VISION BASED TRACKING OF MOVING TARGET FOR AUTONOMOUS GROUND VEHICLE

SangramK.Behera[1], Hemanth Kumar[2], Bharath.M.K[3], GokulM[4] and S.M.Vaitheeswaran[5]
[1] Student (M.Tech, NITK Surathkal), Mangalore, Karnataka, 575025
[2, 3] Project Assistant, NAL, CSIR, Bangalore, Karnataka, 560017
[4]Student (M.Tech, Amrita Vishwa Vidyapeetham)
[5] Principle Scientist, ALD, NAL, CSIR, Bangalore, Karnataka, 560017 (corresponding author)

Abstract

The tracking of moving objects using a camera mounted on an autonomous ground vehicle time to keep the target within the image on an autonomous ground vehicle is addressed. The Continuously Adaptive Mean Shift (CAMShift) algorithm is used to identify the moving target based on the color distribution of the target histogram. The CAMShift algorithm adaptively adjusts the track window`s size and the color distribution pattern of targets during tracking and uses a background-weighted histogram to distinguish the target from the background. A control scheme is designed for yaw and position control to keep the tracked object in the field of view. The system is practically implemented and tested using the Arduino platform and a cheap low cost web based camera.

Nomenclature

x_c, y_c = Center of mass of search window.
M_{00}, M_{10}, M_{20} = Zeroth order, First order and Second order Moment.
l, w = length and width of the search window.
X, Y = Center pixel location of detected target
UART = Universal Asynchronous Receive and Transmit

I. Introduction

There is an increased interest in the development of autonomous ground vehicles from both industry and academia to replace the human or human operated vehicles for performing explorations in unknown and dangerous environments. An essential function required for the autonomous vehicle is to accomplish real time tracking of moving objects using a camera. The goal of object tracking is to find an object’s location in consecutive video frames. Object tracking’s applications include perception and control for autonomous surveillance systems, identifying and neutralizing threats in missile defense, optimizing traffic control systems, and improving human-computer interaction. Many different algorithms have been proposed for object tracking, including mean-shift tracking, optical flow, and feature matching. Each algorithm has strengths in certain environments and weaknesses in others.

Tracking algorithms can be classified into two major groups, namely state-space approach and kernel based approach. State-space approaches are based largely on probability, stochastic processes and estimation theory, which, when combined with systems theory and combinatorial optimization, lead to a plethora of approaches, such as Kalman filter, Extended Kalman Filter (EKF) [3], Unscented Kalman Filter (UKF)[4], Particle Filter (PF)[5]. The ability to recover from lost tracks makes State-space approach one of the most used tracking algorithms. However, some of them require high computational costs so they are not appropriate for real time video surveillance systems. The Mean Shift (MS) algorithm is a non-parametric method which belongs to the second group. MS is an iterative kernel-based deterministic procedure which converges to a local maximum of the measurement function under certain assumptions about the kernel behaviours [6]. CAMShift (Continuously Adaptive Mean Shift) algorithm [7] is based on an adaptation of mean shift that, given a probability density image, finds the mean (mode) of the distribution by iterating in the direction of maximum increase in probability density. CamShift algorithm has recently gained significant attention as an efficient and robust method for visual tracking. A number of attempts have been made to achieve robust, high-performance target tracking [8][9][10].

.

In this paper the CAMShift algorithm is used to identify the moving target based on the color distribution of the target histogram. The CAMShift algorithm adaptively adjusts the track window`s size and the color distribution pattern of targets during tracking and uses a background-weighted histogram to distinguish the target from the background. A control scheme is designed for yaw and position control to keep the tracked object in the field of view. The system is practically implemented and tested using the Arduino platform and a cheap low cost web based camera.

The rest of the paper is organized as follows: Section II presents the original mean shift and CAMShift algorithms. Section III discusses the implementation of a Pulse Width Modulation (PWM) scheme to control the gimbal system of the camera to keep the object in the field of view over a wide angular range. Experimental results are presented in Section IV. Section V concludes the scheme.

