it is not perfectly constant in reality due to the changing code running speeds. A zero is also calculated that is close to the two integrator roots, in order to achieve a stable system or to form a root locus curve of the transfer function to keep it inside the stable region/unit circle. To obtain a stable system, $K_p = 0.2$ is chosen.

$$\begin{aligned} 1 + D(z)G(z) &= 1 + \frac{K_d z - K_d + K_p T z}{Tz} \frac{T^2(z+1)}{2(z-1)^2} = 1 + \frac{K_d \frac{T+2w}{T-2w} - K_d + K_p T \frac{T+2w}{T-2w}}{T \frac{T+2w}{T-2w}} \left(\frac{T^2 \left(\frac{T+2w}{T-2w} + 1 \right)}{2 \left(\frac{T+2w}{T-2w} - 1 \right)^2} \right) \\ &= 16w^2 + 32w^3 + T^5 K_p + 4T^3 K_d w - 4T^3 K_p w^2 - 8T^2 K_d w^2 \end{aligned} \quad (2)$$

Once the characteristic equation is calculated by setting $K_p = 0.2$ and $T = 0.6$, the Routh–Hurwitz array is utilized to analyze the characteristic equation to choose a suitable K_d range. Therefore, according to the Routh–Hurwitz test, K_d should be chosen as $K_d \leq 3.263$ to guarantee the stability of system. If it were decided that K_d would make the roots of the transfer function close to the unit circle, this would create extra oscillation that is not preferable for the UAV. Likewise, it cannot be greater than $K_d \leq 0.00623$, and, as a result, $K_d = 1$ is chosen in Eq. (3).

$$1 + D(z)G(z) = 1 + \frac{K_d z - K_d + K_p T z}{Tz} \frac{T^2(z+1)}{2(z-1)^2} = 2z^3 - 4z^2 + 2z + TK_d z^2 - TK_d + K_p T^2 z^2 + K_p T^2 z \quad (3)$$

We give the characteristic equation of the closed loop system in Eq. (3). Any K_p values fulfilling the condition $K_p > 0$ could be chosen. Henceforth, $K_p = 0.2$ is chosen by dealing with the physical limitations of the actuators. In addition, the choice of K_d value relies on the T sampling rate, because TK_d should be $TK_d \leq 2$, and we choose $K = 0.6$.

Finally, in order to devise necessary performance specifications, the gain K value is balanced. Once the controller is added to the overall feedback system, the root locus of the controlled system is checked to choose a reasonable K . Thus, the roots of the transfer function are moved on a real axis between 1 and 0. We can also replace them far from the unit circle to diminish the overshoot.

We need to address certain issues associated with PWM with the controller parameters in Eq. (4). Once the controller has the current and previous error values, the differences between these two values are calculated. As soon as this information is known, the PWM block evaluates the controller output and calculates the average power to apply the propellers. On the other hand, the DC motors cannot respond if the PWM forces the motors under 80 ms (experimentally obtained) of duty cycle. As a result of the respond time problem of the DC motors, Kd and Kp values are experimentally tuned to have essential reaction from the DC motors. Moreover, the envelope of the UAV is too large for the propellers. In this manner, the DC motors need to deliver some extra power to overcome the oversized envelope problem. Consequently, the proper gain values of the controller are tuned as Kd = 3 and Kp = 2, while K = 1 to accomplish a good balance between response time (assessing the oversized blimp) and saturation constraints leading to the values Kp = 2 and Kd = 3. The resulting controller is presented as: $D(z) = Z^{-1}(7 - 5z^{-1})$ and $D(n) = 7e[n] - 5e[n - 1]$.

4.2. Tracker algorithm

The Camshift algorithm is one of the tracking algorithms in the computer vision literature. This algorithm is preferred for the colored videos in computer vision applications. The continuously adaptive mean-shift algorithm (Camshift) is a modified version of the Meanshift algorithm, built by Intel [13]. As long as various targets in the image frame do not have the same color-based histogram and the image data from the video camera are not too noisy, the Camshift tracker algorithm performs adequately for the UAV system to track and follow the target in the image frame.

