# Real Time Object Detection & Tracking System (locally and remotely) with Rotating Camera

Ravi D. Simaria
Embedded system design, School of engineering,
R.K.University, Rajkot, India
*rvsimaria2@gmail.com*

Prof. D. S. Pipalia
Assistant Professor, School of engineering,
R.K.University, Rajkot, India
dhaval.pipalia@rku.ac.in

***Abstract -*** The task of real time detection and tracking of a moving object in a video stream is quite challenging if camera itself is moving. This paper presents an implementation of real time detection and tracking of an unknown object in video stream with 360° (azimuth) rotating camera. It also presents adaption of different object tracking algorithms and their effect on implementation. The system described in this paper contains a camera that is connected to an embedded system (standalone board) or PC/laptop. They (board/PC) are having an image processing algorithm which detects an object first and then tracks it as long as it is in the line of sight of the camera. As the object moves, the PC/laptop/embedded Board gives signal to motor to rotate the camera which is mounted on a stepper motor. To monitor Object in video user can have multiple options. If user is using laptop/PC to track object it is very simple for him because he already has a screen but in case of embedded board user can monitor the activity of the object of interest using HDMI output or streaming video on WEB server. The object can be defined directly by the end user by selecting a portion of the frame in video stream. The embedded board/PC also saves the video stream in a storage device for playback purpose.

***Key Words***: Embedded System Board, Embedded Linux OS/Linux OS, Object Detection, Object Tracking, Rotating Camera, OpenCV Libraries

*****

## I. INTRODUCTION

Detection of a moving object is necessary for any surveillance system. A static camera can detect and track an object as long as the object is inside the frame of the camera. But as the object [3] goes beyond the boundary of the camera frame, the camera stops tracking it, which is a major limiting factor for the use of a static camera. This limitation can be overcome by using a rotating camera. This paper describes a system in which a camera rotates according to the path of a moving object and follows it. Along with tracking of an object, the recording of a video stream is also necessary for playback purposes. The system will store all the data into a storage device. The storage device can be a hard disk or a flash drive. It is the responsibility of the system to delete previously stored data and start storing of fresh data from the current frame in case of storage device becomes full. The system is also connected to a display by which an end user can monitor the activity of an object. The end user can also choose the object of interest by selecting a portion of the frame on a monitor screen. So it can be very intelligent video surveillance system using an embedded system.

Two different designs, to track an unknown object in a video stream, are presented in this paper. 1. Embedded board with simple algorithm to track an object and control the motor. 2. PC/Laptop with complex image processing algorithm to track object and then control the motor using an embedded board locally or remotely. Embedded board uses embedded Linux and PC uses general purpose Linux. Working of the algorithms on both the OS are same.

The aim [1] of an object detection system is to estimate the object location in the image sequence when initial position of the object is given in the first frame. Additionally based on the application it should also calculate object based information such as orientation of the object, area of the object and/or shape of the object. Alike, the aim of an object tracker is to follow the detected object in every video frame. This work is quite challenging. If the object goes beyond video frame then camera should move in that direction and it should follow the object until it gets disappeared from the camera’s sight. If the object comes again in the sight of the camera then it should be detected and tracked again. So object occlusion should be handled properly. Because of advancement in technology and availability of low cost cameras in market, realization of this kind of system at low budget has become possible.

### *A. Embedded system*

For this application, an embedded system which is capable to process real time data e.g., video and audio is required. Many processors available in the market support this requirement. Various DSPs and ARM processor are the examples. The fact that the ARM processor supports various embedded OSs which in turn support different languages like C, C++, Python etc., makes it easy to develop an algorithm with an ARM processor. Some single board computers [7] with ARM processor and USB ports which are suitable to application’s needs are available. The camera and the storage device can directly be connected to USB port which removes any requirement of an interfacing work.

### B. *Embedded Linux OS/ General Purpose Linux OS*

Operating systems [19] based on Linux kernels are currently used in embedded system devices such as consumer electronics, networking equipment, machine control etc.

