AMERICAN RESEARCH JOURNALS
An Academic Publishing House

Research Article   **Open Access**

Gimballed Camera Control for On-Point Target Tracking

Indhu B[1]*, VPS Naidu[2]

[1]*Dept. of Inst. & Cont. Engg., NIT, Trichy, India.
[2]MSDF Lab, CSIR-NAL, Bangalore, India.
vpsnaidu@gmail.com

Abstract: Detailed design of tracking system that enable a Micro Aerial Vehicle (MAV), equipped with a pan-tilt gimbaled camera to be stabilized and to track the target, with target asin center of image. The algorithm performs automatic tracking of target while camera gimbal is moved according to the target position in image. 2- Axis gimbal camera set up is made with servo motors and FPV 10x zoom camera. Gimbal camera set up is integrated with APM and XBee to replicate real set up of MAV in laboratory. And it is controlled with GCS set up made of joystick, XBee, Matlab and Mission Planner software. Thus, closed loop for control of gimbal camera designed. Image processing algorithm is used to track the target and to compute the position of target in image. Accordingly, control algorithm will generate control PWM signal for control of camera gimbal.

INTRODUCTION

The past decade has outstanding growth in the use of unmanned air vehicles (UAVs). Even though large UAV are capable of executing complex mission its availability is limited and costly.Thus development of low cost, small UAV (i.e Miniature Aerial Vehicle-MAV) is increased [1, 2]. MAV are used for military and civil application.One of the most significant limitations to small UAV ISR (Intelligent Surveillance and Reconnaissance) systems is their inability to carry a stabilized gimbal capable of delivering the stabilization performance required for high target resolution while the platform stays outside of its detection footprint [3].

In order to have stable gimbal, the target should be locked to the camera. Target locking is the act of maintaining the target in the sensor's center field of view, under target motion. In order to achieve it, target should be tracked continuously and to have the object in center of image, the camera is moved in Pan and tilt direction according to target position in image.

SYSTEM ARCHITECTURE

Control loop for control of gimbaled camera is shown in Fig-1. In typical operational scenario, when the MAV is in autonomous flight, the system operator may select a target of interest on the screen. Once a target is identified by the operator, the image processing tracking algorithm computes the target tracking box and provide the position of the centroids of this box in every image frame of continuous video from the camera [4]. From the object centroid detected by the algorithm, the required pixel shift to bring the target to center of image is computed. The computed pixel difference is given as the input to an integrated MAV-gimbal control algorithm. Depending up on the pixel error, control algorithm will generate PWM signal for servos of pan\tilt unit to keep the target in the center of the image frame [5].
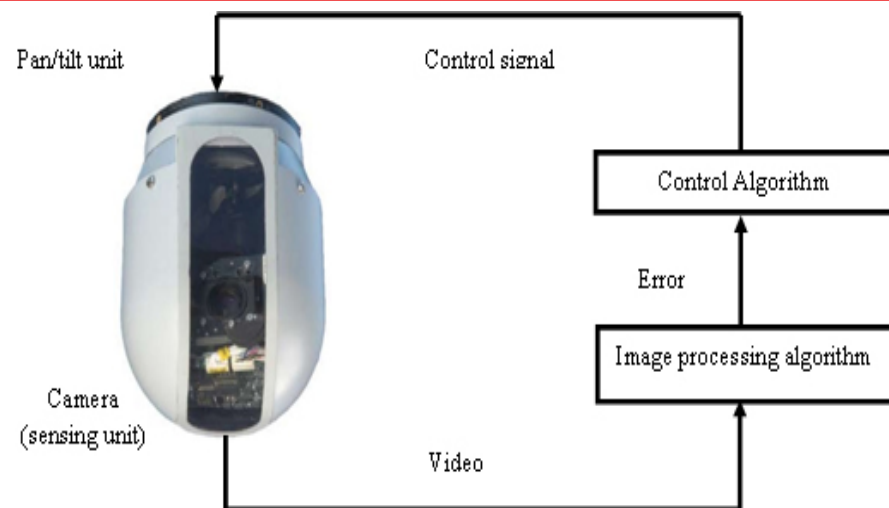
Fig1. *Control Loop for Control of Gimbaled Camera*

HARDWARE INTEGRATION AND SOFTWARE IMPLEMENTATION

For testing of algorithm, flying MAV is not feasible. So with the hardware and software used for MAV mission, a laboratory set up is made. Following hardware and software are required for making a laboratory set up.