II. Detecting and Tracking target using fixed Camera

The continuous adaptive mean-shift (CAMShift) algorithm is used to detect and track the target from the video frame of fixed camera. CAMShift algorithm is the modified version of the mean shift algorithm. Mean shift is a kernel-based tracking method which uses density-based appearance models to represent targets. The method tracks the targets by finding the most similar distribution pattern in a frame sequences with its sample pattern by iterative searching. It has been widely used because of its relative simplicity and low computational cost, but mean-shift would fail in changing the track window`s scale, as targets move toward or away from the camera.[1]

The CAMShift algorithm is proposed to overcome the problems of mean shift tracking. CAMShift adaptively adjusts the track window`s size and the distribution pattern of targets during tracking. CAMShift algorithm can be used to track the distribution of any kind of feature that represents the target. Here we use color data to represent targets for CAMShift, this would give a low complexity and practical performance to the method.

A. CAMShift Algorithm: CAMShift basically works with a multicolor histogram to represent the target. This accumulated histogram is used to compute the probability distribution of the corresponding target in every frame for CAMShift tracking. Sometimes CAMShift fails to track targets in several complex situations such as tracking objects which have similar colors with the background, to reduce the target and background color interference, a popular approach is to generate the target color histogram with a weighted scheme. A weighted histogram calculation function gives higher weights to pixels closer to the object center, since the further pixels are more likely to be from background. [1]

CAMShift basically works as follows. First, target`s initial search window is selected and its color histogram is computed. Each frame of the sequence afterwards is converted to a probability distribution image relative to the target`s histogram. Then the new size and location of the target are computed via mean-shift from this converted image, and are used as the initial size and location of the target for the next iterations of the algorithm.

The principles of our algorithm can be summarized in the following steps. [1, 2]

1. Initialize the search window`s location and size.
2. Calculate the histogram of initial search window.
3. Calculate the probability distribution image of the current frame.
4. Calculate the new location and size of the target search window using mean-shift.
5. Use the new location and size obtained in step 4 to reinitialize the search window in the new frame, and jump to step 2.