There are many versions of embedded Linux available in the market. Some of them are:

- Android (embedded)
- Maemo
- Debian
- Raspbian

### C. *Object Detection*

Objection detection [4] is a method to identify an object in a video or a picture frame. The object can be anything that is of interest for further analysis. Numbers of methods have been developed to detect a moving object using an active camera. Usually a moving object detected by static camera using background subtraction.

An object [1] can be represented by its shape, position and appearances. Some of the commonly used object representations are:

- Points
- Articulated shape models
- Skeletal models
- Probability densities
- Templates
- Active appearance models
- Multi view models

The choice of a unique feature is very crucial in object detection. Most necessary property of feature is in its uniqueness so that the object can be easily distinguished

### D. *Object Tracking*

The basic aim of object tracking is to estimate object location in image sequences. To track an object, many kind of calculations are necessary e.g. object orientation, object shape, direction of object, speed of object, area of object and etc. Major challenges [1] involved in object tracking are:

- Illumination change
- Pose variation
- Object deformation
- Scale variation
- Motion blur
- Noise in image/video sequence
- Partial/Full occlusion of object

Object tracking with moving a camera becomes challenging because many difficulties arise due to these challenges.

### E. *Rotating Camera*

In the present application, the object will be tracked as long as it is in the line of sight of the camera. A simple way to do this is to use a 360° rotating camera. The camera is mounted on a stepper motor so that it can rotate by 360° (clockwise and counter-clockwise in azimuth). The stepper motor is used to get an accurate position. Whenever the object goes out of boundary of the camera, the motor will rotate to keep the object in the frame. Selecting a suitable stepper motor [8] is very important as many factors like revolution per minute, step size, voltage specification etc. are to be taken in to consideration. For the present application, step size of the motor should be small so that the camera can cover larger area. This paper proposes 2 algorithms to move the motor for tracking the object. Interfacing of stepper motor with embedded board is easy using motor driver IC [17].

Choosing camera is crucial task for user because every embedded board doesn’t support all the cameras. So it is responsibility of user that he should check camera compatibility of embedded board first [22].

### F. *OpenCV libraries*

OpenCV is released under a BSD license and hence it’s free for both academic and commercial use. It has C++ [15], C, Python [12] and Java interfaces and it supports Windows, Linux, Mac OS, iOS and Android. OpenCV was designed for computational efficiency and with a strong focus on real-time applications.

## II. METHODOLOGY

To track the object, embedded board needs an image processing algorithm. Here these paper describes a design using 3 different algorithms to track the object in a video stream. One common thing about these algorithms is that they all use OpenCV libraries. All algorithms have their own benefits and limitations. These algorithms are:

1. Object tracking using Dominant Color:

This algorithm is developed using OpenCV library. It can easily run on embedded boards. This algorithm is needed to develop because embedded board comes with limited resources like limited processor speed and limited on-chip RAM.

This algorithm [12] detects an object of most dominant color. Basically it uses k-means clustering on the video frame and it gives centroid of most dominant color in Frame. This algorithm is limited to some object only which have dominant color. As it is very simple, it can run smoothly on embedded boards.

2. CAMSHIFT Algorithm

The CAMSHIFT algorithm [20] is based on the MEAN SHIFT algorithm. The MEAN SHIFT algorithm works well on static probability distributions but not on dynamic ones as in a movie. CAMSHIFT is based on principles of the MEAN

SHIFT but also a facet to account for these dynamically changing distributions.

CAMSHIFT's is able to handle dynamic distributions by readjusting the search window size for the next frame based on the zeroth moment of the current frames distribution. This allows the algorithm to anticipate object movement to quickly track the object in the next scene. Even during quick movements of an object, CAMSHIFT is still able to correctly track.

CAMSHIFT works by tracking the hue of an object, in this case, flesh color. The movie frames were all converted to HSV space before individual analysis.