4.3. PWM design

The reference input frame and blue frame are presented in Figures 4 and 5. According to the reference frame, the current position information of the target on the image (Camshift output) is subtracted to calculate the error signal for the controller. Once this error signal value is calculated, the control algorithm decides the driving direction of the UAV. Then, according to this decision, the UAV is thrust forward-backward or right-left by the propellers. For example, let us assume that the target is located close to the center of the X axis on the image frame and far from the Y axis. Since the target is not located at the center of the image, the UAV needs to be thrust left or right to bring the target object into the center or to centralize the Y axis. The control algorithm mainly picks one of the motion directions (right-left-forward-backward) and gives average power information about the PWM block.

Once the motion direction of the UAV is chosen, the current error signal value at time n and previous error signal value at time $n - 1$ is evaluated by the controller of Eq. (7). The difference (controller output) is critical information for deciding the percentage level or average power to drive the DC motors. As a result, the PWM algorithm decides various parameters, such as how much trust (average power) is needed according to the current position of the target.

Furthermore, one larger frame is experimentally drawn over the reference (input) frame. The limits (X and Y axis values) of this larger frame over the reference input are represented in the Table and Figure 4. The UAV has to be moved slowly according to its closeness to the reference frame in order to overcome huge inertia. The inertia of the UAV is utilized to move it, and the power of the propellers is decreased until it tracks the reference input. Finally, PWM adjusts the average power of the propellers according to the closeness of the target to the reference frame.