Hardware

Hardware component includes AutoPilot Mega (APM 2.5) board, XBee PRO RF transceiver, video transceiver, joystick, servo and camera.

- **APM 2.5:** APM 2.5 is a complete open source autopilot system. It allows the user to turn any fixed, rotary wing or multi rotor vehicle into a fully autonomous vehicle. It will generate PWM signal to run the servo and integrate the system [6].

- **XBee Pro Module:** The modules operate within the ISM 2.4 GHz frequency band and are pin-for-pin compatible with each other. The XBee PRO RF modules provide quick, robust communication in point to point, peer to peer connectivity with ease. Once it is configured, it communicates with each other. For configuring the module, XCTU software is used [7].

- **Video Transmitter and Receiver Module:** Parton 5.8GHz 1200mW transmitter and receiver kit support the unique needs of low-cost, low-power wireless sensor networks [8].

- **Joystick – Logitech Attack 3 Module:**The Logitech Attack 3 is a versatile joystick is used for giving input. The Attack 3 connects to your PC with a wire through a USB Type-A connector [9].

- **Camera:** FPV 10 X Zoom cameras shown in *Fig-2* is used. It will produce image in NTSC format and in 700TVL. It has option to adjust focus and zoom. It can zoom till 10x and able to produce high quality image even at 10X. The image produced by this camera is of 720×480.

- **Servo:** HS-65 is offered with a choice of shock resistant Metal or zero wear Karbonitegears [10].

- **Gimbal:** 2-axis Gimbal is made with two servo, by means of which it can rotate in pan/tilt direction. *Fig*-2 shows Gimbaled Camera made with FPV 10X camera and servos in NAL, Bangalore.
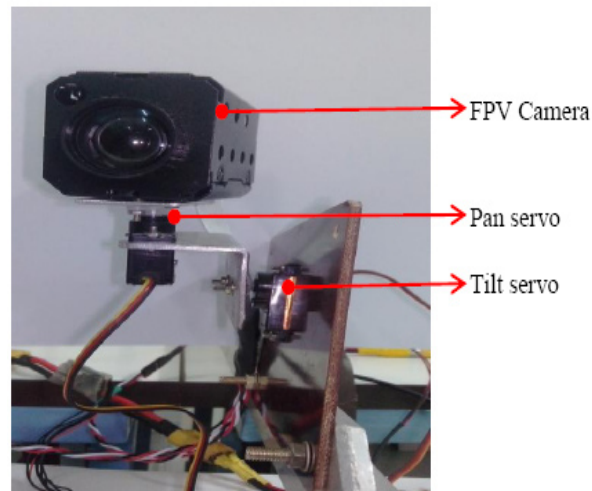
Fig2. *Gimbaled Camera made with FPV 10X camera and servos*

Software

Software such as Mission Planner and Matlabare used in the laboratory setup. Mission is a ground control station.Mission Planner can be used as a configuration utility or as a dynamic control supplement for your autonomous vehicle[11]. Matlab is used for image processing. Image from camera is processed in Matlab and position of target in image is computed. By means of which PWM signal required for bringing the target in centre of image is calculated. Controlled signal is sent to servo only by mission planner. Thus, to send continuous signal from Matlab to mission planner, virtual socketing is made.Mission planner has an option for python scripting by which we can control the MAV through scripting instead of giving inputs in GUI.Python scripting option in Mission Planner is used to create socket. In Matlab virtual socket is made forever and control signal is send forever. Missionplanner read the signal from socket and gives input signal to the mentioned channel. By configuring the input and output pins for gimbal servo, control signal from Matlab is able to control the gimbal servo.