3. CMT algorithm

Consensus-based [2] Matching and Tracking of Key points (CMT) is an award-winning object tracking algorithm,

CMT is able to track a wide variety of object classes in a multitude of scenes without the need of adapting the algorithm to the concrete scenario in any way. CMT is able to achieve excellent results on a dataset that is as large as 60 sequences. It is open source algorithm under the BSD license.

The main idea behind CMT is to break down the object of interest into tiny parts, known as key points. In each frame, it tries to again find the key points that were already there in the initial selection of the object of interest. This is done by employing two different kinds of methods. First, it tracks key points from the previous frame to the current frame by estimating what is known as its optic flow. Second, it matches key points globally by comparing their descriptors. As both of these methods are error-prone, it employs a novel way of looking for consensus within the found key points by letting each key points vote for the object center. Based on the remaining key points, the new bounding box is computed and the process continues.

CMT algorithm is capable of taking dynamic input given by user. This is an advantage of this algorithm.

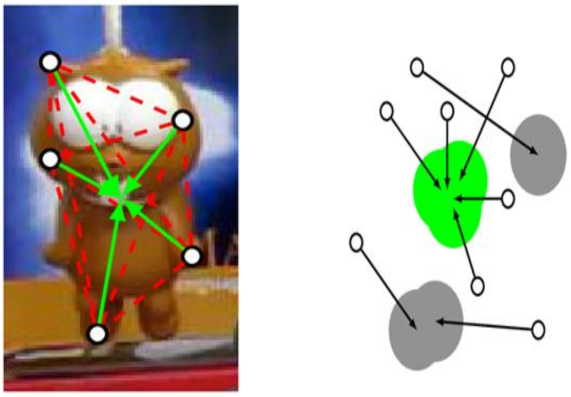


Fig. 1 CMT key points detection and clustering

## III. IMPLEMENTATION

### A. *first approach:*

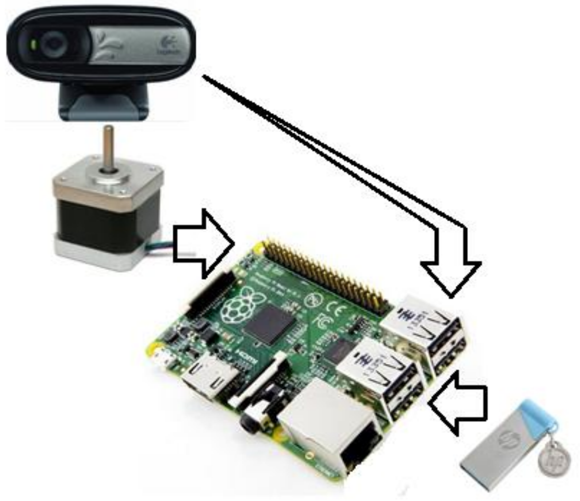


Fig. 2: Interface of camera on a stepper motor and storage device with embedded system.

First approach uses embedded board to track into the video stream. Different algorithms like CMT, CAMSHIFT and openTLD are lagging on embedded board because of limited resources. So the algorithm on embedded system should be very simpler so it can run in real time. So one algorithm that detects an object using dominant color and provides center of the object. It uses k-mean clustering on HSV conversion of frame. Algorithm is written into python and it uses OpenCV library. After center coordinates are obtained, they are pass on to stepper motor algorithm which rotates motor according to the logic. These two algorithms are connected through means signal passing is done through named pipe IPC to pass data between two different processes. Also third Writing algorithm is used to save video on hard disk. The output video stream can be seen on HDMI monitor or TV using HDMI cable.