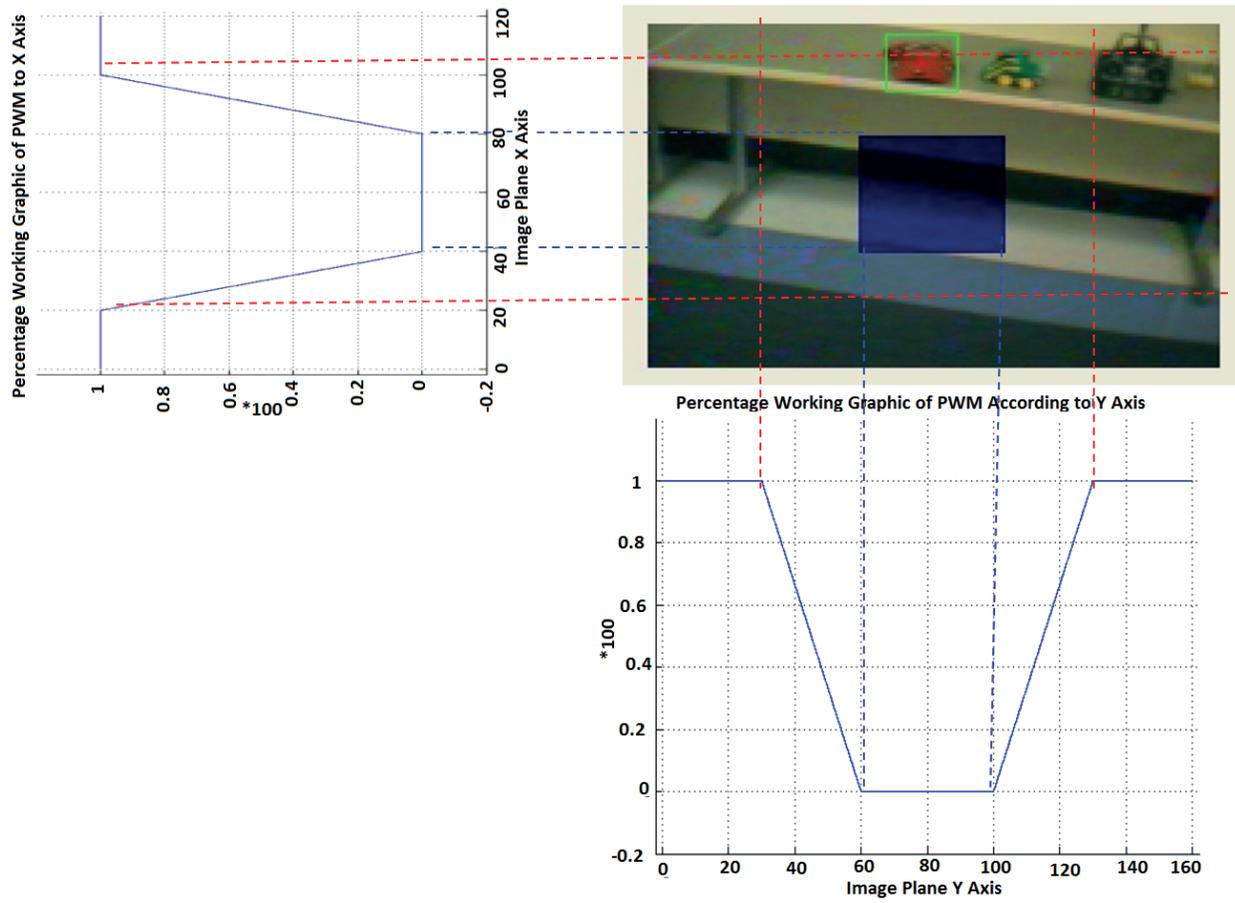


Figure 4. Percentage level of PWM.

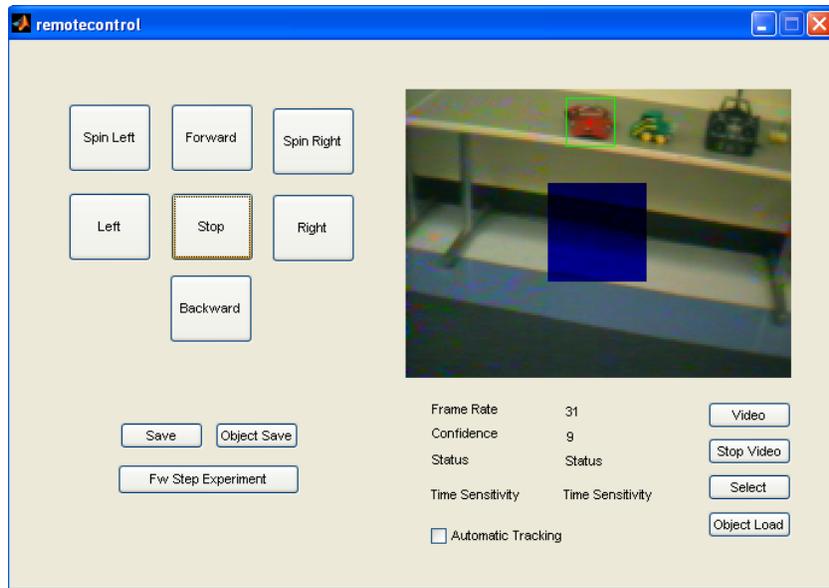


Figure 5. GUI of the blimp control software.

Table. Percentage level of PWM.

Forward–backward	P. PWM $\times 100$	Right–left	P. PWM $\times 100$
$X < 20$	1	$Y < 30$	1
$20 \leq X < 40$	$Cont.out/maxpwm2$	$30 \leq Y < 60$	$Cont.out/maxpwm$
$40 \leq X < 80$	0	$60 \leq Y < 100$	0
$80 \leq X < 100$	$Cont.out/maxpwm2$	$100 \leq Y < 130$	$Cont.out/maxpwm$
$100 \leq X < 120$	1	$130 \leq Y < 160$	1

4.4. Implementation

Once the hardware of the system is designed, we can combine the hardware of the UAV, linear controller, image tracker (Camshift), and PWM to develop a graphical user interface (GUI) in Figure 5 for the user to drive the blimp automatically or manually.