Fig-3 shows closed loop for control of gimbal camera and its laboratory set upformed by connecting all the hardware.
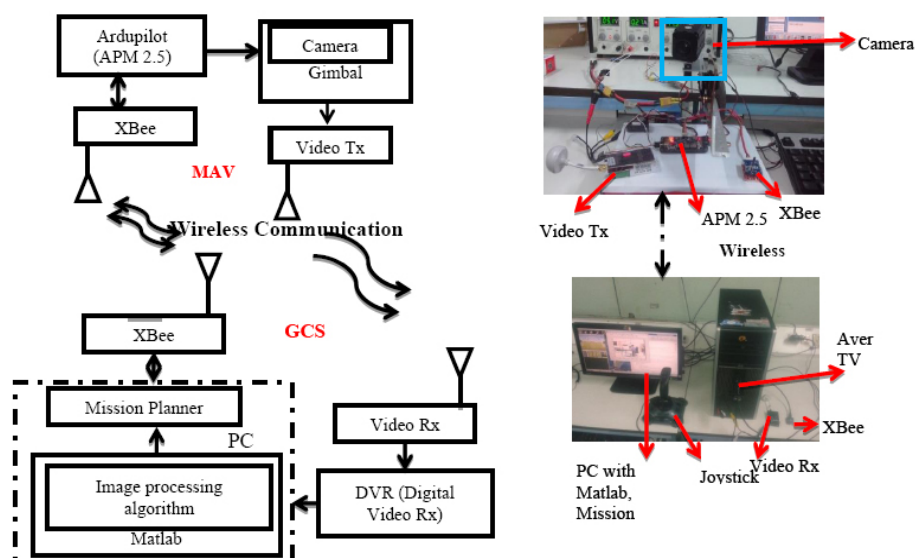

Fig3. *Closed loop for control of gimbal camera and its laboratory set up*

CONTROL ALGORITHM

For controlling of camera, object has to be tracked continuously. For tracking of target in image, Mean shift algorithm with background histogram is used. Position of target in image is computed by Mean shift algorithm. Deviation of the position of target in image from the center of image is calculated. According to the deviation, controlled input to servo is computed by the algorithm. Since we use 2-axis gimbal, pan motor and tilt motor corresponds to x and ycoordinates position of target in image respectively. Let (x_c, y_c) be the center of the image, computed from image height and width of image. x_d, y_d be the computed position of target in image.

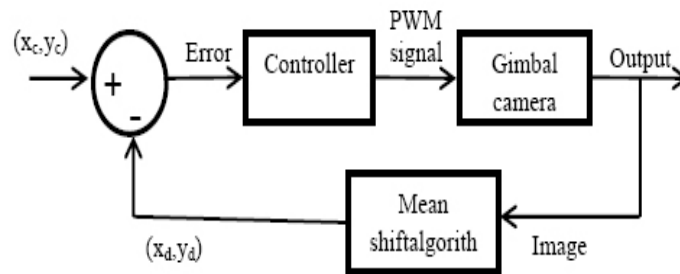$$dx = x_c - x_d \tag{1}$$

$$dy = y_c - y_d \tag{2}$$

Fig4. *Control scheme*

Fig-4 shows control scheme for control of gimbal. According to error control algorithm will compute PWM signal to rotate the servo. PWM generation for servo control depends up on the properties of servo used in gimbal and alignment of servo arm with the camera

Properties of Servo

Angle range: $\pm 45°$

PWM range: 1100μs-1900μs (increase or decrease above PWM range won't have effect in movement of servo arm)

Neutral position: 1500μs

For maximum angle change 90^0, PWM change of 800μs is required. Thus for 1^0 angle rotationPWM change of 8.89μs is required. For PWM change from 1100μs to 1500μs servo arm rotates from -45^0 to 0^0 and for PWMchange from 1500 to 1900 servo arm rotates from 0^0 to $+45^0$. x co-ordinate and y co-ordinate of position of target in image is used to control Pan and tilt servo respectively. Open loop response of system is considered to study the movement of camera and its effect in change of position of target in image.

Open Loop Response of System

Camera produces an image of size 720× 480 pixel with 360×240 as center of image.When pan PWM is decreased, the target in image moves right in image. When pan PWM is increased, the target in image moves left in image. When tilt PWM is decreased, the target in image goes up in image. When tilt PWM is increased, the target in image goes down in image. From these inferences, to control the camera, following points is must.

1. When target is in left half of image (in real target will be in right side of camera FOV), to bring target in center of image, PWM given to pan servo should be decreased.

2. When target is in right half of image (in real target will be in left side of camera FOV), to bring target in center of image, PWM given to pan servo should be increased.

3. When target is in upper half of image (in real target is below the camera FOV), to bring target in center of image, PWM given to tilt servo should be increased.

4. When target is in lower half of image (in real target is above the camera FOV), to bring target in center of image, PWM given to tilt servo should be decreased.