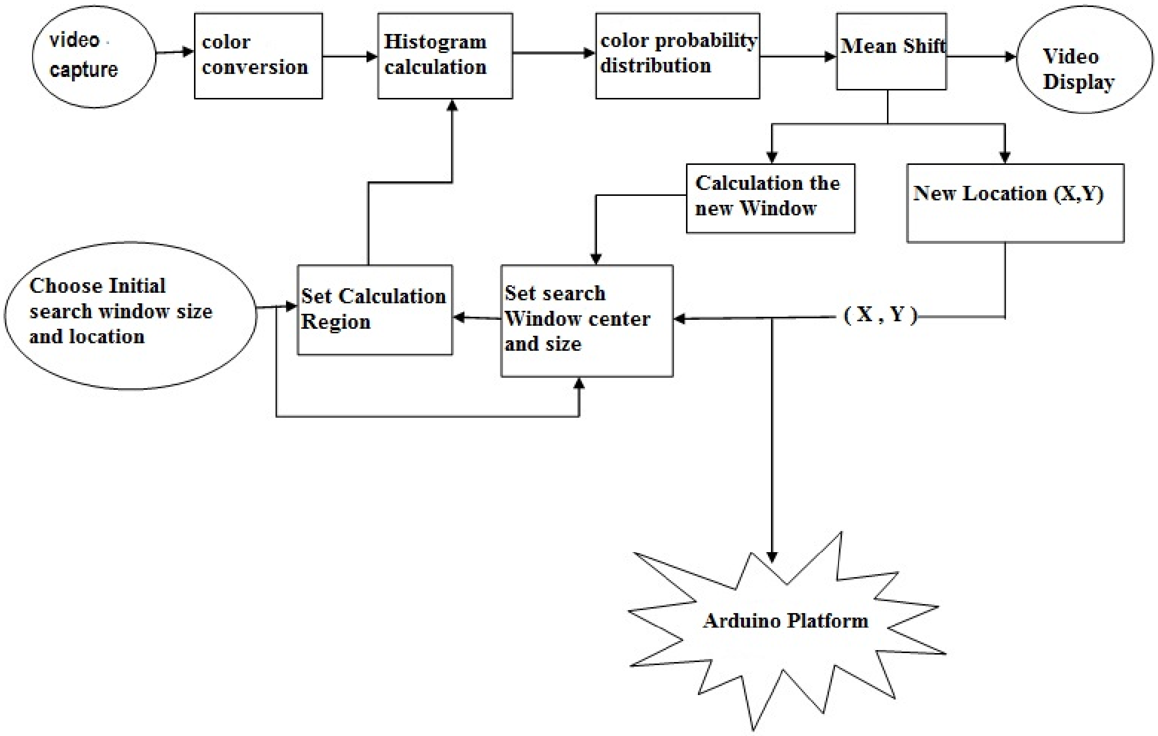


Figure 1: Flowchart of the Proposed Algorithm

1) Search Window Initialization:

In the CAMShift algorithm, the initial location of a target is selected manually by a user. After manually locating the target by a surrounding rectangle, its two dimensional colour histogram is calculated for further processing in the next steps.

.

2) Color Histogram Generation:

To obtain robust results, we use HSV (Hue Saturation Value) color space in our algorithm for the color histogram generation. HSV color space separates out color(H) from its saturation(S) and brightness(V) values, which would improve our tracking performance. For target representation, we only choose hue(H) color channels that represent the target`s main color features. In our project work, we are taking red blob as moving target. The histogram of red blob is shown below.

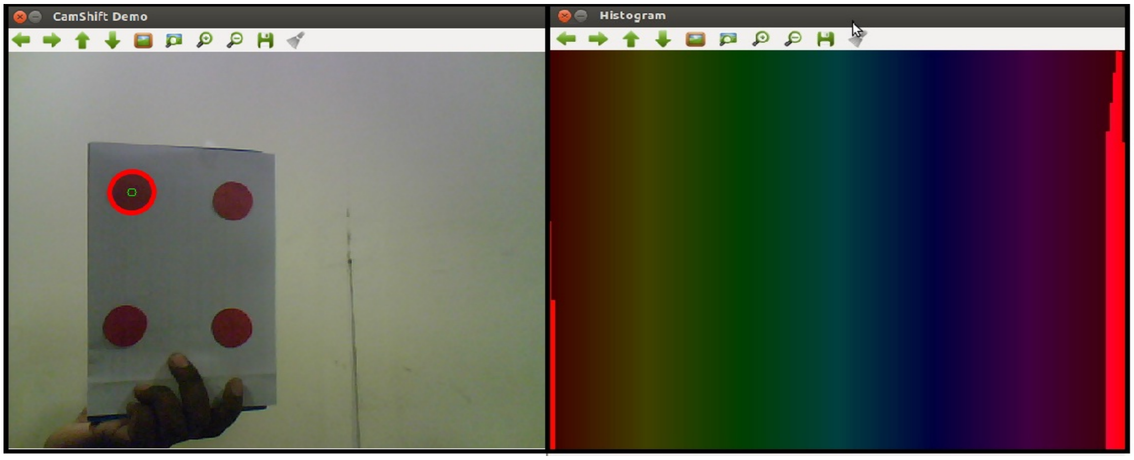


Figure 2: Histogram of the red blob during tracking

3) Probability Distribution Image Generation

A common method to generate a probability distribution image is histogram back projection. Histogram back projection of the target histogram with a frame generates a probability distribution image in which each pixel`s value associates with the corresponding bin of the target histogram. In step 3 we calculate the back-projection of the target histogram with the resulting image of step 2. This will produce a probability distribution image which will be used by mean shift to calculate the target`s new position.