The goal of this setup is to position the UAV so that the target object, which is initially located in an unknown position on the video, is centered on the video. Therefore, the reference input that is chosen as a small frame in the center of the FOV of the camera needs to be chosen to close the loop with visual feedback (camera) and the previously calculated linear controller. Because of the presence of hardware limitations and external disturbances, including a large volume envelope relative to the propellers' thrust (resulting in substantial drag), motor time constants, and image processing time, it is essentially impractical to keep the tracked object at the exact center of the image (a single point at the center) using this setup. In this way, the performance specifications are relaxed such that the target object can be confined to a small region that is highlighted in Figure 4. The motors drive the propellers to have a rough linear response, as long as the velocity of the UAV does not exceed the saturation cutoff explained in Section 3.1. For this impact, each sampling period is broken down into shorter subperiods. Only some of these subperiods are subsequently delivered to power the motors.

5. Experiment

Any object in a room is focused on to be tracked. Then the power signal on the propellers and the location information of the object in the frame are saved and presented in this section. The delivered power cycles for the propellers to the right–left and the forward–backward forces and the output of PWM are depicted in Figures 6 and 7. The coordinates of the tracked moving object on the image frame are depicted in Figure 8. In Figure 8, the red box shows the reference coordinate/input where the object is kept in this region. It is demonstrated in the plots that the proposed controller effectively moves the blimp so that the image of the moving target is kept in a small region in the center of the image plane in the presence of disturbances (as an indoor application, a disturbance might be the air circulation in the room). Although the blimp has huge volume and inertia, the proposed controller and PWM could regulate the DC motors to keep the moving target in the center of the image frame. For this paper, a tracker algorithm and a controller for the UAV are successfully built up by using Camshift and a linear controller, which together illustrate a different method than in the literature.

6. Conclusion

This study exhibits a robust approach to track any stationary and/or moving item with an UAV with extremely constrained actuation capacity (low thrust ON/OFF DC motors) by utilizing a combination of a PD (linear) controller, Camshift (image tracker algorithm), and PWM to partially overcome the ON/OFF nature and saturation nonlinearities. To achieve this task, a rough model is approximated and experimentally validated for