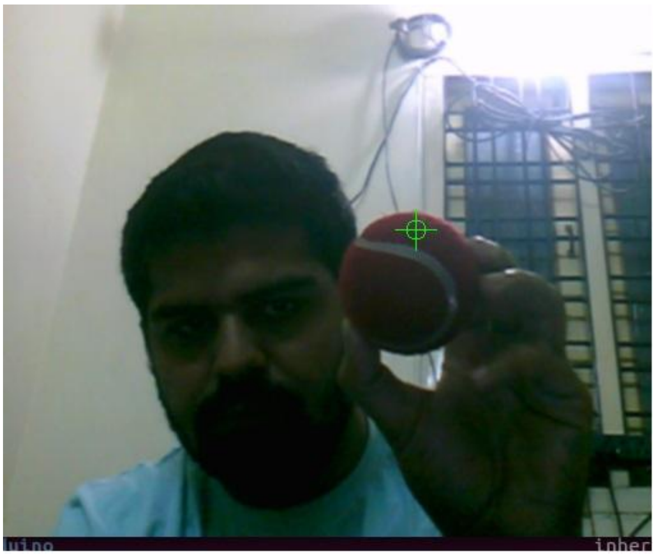


Fig. 3: object tracking using Dominant color.

The limitation of this algorithm is that it only works with dominant color object. So it will detect only few dominant colors. The main problem occurs when two dominant color object comes into one frame, then behavior of algorithm would become undefined.

### B. *Second approach:*

Now, to use complex algorithms like CMT [5] and CAMSHIFT, a special arrangement is needed. In this design PC/laptop and general purpose OS (Ubuntu distribution) are used to run this algorithm. These algorithms are running smoothly on PC because of good processor speed. Signal passing to an embedded board which is connected to Stepper Motor is done through serial communication using UART based communication. The board is running motor algorithm and when it gets the signal, it will move motor accordingly. Python provides Serial module [14] and C++ provides libserial library [16] for serial communication.

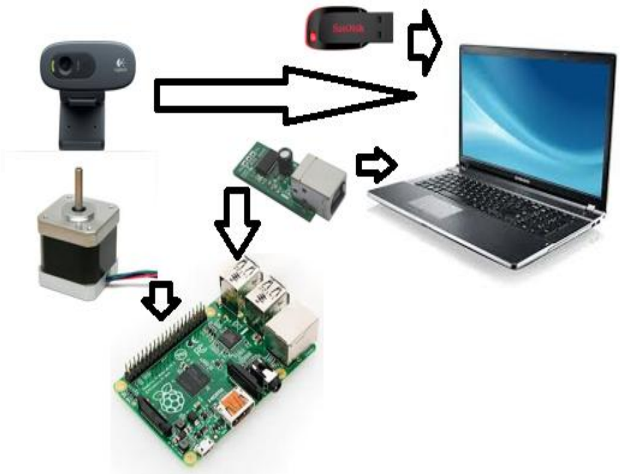


Fig. 4: Interfacing of embedded board with PC which is running image tracking algorithm.

According to the object found in video, a signal is passed to the embedded board serially. Motor algorithm running on embedded board makes decision about how to rotate a motor. And writing algorithm on PC saves video stream to hard disk for playback Purpose. Serial to TTL converter can be used to send data from laptop to embedded board.

In previous image (fig. 4) these design is illustrated. Here camera is connected to PC/laptop which is running an object tracking algorithm. Motor is connected to an embedded board which is running motor algorithm and finally board is connected to PC using Serial cable.

1. CAMSHIFT:

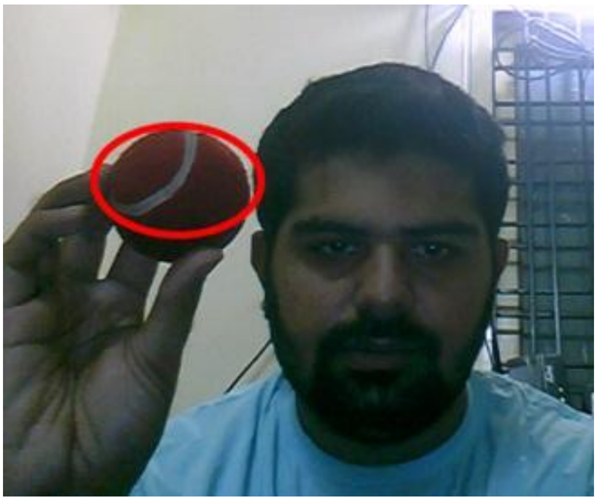


Fig. 5: object tracking using CAMSHIFT algorithm.