In order to find relation between pixel difference and PWM required for having the target in center, Open loop response of system for the target placed 5 feet away from the camera is considered. Initially servo PWM for pan is at 1500μs and servo PWM for tilt is at 1100μs.At this position of camera, the target is at (361,241) in image (i.e. approx. center of image). Now, tilt servo PWM is kept constant and pan servo PWM is changed sequentially.

Table1. *Target position in image for change in pan servo PWM*

Position of target in image	PWM(μs)										
	1300	1400	1450	1480	1490	1500 (initial)	1510	1520	1550	1600	1700
x	617	488	431	386	373	361	448	338	290	232	103
y	241	240	241	239	240	241	239	240	239	241	240

When servo PWM is changes and its corresponding pixel shift in position of target in image is calculated. *Table-1* shows the target position shift in image for change in pan servo PWM from its initial value while tilt servo is kept constant. There is no considerable change in y-coordinate of position of target in image. *Fig-5* shows the pixel change in position of target in image with respect to the initial position of target in image. When PWM is increases more than 200μs targets goes out of image. It inferences at a distance of around 5 feet, 1 pixel change requires 0.7692μs PWM change. So for dx pixel, (0.7692 × dx)μs PWM change is required. But in real time when servo suddenly moves large, image processing algorithm cant able to track the target, so rate of change of PWM is decreased. Rate of change is acquired from experiments.

If target is in right half of image pixel difference dx computed will be in negative and vice-versa. From the inference of all these experiments, the empirical formula used for gimbal camera pan servo control to have target in center is

$$n(f) = n(f-1) - \frac{0.7692 \times dx}{3}$$ (3)

Where

dx - Difference in position of target from center of image in x-coordinate
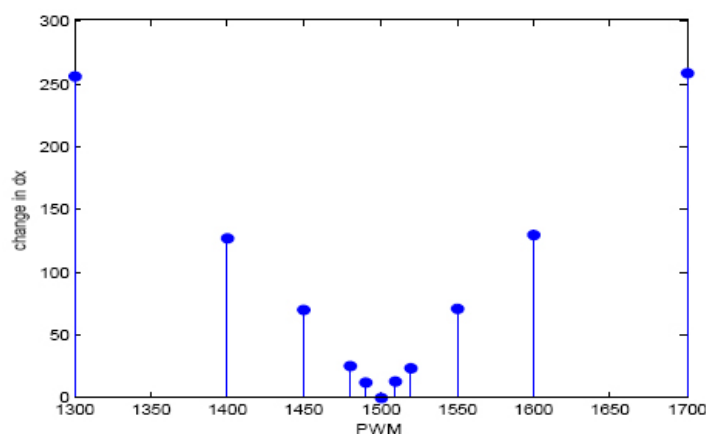
$n(f)$ =PWM of pan servo at frame f in μsec.

Fig5. *Target position shift in image for pan PWM change*

Similarly for the tilt servo, the empirical formula used to have target in center is

$$m(f) = m(f-1) + \frac{0.7692 \times dy}{3}$$

(4)

Where

dy - Difference in position of target from center of image in y-coordinate

$m(f)$ = PWM of tilt servo at frame f in μsec.

RESULTS

Closed loop experiments are carried out with empirical equation, if we add the control equation in the APM 2.5 source code it will takes more time for processing. In order to overcome this problem, PWM is computed in Matlab. Through virtual socket, Mission Planner able to read the PWM values from Matlab and given as input to the APM 2.5 board.

Stationary Object

Consider a stationary target, which is kept 5ft away from camera in such a way that when pan servo is at PWM 1200μs and tilt servo is at PWM 1100μs, target is at corner of image. (i.e. maximum distance from center of image).Depending upon difference in pixels from center of image, algorithm compute PWM is shown in *Table-2*. Change of position of target in image according to servo arm rotation is shown in *Fig*.6. Algorithm took 0.856 secto bring the target from corner of image to center of image.