4) Mean-shift Application

To calculate the new location of a target, the mean-shift algorithm is used. Mean-shift takes a probability distribution image and an initial search window, computes the window`s center of mass, and then re-centers the window at the computed center of mass. This movement will change what is under the window, and so the re-centering process is repeated until the movement vector converges to zero. The last calculated center of mass will be the new location of the target. [1, 4]

The following equations are used to calculate the search window`s center of mass (x_c, y_c):

$$\begin{cases} x_c = \frac{M_{10}}{M_{00}} \\ y_c = \frac{M_{01}}{M_{00}} \end{cases} \qquad (1)$$

Here the zeroth and first moments are calculated as:

$$\begin{cases} M_{00} = \sum_x \sum_y I(x,y) \\ M_{10} = \sum_x \sum_y xI(x,y) \\ M_{01} = \sum_x \sum_y yI(x,y) \end{cases} \qquad (2)$$

Where I(x,y)is the intensity value of point (x,y) in the probability distribution image.

Search window`s new size can be computed as follows:

$$\begin{cases} l = \sqrt{\frac{(a+b)+\sqrt{b^2+(a-c)^2}}{2}} \\ w = \sqrt{\frac{(a+b)-\sqrt{b^2+(a-c)^2}}{2}} \end{cases} \qquad (3)$$

Where l and w are the long and short axes of the search window respectively, and a, b, and c are obtained from

$$\begin{cases} a = \frac{M_{20}}{M_{00}} - x_c^2 \\ b = 2\left(\frac{M_{11}}{M_{00}} - x_c y_c\right) \\ c = \frac{M_{02}}{M_{00}} - y_c^2 \end{cases} \qquad (4)$$

And the second order moments are calculated from

$$\begin{cases} M_{20} = \sum_x \sum_y x^2 I(x,y) \\ M_{02} = \sum_x \sum_y y^2 I(x,y) \\ M_{11} = \sum_x \sum_y xy I(x,y) \end{cases} \quad (5)$$

Finally the new size and location of the search window are used to iterate the algorithm from step 2.

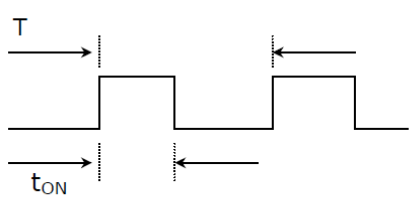
III. Tracking the Moving Target using PAN-TILT Gimbal Control

Tracking of the moving target is achieved using PAN-TILT gimbal mechanism. Gimbal structure refers to arrangement of servos to achieve 2-degrees of freedom. The two degrees of freedom refers PAN and TILT movement of servos. PAN servo is responsible for moving the camera in horizontal direction, and TILT servo is responsible for moving the camera in vertical direction. Using CAMShift algorithm the target blob is tracked and the centeroid of the detected target is obtained and this value is sent to the microcontroller which controls the two servo motors to keep the target in the field of view of the camera. Communication of the centeroid value of target from PC to microcontroller is achieved by using UART(Universal Asynchronous Receive and Transmit) protocol.

A. Servo motors with PWM Control

Servo motors are small controllable motors with many different speeds, torques and sizes. A servo motor will have 3 wires power, ground and control. A servo-motor is an actuator with a built-in feedback mechanism that responds to a control signal by moving to and holding a position, or by moving at a continuous speed. Servo's receive pulse width modulated (PWM) signals to determine in which manner to move. Servo motors are controlled through the control line. The pulse width signals sent to the Servo control wire determines how the motor will move, either clockwise or counter clockwise.

Pulse width modulation is a kind of averaging the power delivered to the motor rather than varying the input voltage to the motors. PWM will vary the speed of servo by changing the signal duty cycle which will automatically vary the power operating the Servo motors in order to increase or decrease the speed of the servos:



T is the period in (s).
F = 1/T is the frequency in (Hz).
(t_{ON} / T) is the Duty Cycle in (%).