CAMSHIFT algorithm [6] is written in C++. It tracks object very smoothly. It accepts input directly by the user. A user just have to select the object in the video stream using a mouse. User can select the object by drawing a Bounding Box on the video stream. It will give coordinates of center point and algorithm uses that to send to motor algorithm as signal using serial programming in C++ [16].

The limitation of this algorithm is re-detection. If the object make occlusion from screen and reappears then detection of object would not be guaranteed. But in these application, advantage is that camera is moving so it would not let object get occlusion in horizontal direction.

2. CMT

CMT algorithm [5] was originally written in python. User can give input by selecting bounding box on screen. It supports many arguments with itself. It tracks any kind of object with re-detection feature.

Limitation of CMT algorithm in our application is that the user have to choose object very carefully because algorithm use key points to detect the object. If the user selects an object with most of background then moving object tracking becomes very tough because algorithm uses key points randomly. So if it takes most of the key points from the background, it will track background only. If user chooses an object with no background selected or he chooses an object with moving background it will work perfectly. One more point is that the user have to select the object such that CMT will find minimum key points.

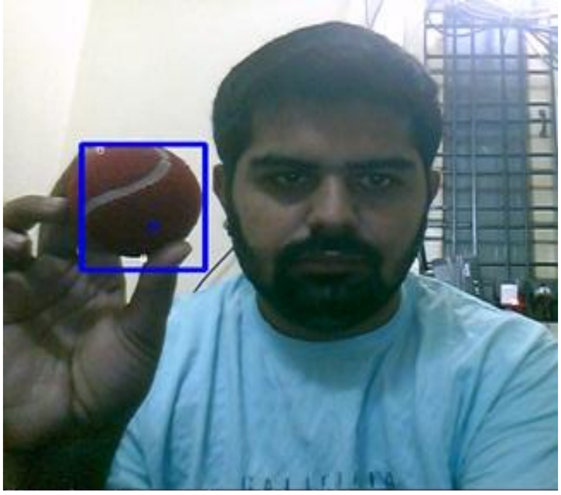


Fig. 6: object tracking using CMT algorithm.

*C. Stepper Motor Algorithms:*

There are two different algorithms created for stepper motor using basic stepper motor programming [18]. In both the approaches, the stepper motor is connected to embedded board's GPIO. Half stepping can be used for more accurate camera position. Two different algorithm developed are:

1. Boundary based Object Tracking:

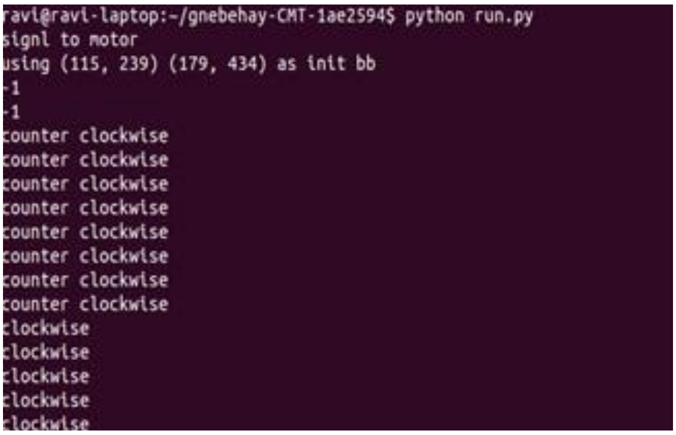
It is very simple algorithm. It takes the object co-ordinates and match them with boundary. If the object goes out of the boundary, it rotates the motor with 12 (calculated using trial and error method) steps in the direction of the object. So it will always follow the object in the video.

The main advantage of this algorithm is that the processor have to do less comparison of coordinates with boundary points. So it is an efficient algorithm but chances of object occlusion are larger.

2. Centre based Object Tracking:

It is an optimized algorithm. In this algorithm, the object would always be at the center of the video frame. Whenever the object tries to move, its coordinates will be matched with the center of video stream and the motor will be moved such that the object will be in center of the video stream again.