Table2. *Difference in position of target from center of image and PWM computed by algorithm*

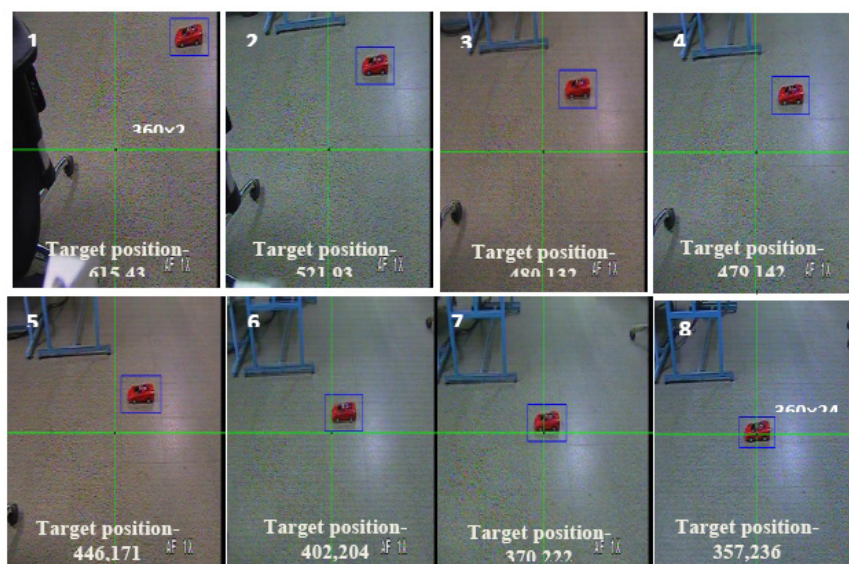| Frame no	Delta x(dx)	PWM given for pan servo	Delta y(dy)	PWM given for tilt servo
1	-255	1265	197	1150
2	-161	1306	147	1188
3	-120	1337	108	1215
4	-119	1367	98	1241
5	-86	1390	69	1258
6	-42	1400	36	1267
7	-10	1403	18	1272
8	3	1402	4	1273

Fig6. *Change of position of target in image respective to servo arm rotation*

Moving Object

Initial position of servo arm should be given for algorithm to track the target, because PWM is changed consecutively according to pixel difference in image. If the manual input of Initial position of servo arm is different from the actual position of arm, then in succeeding frame algorithm will lose the target from frame. Position of servo arm (i.e initial PWM) displayed in the Mission Planner is manually entered in the algorithm. The target selected by the operator in first frame is given as input to the algorithm, and then algorithm starts moving the camera to have target in center of image. Laboratory experiment is conducted by moving target randomly with camera at fixed place. Since camera is kept constant, target is made to move within the FOV of camera.

- Target is moved randomly with some boundary condition. (i.e. target should be always in image frame)

- Initial position PWM of servo arm should be given correctly.

- *Fig-7* shows the pixel difference dx and its corresponding PWM generated for pan servo by the algorithm.

  ➤ Since it is feedback control, control action is taken after the effect. Thus, when target suddenly changes its path, error is large at that frame, then algorithm starts decreasing it

  ➤ When error is large, the controller output is also large and vise-versa.When the error is around zero, PWM is constant at that point.
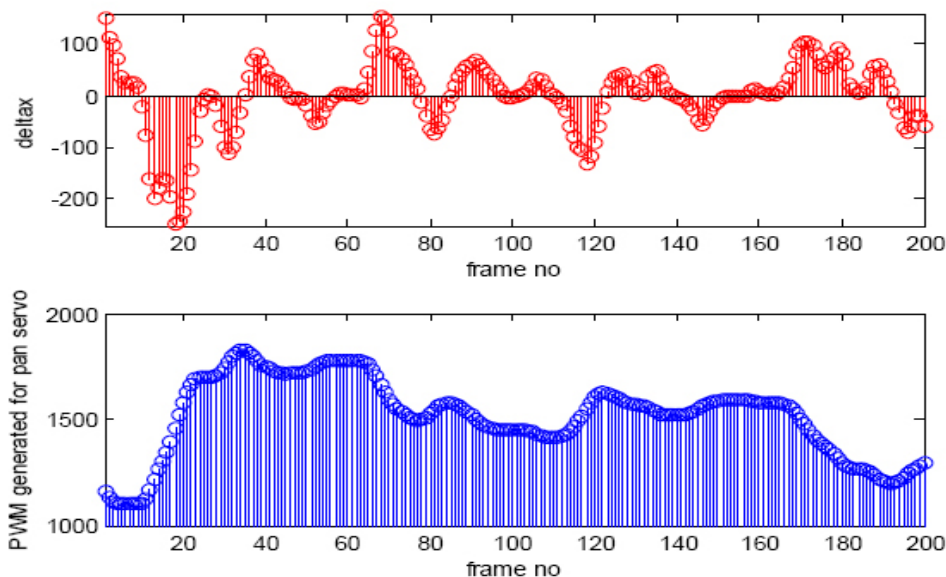