Figure 2: example of a PWM signal

Figure 3 indicates how different pulse widths correspond to different position of the motor. The servo receives a pulse every 20milliseconds (.02 seconds), the length of the pulse width determines how much the motor will rotate. When the PWM is less than 1.5ms the motor will move to the 0 position and hold. When PWM t_{ON} = 1.5ms, the motor will rotate to the 90 degree position and for PWM greater than 1.5 ms, the motor will rotate to the 180 position. For every 5 Micro second PWM signal there is one degree of rotation.

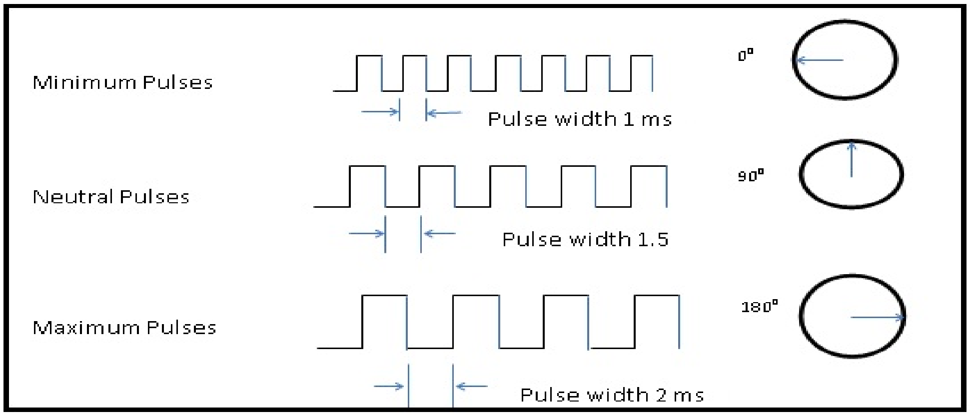


Figure 3: PWM signal to control servo motor

B. Serial Communication

The tracked centroid of the target is communicated to microcontroller using UART protocol. The centre consists of two values X and Y. Since UART transmits 1 byte at a time first the X value is sent and then the Y value. At the receiver the controller verifies that corresponding X and Y values are received to control the respective motors. Thus time sharing is achieved to control the motors accurately with respect to the centroid of the target in order to keep it in the field of view of the camera.

C. Position Control

The objective of position control, is to always keep the chosen target within the reference frame of the image as it drifts away from the reference frame. The servo motor connected to the PAN-TILT gimbal structure has maximum angle of rotation 0 to 180 degree. Initially both PAN and TILT servo is fixed at 90 degree position.