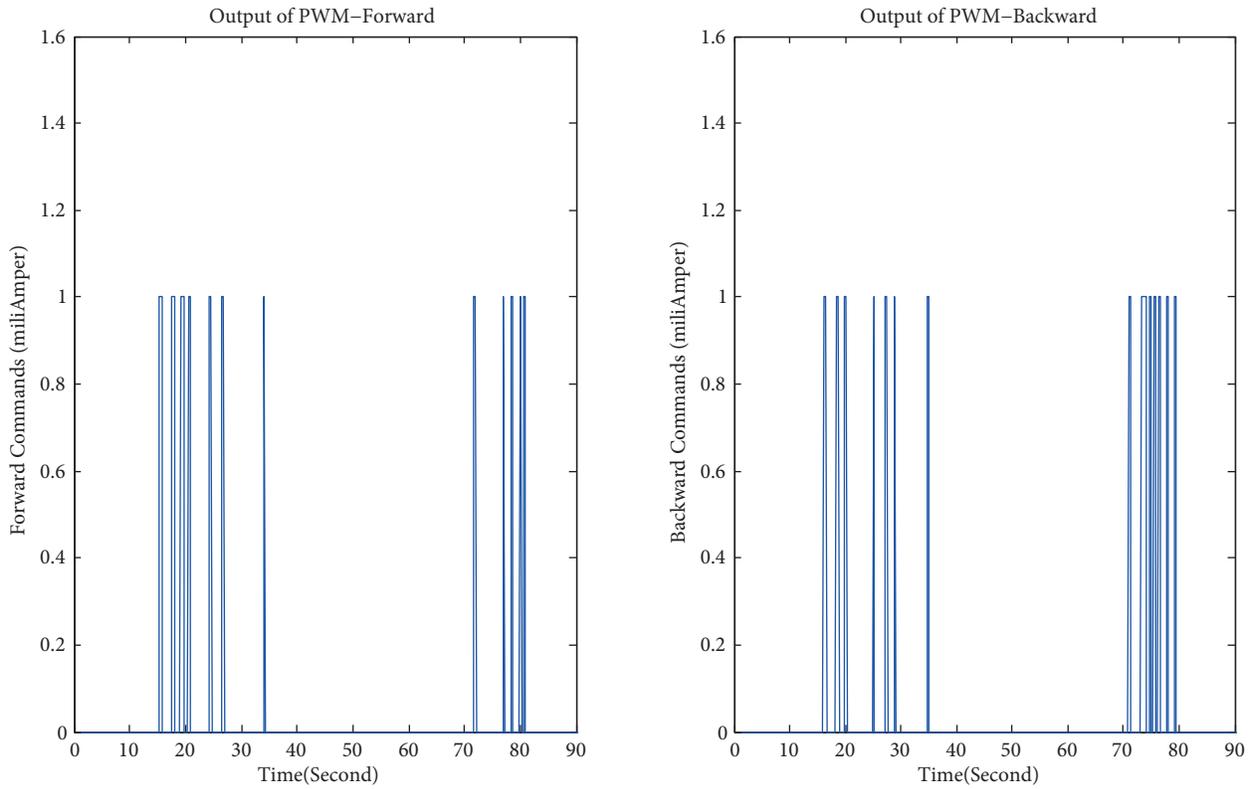


Figure 6. Forward and backward commands.