Fig7. *delta x(dx) and PWM computed for pan servo by the algorithm.*

- From *Fig-7*and *Fig-8,* we can inference that for frame 1, dx is more, and dy is less, so dx is decreased successively. But, since dy is very small, change in PWM is very small. So it does not have effect in pixel shift. When PWM changes continuously, after some iteration, we can see change in error. This is because, resolution of servo is 10 (i.e. ability of servo to detect minimum change in PWM). Thus according to empirical formulas eq. 1 and eq. 2, to have minimum PWM change it should have minimum pixel difference 40. When it have pixel difference less than 40, PWM generated won't have effect in system, thus error won't change. But in succeeding frames, for constant error, PWM will increase consecutively. When servo control circuit detects change in PWM, it rotates. Then there will be a change of error. To overcome this problem is PID controller is used.
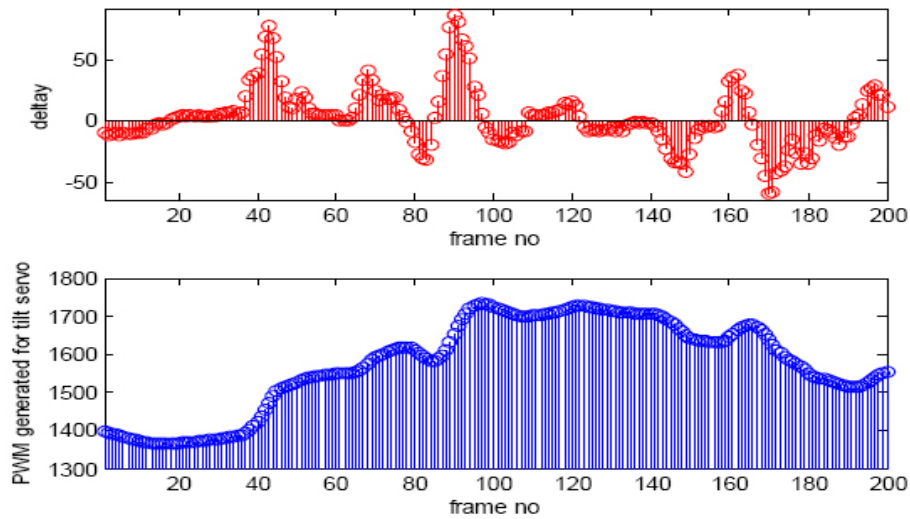
Fig8. *delta y(dy) and PWM computed for tilt servo by the algorithm.*

PID Controller

For control of camera, proportional controller is used as in eq. (3) and eq. (4) control input is proportional to error. So it has offset. To overcome this PID controller is used.Let $e(t)$ be the pixel difference between centers of image and target position. The ideal PID controller output written in the continuous time domain form [12] is

$$u(t) = u(t-1) + k_p\left(e(t) + \frac{1}{T_i}\int e(t)dt + T_d\frac{de(t)}{dt}\right) \qquad (5)$$

Since our system is discrete, the discretized PID controller equation is

$$u(t) = u(t-1) + k_p\left(e(t) + \frac{T_s}{T_i}\sum_0^t e(i) + T_d\frac{e(t)-e(t-1)}{T_s}\right) \qquad (6)$$

From eq. (2) k_p =0.2564, it depend upon the competence of image processing algorithm to track object when position of target in image is changing due to camera movement.