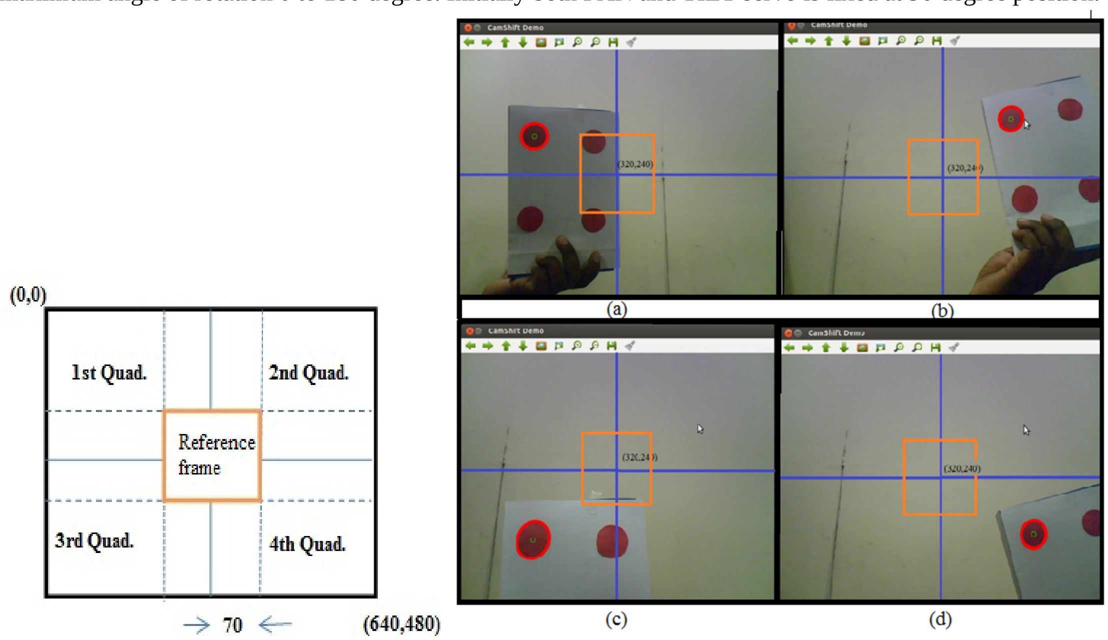


Figure 3: Different location of object on video frame (a) 1st quadrant (b) 2nd quadrant (c) 3rd quadrant (d) 4th quadrant

A reference frame is initialized before we start the tracking procedure. The reference frame can be varies based on the target size as required. The reference input cannot be a single point. It is almost impossible to stabilize the system on this single point .If a single point was chosen, the gimbal would always oscillate around this point, however it would never stop because there is a slight variation in detection using CAMShift algorithm i.e the centroid varies with in a small pixel range. So a small frame is defined as a reference input which is highlighted in orange color on the video seen as shown in figure 3. The coordinates of the reference input are also in the table 1 which exactly represents a frame in center of the video.

Table 1: Reference Input.

Axis	X	Y
Xrefmin	250	-
Xrefmax	390	-
Yrefmin	-	170
Yrefmax	-	310

The differences between reference input seen in the table 1 and current position of the target object is defined as error signal in code. According to this error value, the driving direction of the gimbal is chosen. In other words, according to results of this comparison, gimbal can be driven right-left or up-down. For instance, if the current position of the target object is close to the center of X axis and far from the Y axis, it means that the gimbal should be made a thrust to up or down in order to centralize Y axis. If it is close to the center of Y axis and far from the X axis, the gimbal should be made a thrust right or left in order to centralize X axis. After deciding right-left or up-down pairs, the algorithm chooses one of the right or left pairs and one of the up or down pairs to decide the moving direction of the gimbal.

After deciding the motion direction of the gimbal, error signal goes through the controller. The controller output is used to decide the pulse width of the PWM signal. There is a larger frame was experimentally set around the reference frame. All boundaries of the larger frame around the reference frame and gimbal moving direction control through PWM signal are explained in table 2.

Table 2: Target position in video frame, moving direction of gimbal and Percentage level of PWM

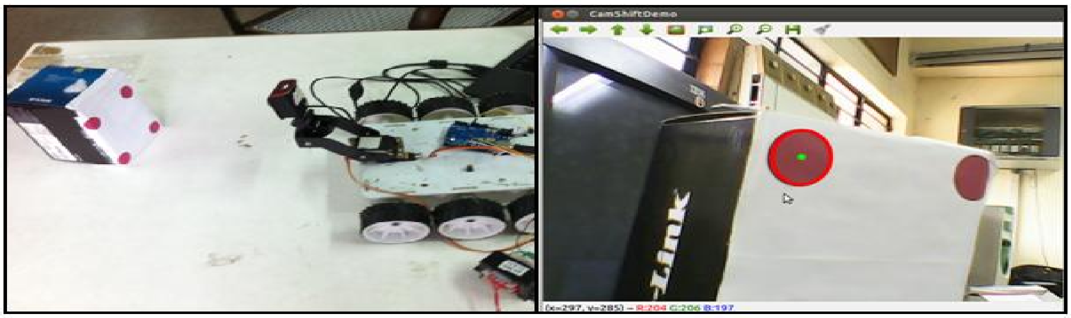
Target Position	Gimbal moving direction	PWM pulse width (in ms)	Target Position	Gimbal moving direction	PWM pulse width (in ms)
X<125	Left	$1 < t_{ON} < 1.25$	Y < 85	Up	$1 < t_{ON} < 1.25$
125 < X < 250	Left	$1.25 < t_{ON} < 1.5$	85 < Y < 170	Up	$1.25 < t_{ON} < 1.5$
250 < X < 390	Reference frame	$t_{ON} = 0$	170 < Y < 310	Reference frame	$t_{ON} = 0$
390 < X < 515	Right	$1.5 < t_{ON} < 1.75$	310 < Y < 395	Down	$1.5 < t_{ON} < 1.75$
515 < X < 640	Right	$1.75 < t_{ON} < 2$	395 < Y < 480	Down	$1.75 < t_{ON} < 2$