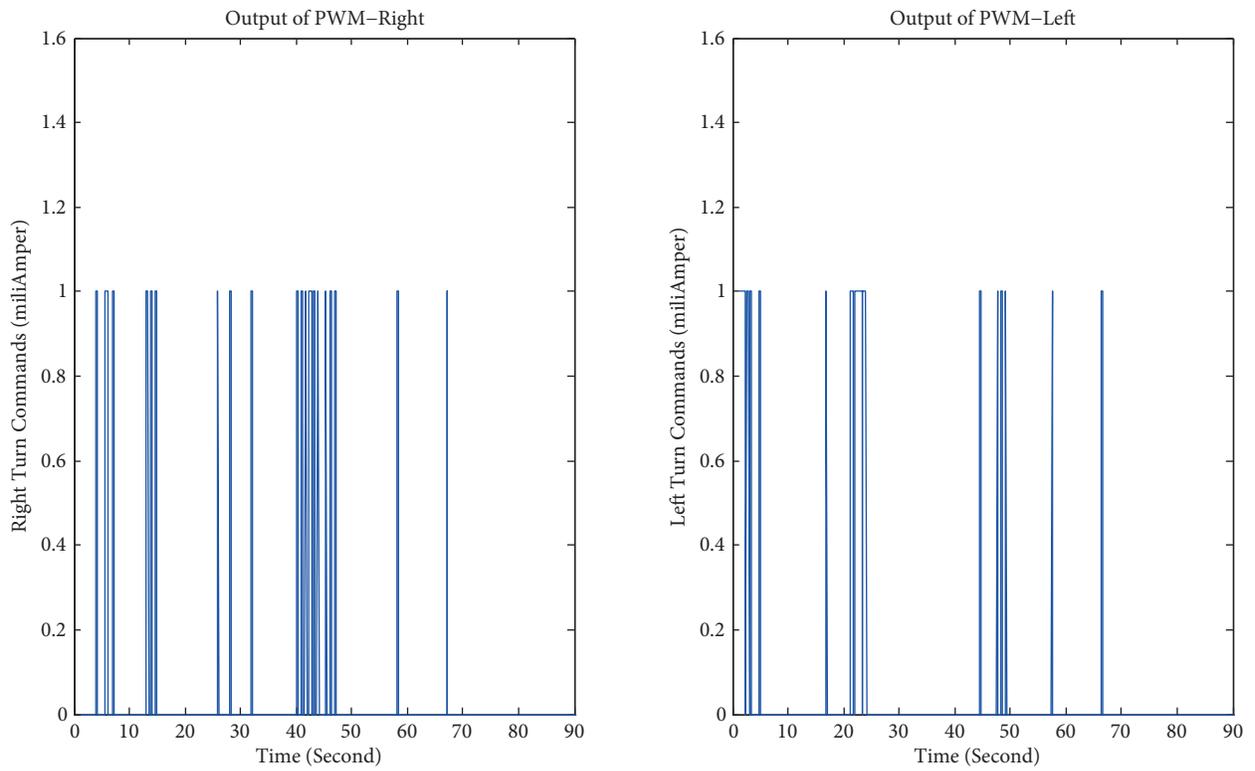


Figure 7. Right and left commands.

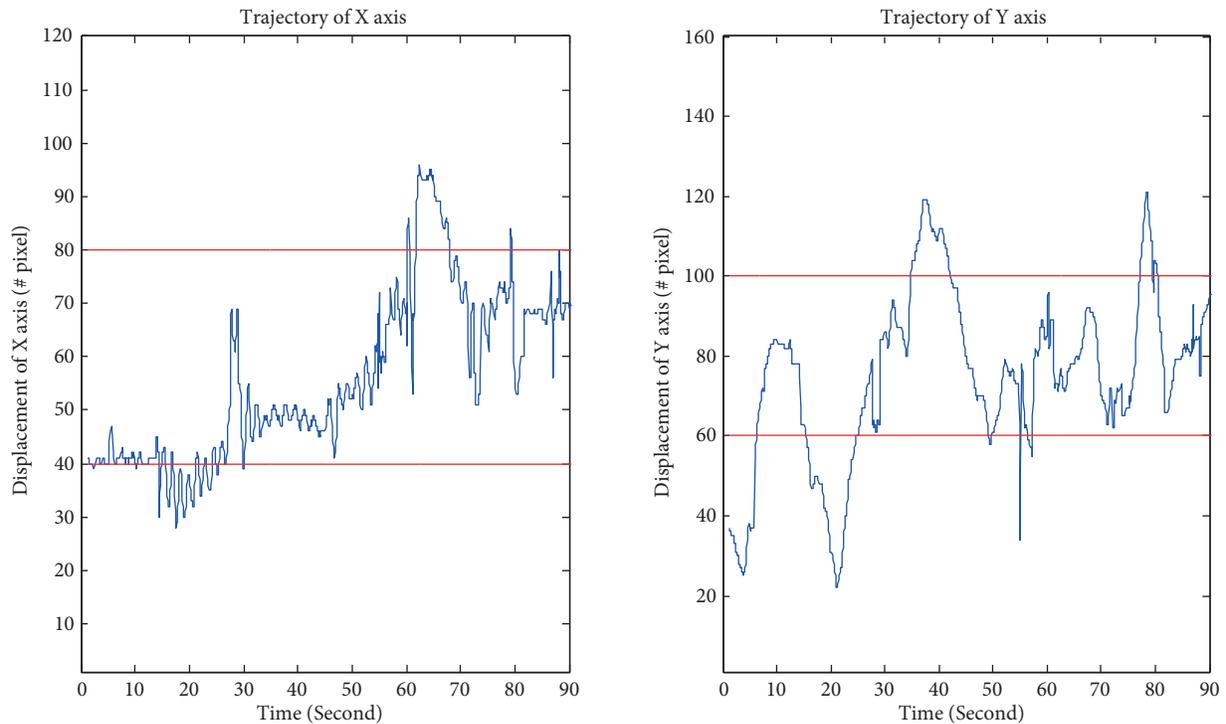


Figure 8. X and Y coordinates of target on image.

the UAV, where drag impacts are considered via saturation. Then this model is utilized to plan a conventional PD controller while ignoring the ON/OFF nature of the motors. Finally, PWM is added to the system to approximately implement the PD linear controller despite the ON/OFF nature of the actuators. The adequacy of this method is exhibited with several experiments. In these experiments we present the capacity of the closed loop system, which could keep the tracked object in a given region of the image (reference frame) in the presence of substantial drag forces, disturbances, and the inertia of the platform. In future research, we will expand these methods to other platforms such as quad-rotors.