T_s -time interval between succeeding inputs to controller
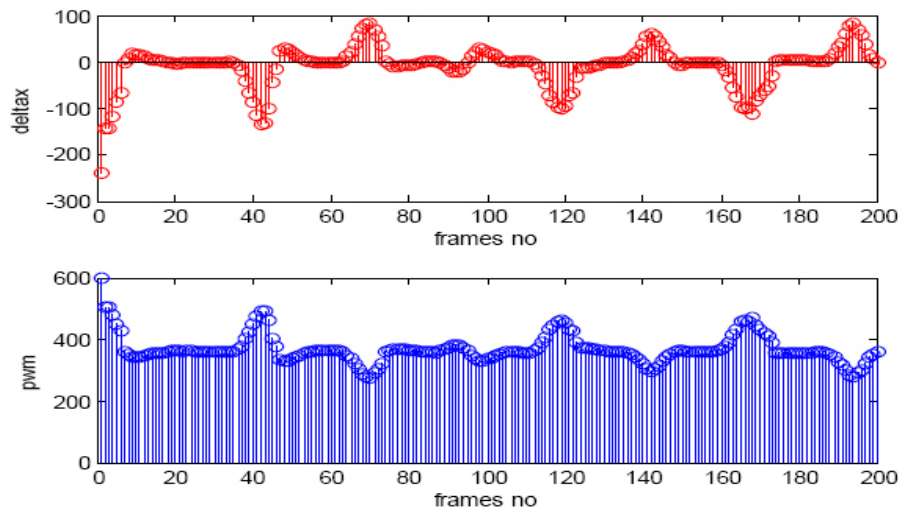
T_i -Integral time

T_d -Derivative time

Fig9. *delta x (dx) and PID controller output for pan servo*

Tuning of PID controller is done by experiments. Since, integral controller have wind up problem, it increases settling time. In order to avoid it, the integral term is reduced to zero below a threshold (ε). Threshold (ε) value is pixel value below which integral term have adverse effect. ε is obtained by experiments. Thus, controller act as PID above threshold and below threshold it act as a PD. Fig-9 shows delta x (*dx*) and PID controller output for pan servo. Offset is reduced and system steady state error also reduced.

Conclusion

Technical development and low cost of MAV's increased its use in both military and civilian purpose. MAV has camera with gimbal set up to control motion of camera. In order to get a stabilized video, it is necessary to lock the target to camera sensor and to have high resolution of target in image, camera gimbal is controlled in such a way that target will be around center of image. For locking of target to camera sensor, target has to be tracked continuously and according to the movement of target in image camera gimbal is moved. To track the target continuously image processing algorithm is used. Despite various method, background weighted histogram based mean shift target tracking has less error and efficient. It is used to track the target and to compute the position of target in image. In order to achieve the objective and to use in real time, hardware integration with software is required. Using the open loop response, empirical formula for controlling is derived. Now the closed loop system is able to lock the target to camera sensor with target in center of image. To improve the performance PID controller is also added.

References

1. Bortoff,S (1999). The university of torontorc helicopter: a tested for nonlinear control. IEEE: International Conference on Control Applications Proceedings, (Hawaii), pp. 333–338.

2. Sugeno, M. Hirano,I. Nakamura,S and Kotsu,S (2007). Development of an intelligent unmanned helicopter. IEEE: International Conference on Fuzzy Systems Proceedings of 1995, (Yokohama, Japan), pp. 34–35.

3. Nicholas J. Brake(2012). Control System Development for Small UAV Gimbal. Master of Science in Aerospace Engineering, California Polytechnic State University.

4. PlamenPetrov, OgnianBoumbarov and KrasimirMuratovski (2008). Face Detection and Tracking with an Active Camera. IEEE: 4th International Conference on Intelligent Systems.

5. Bo Zhang, Jun Huang and Jiniong Lin (2010). A Novel Control Algorithm for Object Tracking by Controlling PAN/TILT Automatically. ICETC: 2nd International Conference on Education Technology and Computer.

6. The APM 2.5 Board Overview- http://copter.ardupilot.com/wiki/apm2.5 board overview accessed on 4th September -2014.

7. Product manual for XBee/XBee-PRORF Modules, Digi's International Inc, (2012) IEEE® 802.15.4 RF Modules by Digi's International.

8. Product description of FPV 5.88g 1200mw 9CH Wireless AV Transmitter & Receiver (FPV-5812) http://transmitters88.en.made-in-china.com/product/SosnYCbyyvRM/ accessed on 9th September -2014.

9. Attack™ 3 Joystick -http://www.logitech.com/en-in/support/attack-3-joystick?osid= 14& bit =64 accessed on 4th September -2014.

10. Hitech 65 mg servo-http://hitecrcd.com/products/servos/micro-and-mini-servos/analog-micro-and-mini-servos/hs-65mg-mighty-metal-gear-feather-servo/product accessed on 8th November -2014

11. Mission Planner-http://planner.ardupilot.com/accessedon 4th September -2014.

12. Astrom, K. 1 and Hagglund,T(1995). "PID Controllers: Theory, Design and Tuning" International Society for Measurement and Con., 2nd Edition, NC, USA.