The table 2 represents the initial condition for servo movement. For the servo motor to rotate 1 degree it requires 5 microseconds duty cycle of PWM signal. Based on the centroid position of the tracked target microcontroller commands the servo motors to increment or decrement by 1 degree to maintain the FOV.

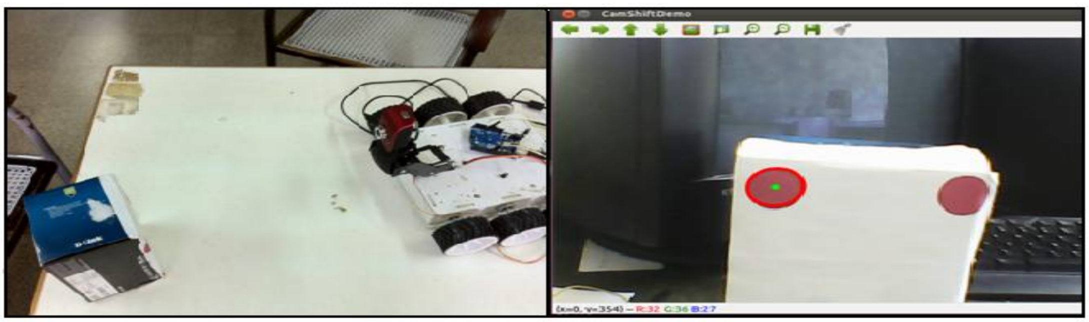
IV. Experimental Result

The experimental setup consists of a PAN-TILT gimbal structure with a camera mounted on it. The target used for tracking was a colored blob. The experimental setup is as shown in figure 4.
Figure 4(a) illustrates PAN and TILT rotation to 52 and 35 degree respectively from initial position of servos in order to bring the red blob to the reference frame. As we move the target towards left, accordingly the PAN-TILT servos move to 80 and 68 degrees respectively in order to bring the red blob to the reference frame, which is shown in figure 4(b). Similarly in figure 4 (c) and (d) shows different orientation of servos in order to bring

the red blob to the reference frame and the servo orientation were validated against the gyroscope sensor value are also shown in table 3.

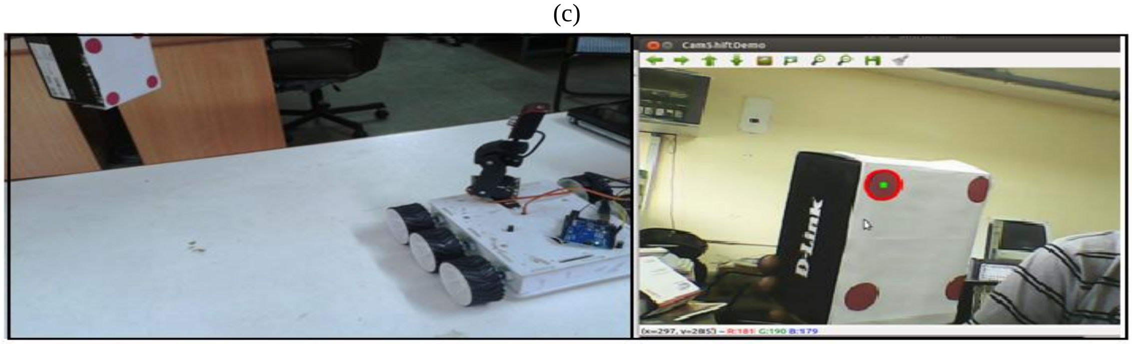


(a)



(b)

(c)



(d)

Figure 4: Tracking the moving red blob using PAN-TILT Gimbal Structure in UGV.

