eYSIP2016

Object Tracking Camera

Abhishek Rathore
Gopineedi Harsha Vardhan
Mentors:
Akshat Jain
Pushkar Raj
Rama kumar
Duration of Internship: 21/05/2016 – 10/07/2016

2016, e-Yantra Publication

Object Tracking Camera

Abstract

Object tracking is the process of locating a moving object over time using a camera. It has a variety of uses, some of which are: human-computer interaction, security and surveillance, video communication and compression, augmented reality, traffic control, medical imaging and video editing. Object tracking can be a time consuming process due to the amount of data that is contained in each frame.

The objective of Object tracking is to associate target object in consecutive video frames. The association can be especially difficult when the object is moving fast relative to the frame rate. Another situation that increases the complexity of the problem is when the tracked object changes orientation over time.

This system will track an object given through a Region-Of-interest using Real-Time Image Processing.The project starts with learning basic Image processing techniques and Object tracking techniques and in later stages,Advanced tracking techniques are experimented and implemented.

Completion status

The task of detecting and tracking a object using a camera is done **sucessfully**.The tasks of integrating the camera on the autonomous drone and GUI appplication for the system is yet to be done.We completed all of the tasks given to us and uploaded the tutorials on GitHub The Tutorials done are as follows:

- **Colored Object Tracking using HSV:**Tracks a colored ball by taking it HSV color range as its input and tracks the colored object by finding the largest area contour in the frame which will be our object.
- **Object Tracking(Based on ROI):**Tracks any object which is given

in the ROI(taken through mouse) by CAM SHift Algorithm using Histograms and back-projections of the object.

- **Tutorial on Gimbal making:**Explains on how to choose Gimbal and its parts for the drone ,how to select best low-weight parts and designing the gimbal.
- **Re-recognizing the Object:**Re-recognizes the object if it gets lost out of the frame comparing the back-projection of the object with the whole frame and searches for the object when it gets lost
- **Setting Up Raspberry-Pi:**Installing necessary python packages required for tracking and camera packages.Interfacing the servos with the GPIO pins of R-Pi.
- **Object Tracking Camera interfaced on a Raspberry-Pi:**Cam Shift algorithm and Re-recognizing algorithms are implemented on the R-pi which has a interfaced Object tracking camera on it.Thi system tracks any object given through a ROI.

1.1 Hardware parts

- Raspberry Pi B+ Datasheet
- Raspberry-Pi Camera module Datasheet
- 3-D printed Camera mount frame Image
- Tower Pro Micro servo 9g (2 No.) Datasheet
- Connecting wires
- External Power supply

1.2 Software used

- Python 2.7.5
  Python Packages
  1. Open CV 2.4.
  2. numpy 1.7.1
  3. RPIO

4. Picamera

- MobaXterm 9.0

1.3 Assembly of hardware

1.3.1 Setting up Raspberry-Pi

We will be using the Raspberry-Pi remotely from our Laptop/PC using SSH connection by MobaXterm using a LAN Cable or by a Wi-fi module that is setup on the Pi.Make sure that the Pi is connected to the Internet as we need to install necessary packages for further setup.

In our case,we connected to the Pi through LAN by assigning it a static IP from the "cmdline.txt" file on the SD card.We are accessing the Internet through the Wi-fi module.

It is also possible to overclock the R-Pi externally through the "config.txt" file from the SD card or internally through the "/boot/config" file.The need for overclocking is to meet the excessive processing needs of our Gimbal system.(Although it is not recommended as it may cause heating problems).

In our case,we overclocked the R-Pi from 700 Mhz(default) to 900 Mhz.

1.3.2 Setting up Camera(Pi-Cam)

After fixing the Pi-cam in its connector located between the Ethernet port and HDMI port,boot up your Raspberry-Pi and open the terminal and run the following command shown in Fig 1 and enable the camera(if not enabled) and reboot ur R-Pi

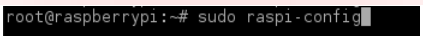
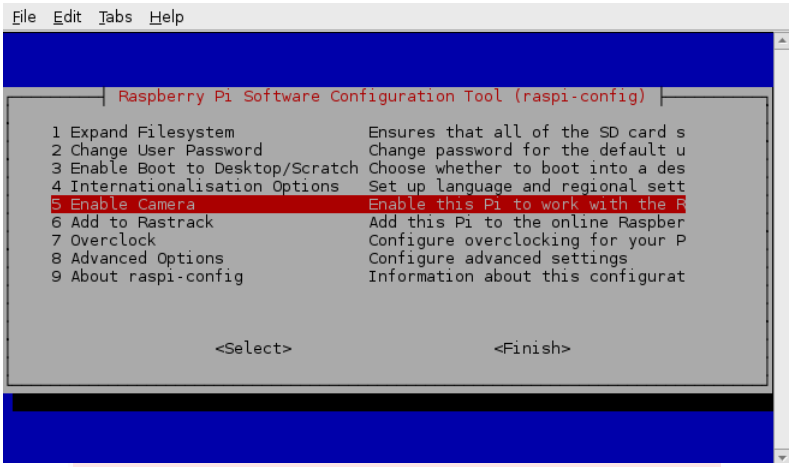


Figure 1.1: Fig 1



We can check the working status of the camera by using "raspistill -o image.jpg" which takes a picture with the camera and stores it in image.jpg.

1.3.3 Setting up the Pan-Tilt System

For making the Pan-Tilt system we will be using Two Tower Pro Servo Motors(Fig 3) each weighing 9g (we used micro servos as they would draw a little amount of current which can be produced by the R-Pi itself)

Figure 1.2: Fig 3

We connected the first servo to the Pin 16(GPIO 23) and second servo to Pin 18(GPIO 24) on the board.We use this pins as signal channels when we use the servos from R-Pi.All the available GPIO ports that can be used are shown in Fig 4.