This algorithm is more efficient in object tracking because object will be at center at any occasion, so chances of occlusion of object are lesser. But in this algorithm comparisons of coordinates with center points become larger.



```
ravi@ravi-laptop:~/gnebehay-CMT-1ae2594$ python run.py
signl to motor
using (115, 239) (179, 434) as init bb
-1
-1
counter clockwise
counter clockwise
counter clockwise
counter clockwise
counter clockwise
counter clockwise
counter clockwise
counter clockwise
clockwise
clockwise
clockwise
clockwise
clockwise
```

Fig. 7: Signal passing to stepper motor according to object direction.

Both algorithms are developed in python and both worked perfectly. Both uses serial module with them to get signal from PC (2nd approach). In the 1st approach, algorithms get signal using IPC (Inter Process Communication). Here, as shown in the image (Fig.7), if object goes left side of camera it will sent counter clockwise signal to motor (mirror image) and when object goes right side of camera it will send clockwise signal to motor and -1 in image(Fig. 7) is denotes that algorithm is not able to detect object into the stream.

*D. Video playback algorithm*:

These simple algorithm is developed using video writing class object [21] in python that saves video stream to the hard disk (or flash drive). Whenever the hard disk is full, it stops recording, delete all the previous data and start recording afresh. It uses OpenCV libraries for recording purpose and OS module in python to check whether hard disk capacity limit is reached or not and delete previous data.

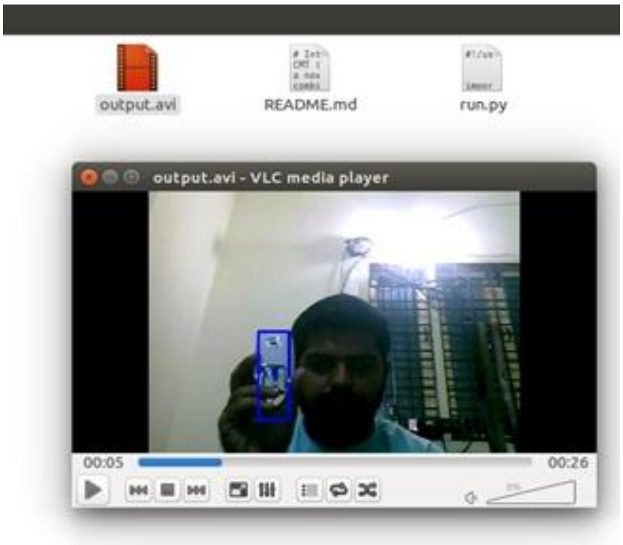


Fig. 8: playing an output video of CMT algorithm saved by writing algorithm using VLC media player.

### *E. Extended work:*

Object tracking algorithms like CMT and CAMSHIFT works very fast with PC and controlling of motor is done through embedded board. This is local configuration. Now same thing can be done using a new arrangement that makes it work remotely. In proposed configuration, the embedded board is placed at some remote location and connected to the internet via Ethernet and is powered by a good power source (on battery it will last maximum one day). The camera is connected to the embedded board and video stream server installed on board called motion. What motion do? [10] It takes the video stream from camera and stream it on internet. This stream on PC [11] can be used as input using OpenCV libraries and tracking of an object in that stream is possible. It is possible because versatility of OpenCV libraries. According to the object's coordinates, a signal is sent to the board which is running stepper motor algorithm. This time signal is passed using one more IPC called socket which is part of the socket programming. User can monitor object on PC screen and writing algorithm is writing video on hard disk. This arrangement allows user to monitor object remotely. Python has module called socket [13] and C++ have socket API's for socket programming.

This arrangement provides good result but the major problem of this arrangement is that one should have sufficient bandwidth of internet because the board has to stream a real time video over the internet. Also, the Internet lag is always present and video stream will have 2 seconds of lag. If one reduce resolution of video stream, efficiency would be increase.