Table 3: PAN-TILT servo position value (in Degrees) validated against gyroscope value.

Figure	Servo Position		Gyroscope value	
	PAN(YAW) angle	TILT(PITCH) angle	PAN(YAW) angle	TILT(PITCH) angle
Figure 4(a)	52	35	53	36
Figure 4(b)	80	68	81	69
Figure 4(c)	101	120	102	121
Figure 4(d)	85	150	86	151

V. Conclusion and future work

We proposed the tracking of moving target for autonomous ground vehicle on the basis of color distribution of target using an efficient CAMShift algorithm. But by using CAMShift algorithm, we can able to track the colored object in the cluttered environment. It can be improve the efficiency of the tracking by using Kalman filter. We can also able to keep the moving target within FOV of camera using PAN-TILT Gimbal control.

Acknowledgments

The authors SMV, MKB and HKE, NAL gratefully acknowledge the funding received from AR&DB, GoI under the NPMICAV program to carry out the above work. The work is carried out at National Aerospace Laboratories, CSIR, Bengaluru, Karnataka, INDIA.

VI. References

[1] Ebrahim Emami, Mahmood Fathy, “Object Tracking Using Improved CAMShift Algorithm Combined with Motion Segmentation” IEEE, vol.- 978-1-4577-1535-8/2011.

Book

[2] G. Bradski, and A. Kaehler, "Learning OpenCV", O`Reilly, 2008, pp.337-341**.**

Proceedings

[3] Chunrong Zhangm, Yuansong Qiao, Enda Fallon, Chiangqiao Xu, “An Improved CamShift Algorithm for Target Tracking in Video Surveillance” 9th. IT & T Conference, Dublin Institute of Technology, Dublin, Ireland, 22nd.-23rd. October, 2009.

[4] G. R. Bradski. "Computer vision face tracking for use in a perceptual user interface". Intel Technology Journal, 2nd Quarter, 1998.

[5] Zulaikha Kadim, Marizuana Md. Daud, Syaimaa Solehah M. Radzi, Norshuhada Samudin, Hon Hock Woon, “Method to Detect and Track Moving Object in Non-static PTZ camera” Proceedings of the International Multi Conference of Engineers and Computer Scientists 2013 Vol I, IMECS 2013, March 13 - 15, 2013, Hong Kong

[6] Sabeti, L.Wu, Q.M.J,"High-speed Skin Color Segmentation for real-time Human tracking " In IEEE International Conference ,pp. 2378 ~ 2382, 2007.

[7] Elad Ben-Israel,Dr. Yael Moses"Tracking of Humans Using Masked Histograms and Mean Shift",Introductory Project in Computer Vision,2007.

[8] Dawei Liang,Qingming Huang,Shuqiang Jiang, Hongxun Yao, Wen Gao,"Mean- Shift Blob Tracking with Adaptive Feature Selection and Scale Adaptation", ICIP 2007. IEEE International Conference on Image Processing, 2007.

[9] J. G. Allen, R. Y. D. Xu and J. S. Jin, "Object tracking using CamShift algorithm and multiple quantized feature spaces", In ACM International Conference Proceeding Series, Vol.100, pp. 3-7, 2004.

[10] S. Kang, J. Paik, A. Koschan, B. Abidi, and M. A. Abidi, "Real-time video tracking using PTZ cameras," Proc. of SPIE 6th International Conference on Quality Control by Artificial Vision, Vol.5132, pp. 103-111, Gatlinburg, TN, May 2003.