References

- [1] Sanz R, González P, Burgard W, Fernández JL. Developing a low-cost autonomous indoor blimp. *Journal of Physical Agents* 2009; 3: 43-53.
- [2] Fukao T, Fujitani K, Kanade T. An autonomous blimp for a surveillance system. In: *IEEE/RSJ 2003 International Conference on Intelligent Robots and Systems*; 27–31 October 2003; Las Vegas, NV, USA. New York, NY, USA: IEEE. pp. 1820-1825.
- [3] Fukao T, Kanzawa T, Osuka K. Tracking control of an aerial blimp robot based on image information. In: *IEEE 2007 International Conference on Control Applications*; 1–3 October 2007; Singapore. New York, NY, USA: IEEE. pp. 874-879.
- [4] Oh S, Kang S, Lee K, Ahn S, Kim E. Flying display: autonomous blimp with real-time visual tracking and image projection. In: *IEEE/RSJ 2006 International Conference on Intelligent Robots and Systems*; 9–15 October 2006; Beijing, China. New York, NY, USA: IEEE. pp. 131-136.
- [5] Kawai Y, Hirano H, Azuma T, Fujita M. Visual feedback control of an unmanned planar blimp system with self-scheduling parameter via receding horizon approach. In: *IEEE 2004 International Conference on Control Applications*; 2–4 September 2004; Taipei, Taiwan. New York, NY, USA: IEEE. pp. 1603-1608.

- [6] Bekiroglu K, Sznaier M, Lagoa C, Shafai B. Vision based control of an autonomous blimp with actuator saturation using pulse-width modulation. In: IEEE 2013 International Conference on Control Applications; 28–30 August 2013; Hyderabad, India. New York, NY, USA: IEEE. pp. 1036-1041.
- [7] Nakatsu R, Hoshino J. Entertainment Computing: Technologies and Applications. New York, NY, USA: Kluwer Academic Publishers, 2003.
- [8] Tarbouriech, S, Valmorbidia G, Garcia G, Biannic J. Stability and performance analysis for linear systems with actuator and sensor saturations subject to un-modeled dynamics. In: IEEE 2008 American Control Conference; 11–13 June 2008; Seattle, WA, USA. New York, NY, USA: IEEE. pp. 401-406.
- [9] Tarbouriech S, Garcia G, Langouet P. Controller design for linear unstable systems with position and rate actuator saturation. In: IEEE 2002 International Conference on Control Applications; 18–20 September 2002; Glasgow, UK. New York, NY, USA: IEEE. pp. 874-879.
- [10] Tingshu H, Teel AR, Zaccarian, L. Performance analysis of saturated systems via two forms of differential inclusions. In: IEEE 2005 Conference on Decision and Control and European Control; 12–15 December 2005; Seville, Spain. New York, NY, USA: IEEE. pp. 8100-8105.
- [11] Valluri S, Kapila V. Geometric control of input saturated systems with guaranteed closed-loop performance and stability. In: IEEE 1999 American Control Conference; 2–4 June 1999; San Diego, CA, USA. New York, NY, USA: IEEE. pp. 2486-2492.
- [12] Tingshu H, Zongli L, Chen BM. Analysis and design for discrete-time linear systems subject to actuator saturation. In: IEEE 2001 Conference on Decision and Control; 4–7 December 2001; Orlando, FL, USA. New York, NY, USA: IEEE. pp. 4675-4680.
- [13] Gary RB. Real time face and object tracking as a component of a perceptual user interface. In: IEEE 1998 Workshop on Applications of Computer Vision; 19–21 October 1998; Princeton, NJ, USA. New York, NY, USA: IEEE. pp. 214-219.