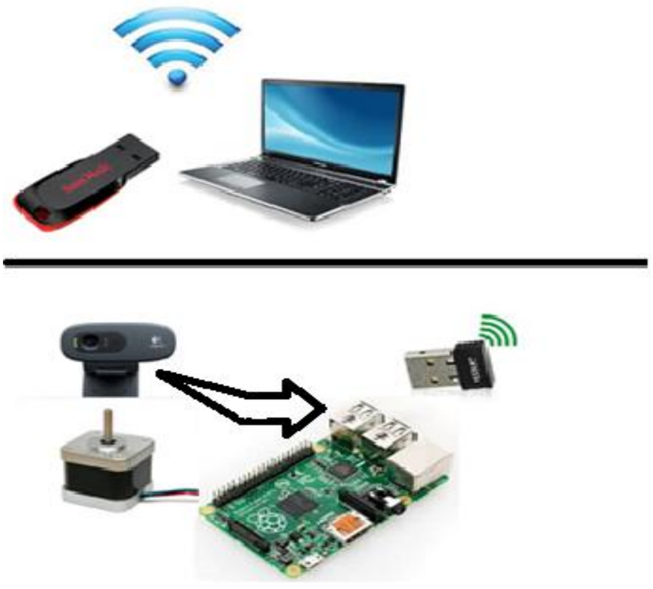
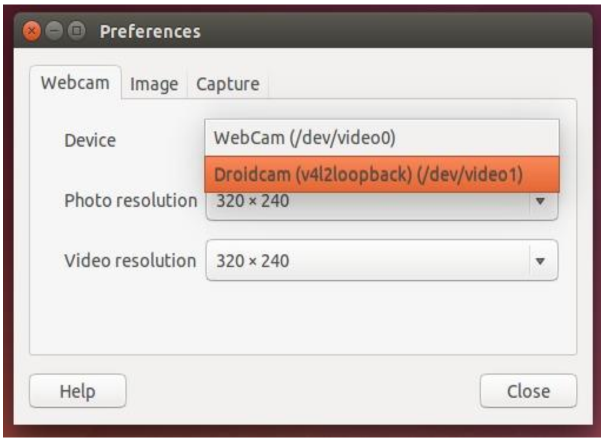


Fig. 9: Interfacing of embedded board with PC remotely

In above image (Fig.9), board is connected to camera and stepper motor both, and also connected to Wi-Fi dongle or Ethernet for video streaming purpose on internet. PC is not directly connected to embedded board but remotely connected by socket programming.

### *F. Camera:*

Last improvement is very crucial for the operation of the system. When webcam rotates, its wire gets wrapped around it. So after 3 or 4 rotations webcam cannot rotate further in one direction. That is not going to create a major problem but sometime user needs more than 3 rotations in one direction. For such applications, a wireless camera can be used. User can use his mobile as wireless camera. There is one software on android named Droidcam [23] is available which turns mobile into a wireless camera. On the other hand user can use Droid client software on Linux to catch that stream. What client software will do? Client [24] creates video driver in Linux kernel which acts as virtual camera. All you have to do is to just open that driver and fetch video stream using OpenCV libraries. This method works perfectly.



Device: WebCam (/dev/video0)
Droidcam (v4l2loopback) (/dev/video1)
Photo resolution: 320 × 240
Video resolution: 320 × 240

Fig. 10 Droidcam client on PC.

## IV. CONCLUSION AND FUTURE WORK

This paper describes object tracking application and its implementation using different designs with rotating camera. This paper also proposes two different algorithms for rotation of the camera according to data given by the object detection algorithm. This paper also describes study and features of different object tracking algorithms in application.

This implementation can be expanded for multiple object tracking as well. With static camera multiple objects can be detected and tracked as long as they are in the line of sight of the camera. But with a moving camera multiple objects can be detected and any one object can be tracked. Multiple cameras can be used to detect and track multiple objects. The cameras can be synchronized to detect and track different objects.

These algorithm can be ported to android application and with high end mobile devices (faster CPUs), they can run flawlessly and one can use mobile to give signal to embedded board to control motor over network.