We calibrated each of the motor's PWM values and assembled to the frame mount which is shown in Fig 5.

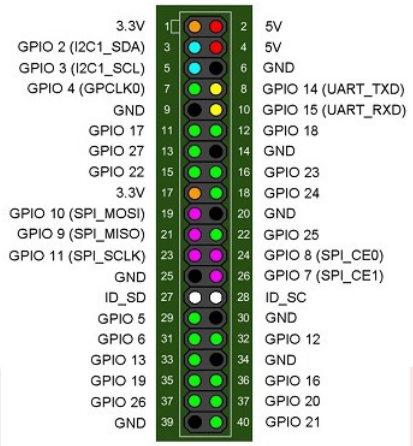


Figure 1.3: Fig 4

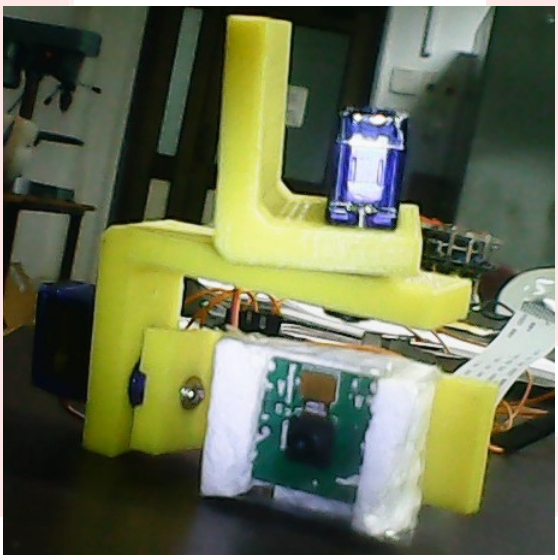


Figure 1.4: Fig 5

1.4 Software and Code

Python codes for different modules are provided below.

- Color Object tracking using HSV color space
- Object Tracking using a ROI
- Re-recognising the object if it goes out of the frame
- Object Tracking using a Pi-Cam interfaced on a Raspberry-Pi

Brief explanation of various parts is done in code.

Interfacing the hardware and other modules can be found in the project page on GitHub

1.5 Use and Demo

- Demo for Colored object tracking using HSV
- Demo for Object tracking using ROI
- Demo for Re-recognizing the object if it escapes from the frame
- Demo for Object tracking camera interfaced on a Raspberry-Pi using servo motors

1.6 Future Work

1.6.1 Hardware

- The Camera mount frame can be further developed into a Gimbal system which can be mounted on a drone
- The servos can be replaced with Brush-less DC motors which are much smoother.
- Camera with much better resolution for high end image processing might be equipped.

1.6.2 Software

- More effective Object tracking algorothms can be used.
    1. Key point matching can be implemented(Contiguous Match Tracker algorithm can be further improved and implemnted here)
    2. Small amount of training can be used for strong tracking of the object
    3. Multiple Tracking methods can be collaborated so that object never gets lost.
    4. Code improvements in present one
- Servo movement can be speeded up by effective algorithms

- PWM controller can be implemented for balancing the gimbal effectively when mounted on a drone.

1.7 Bug report and Challenges

1.7.1 Challenges faced

Hardware:

- Interfacing the servos with the GPIO pins on the Raspberry-Pi and removing their jitters.
- Power requirement of the servos and the camera system from the Raspberry-Pi is high.

Software:

- Re-detection of object when it escapes from the frame
- Movement of camera tracking the object as both camera and object are searching for the same center.
- The track window for CAM shift will be shifted as the camera moves away from the current track window.

1.7.2 Bugs

- The algorithm cannot track high texture objects as it uses HSV color space for back projection .
- The camera is assumed to have the previous track window at its center which is only possible if the servos are working properly.
- The Camera has a low resolution which makes image processing harder and may not provide best results.

1.8 Referred research-papers

- Matrioska Algorithm
- TLD-Tracker

- CAM Shift
- extended CAM Shift
- CAMShift improvised
- CMT

1.9 References

- http://www.pyimagesearch.com/2015/05/25/basic-motion-detection-and-tracking-with-python-and-opencv/
- http://www.theverge.com/2016/3/1/11134130/dji-phantom-4-drone-autonomous-avoidance-tracking-price-video
- https://oscarliang.com/build-brushless-camera-gimbal-handheld-quadcopter/
- https://oscarliang.com/servo-brushless-camera-gimbal-fpv-quadcopter/
- http://www.pyimagesearch.com/2015/09/21/opencv-track-object-movement/
- http://www.openelectrons.com/forums/viewtopic.php?f=4t=6
- https://www.youtube.com/watch?v=PVWyZse9z4E
- https://www.youtube.com/watch?v=C95bngCOv9Q
- https://www.youtube.com/watch?v=W2qR60hrD2w
- https://www.youtube.com/watch?v=1GhNXHCQGsM
- http://blog.christianperone.com/2015/01/real-time-drone-object-tracking-using-python-and-opencv/
- http://ardupilot.org/copter/docs/common-camera-gimbal.html
- https://code.google.com/archive/p/ardupirates/wikis/StabilizedCameraMount.wiki
- http://www.lfd.uci.edu/ gohlke/pythonlibs/scipy
- http://www.instructables.com/id/Pan-Tilt-face-tracking-with-the-raspberry-pi/step4/Connecting-the-servos/
- https://create.arduino.cc/projecthub/junejarohan/ball-tracking-robot-7a9865?ref=tagref$_i d = robotsoffset = 11$