## REFERENCES

[1] Balachander chokkalingam "Evaluation of TLD (aka Predator) algorithm used for real-time tracking of unknown objects in a video stream from eye tracking perspective" master's thesis in Systems, Control and Robotics at Kungliga Tekniska Högskolan (KTH), Sweden, Dec 2012.

[2] Georg Nebehay and Roman Pflugfelder "Consensus-based Matching and Tracking of Key points for Object Tracking" Applications of Computer Vision (WACV), 2014 IEEE Winter Conference, pp. 862-869, March 2014.

[3] P. J. Burt, J. R. Bergen. R. Hingorani, R. Kolczynski, Y.A. Lee, A. Leung, J. Liibin, H. Slivaytser "OBJECT TRACKING WITH A MOVING CAMERA: An Application of Dynaiiiic Motion Analysis" visual motion workshop, pp 2-12, March 1989.

[4] Kye Kyung Kim, Soo Hyun Cho, Hae Jin Kim and Jae Yeon Lee "Detecting and Tracking Moving Object Using an Active Camera" The 7th international conference on advance communication technology ICACT, pp. 817-820, 2005.

[5] "CMT algorithm for image tracking" http://www.gnebehay.com/cmt/ [15 Apr. 2014].

[6] "CAM shift algorithm" https://github.com/Itseez/opencv/blob/master/samples/cpp/camshiftdemo.cpp [15 Apr. 2014]

[7] "Comparison of embedded system boards" https://learn.adafruit.com/embedded-linux-board-comparison/overview [20 Dec. 2014].

[8] "Stepper motor general information" http://www.engineersgarage.com/articles/stepper-motors [20 Dec. 2014].

[9] "How to install opencv on embedded Linux" http://www.pyimagesearch.com/2015/02/23/install-opencv-and-python-on-your-raspberry-pi-2-and-b/ [15 Apr. 2015].

[10] "How to turn embedded board to IP camera (web video server)" http://pimylifeup.com/raspberry-pi-webcam-server/ [15 Apr. 2015].

[11] "how to take video as input from web server" http://stackoverflow.com/questions/16450102/cvvideocapture-works-for-webcams-but-not-ip-cameras [15 Apr. 2015].

[12] "Python programming using opencv" http://opencv-pythontutroals.readthedocs.org/en/latest/py_tutorials/py_gui/py_video_display/py_video_display.html [15 Apr. 2015].

[13] "Python socket programming concepts" http://www.tutorialspoint.com/python/python_networking.htm [15 Apr. 2015].

[14] "Python serial programming concepts" http://elinux.org/Serial_port_programming [15 Apr. 2015].

[15] "C++ OpenCV programming concepts" http://docs.opencv.org/modules/highgui/doc/reading_and_writing_images_and_video.html [15 Apr. 2015].

[16] "C++ serial library documentation" http://libserial.sourceforge.net/ [15 Apr. 2015].

[17] "How to do wiring of unipolar stepper motor with embedded boards" http://www.instructables.com/id/Arduino-6-wire-Stepper-Motor-Tutorial/ [15 Apr. 2015].

[18] "How to control stepper motor using python" http://www.raspberrypi-spy.co.uk/2012/07/stepper-motor-control-in-python/ [15 Apr. 2015].

[19] "Different embedded Linux OS" http://en.wikipedia.org/wiki/Linux_on_embedded_systems [15 Apr. 2015].

[20] "CAMSHIFT algorithm details" http://www.gergltd.com/cse486/project5/ [15 Apr. 2015].

[21] "Writing video to memory using OpenCV" http://stackoverflow.com/questions/16045654/using-opencv2-write-streaming-video-in-python [15 Apr. 2015].

[22] "Embedded system camera support" http://elinux.org/RPi_USB_Webcams [20 Dec. 2014].

[23] "Droidcam for android" internet: http://www.androidapk.ws/app/droidcam-wireless-webcam-apk [25 Apr. 2015].

[24] "Droid cam client for Linux or windows OS" http://www.dev47apps.com/ [25 Apr. 2015].