"FACE RECOGNITION AND TRACKING USING OPENCV"

By

Ritvik Mahapatra

BTech Mechatronics Engineering
Course: Project Based Learning – 2
Course Code: MC 3170

Reg.No-199403001

Under the guidance

of

Dr. Manish Rawat

(Associate Professor, Department of Mechatronics Engineering)

Department of Mechatronics Engineering

School of Automobile Mechanical & Mechatronics

Faculty of Engineering

Manipal University Jaipur, India

December 2021

Contents

Abstract

In this project-based learning II, I have decided to make a Motion following / sensitive camera. Which can track the face of a person (any person or a particular face using the algorithm of a Machine Learning model available in OpenCV library) and follow them continuously in a certain pre-defined range. There are several fields of work or various applications where we might require more range of vision and more field of view in our vision for particular process, be it drone camera or production process where a camera can be remotely controlled to view much of the process of production without any need of human assistance or there might be some unreachable places where human presence is impossible or unrequired or may the case be of a CCTV camera which follows the motion in night or just keeps on changing its field of view or it can also be controlled remotely. There are various application where we require to remotely control a device about various DOF, and it should be sensitive to its surrounding or follow a particular object. Thus, with some knowledge of DOF, some sensors, some servos and control of each of element we can make a motion follower camera which can follow an object. Thus, this basic implementation can have various applications in real world problems. This representation (Project) has a main motive to solve security (CCTV) or production process (Object follower) related problems.

Introduction

In today's world, mankind has various needs and requirements. These may range from one to other. But various needs in the domain of manufacturing or production are fulfilled by the implementations of knowledge in physics or mechanics, to a certain extent.

During production process which is automated we may need a certain machine element to move in certain direction or rotate about a particular axis. This is where concept of Degree of Freedom comes.

Manufacturing is just an example, concept of DOF is applicable on most of the machines around us, may it be a fridge's door, a table fan, AC's which have controllable swing area {they may cool the area around by following you through a motion sensor and setting the swing of AC at that angle only} or a CCTV camera which can rotate itself at any angle by following the motion of an object or person.

We will here discuss a motion following camera which detects the person's face using a basic USB Webcam and then by using a python library called as OpenCV we can make our camera detect the face of the person in its vicinity and find the landmarks or coordinates of the face. Using these coordinates or landmarks which are obtained by the camera we can serially transmit those coordinates to the Arduino by using pyserial library, so those coordinates can be communicated to the servo motors by which the camera can follow the center of the face.

References are given as follows:

1. Doxygen overview. (n.d.). Retrieved from https://docs.opencv.org/4.x/d4/db1/tutorial_documentation.html
2. Face Tracker Using OpenCV and Arduino. (n.d.). Retrieved from https://create.arduino.cc/projecthub/shubhamsantosh99/face-tracker-using-opencv-and-arduino-55412e
3. Face Tracking OpenCV, Python, & Arduino. (2020, July 09). Retrieved from https://www.learnrobotics.org/blog/face-tracking-opencv/
4. Face Tracking Using Arduino. (n.d.). Retrieved from https://create.arduino.cc/projecthub/WolfxPac/face-tracking-using-arduino-b35b6b
5. From mind to design in minutes. (n.d.). Retrieved from https://www.tinkercad.com/dashboard?type=circuits&collection=designs

Literature review

In physics, the degrees of freedom (DOF) of a mechanical system is the number of independent parameters that define its configuration or state. It is important in the analysis of systems of bodies in mechanical engineering, structural engineering, aerospace engineering, robotics, and other fields.

The formulae for DOF is given by 6(n-1)-5p1-4p2-3p3-2p4-p5

For our case we have 3 links (n=3) (2 servos and 1 camera link) and p1=2(pair of links whose DOF is 1(reduced by 5) i.e., about x and y axis), therefore our DOF comes out to be 12-2*5=12-10=2

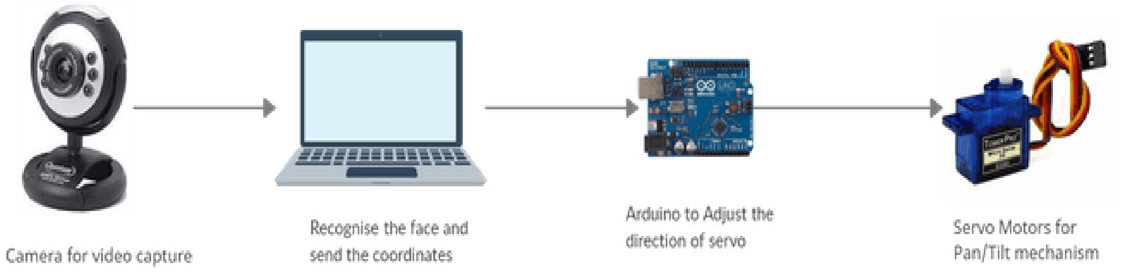
Camera for video capture
Recognise the face and send the coordinates
Arduino to Adjust the direction of servo
Servo Motors for Pan/Tilt mechanism

Fig – 1: Basic working process/Flow

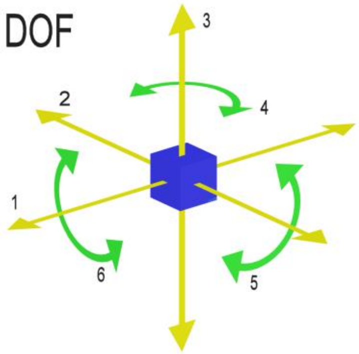

Fig – 2: DOF Visualized in 3-D space

Methodology

Construction and 3-D model

We have two servos, one is fixed with ground (giving 360-degree view about x axis, we will consider for 180 degrees only). The other servo fixed on the rotator of 1st servo, such that their axis is perpendicular to each other (this can rotate about 270 degrees about y axis; however, we will limit it to 180 degrees only).

We have connected/fixed the camera with the help of connections as shown in the figure (initially, along the axis of the first servo). The camera faces the front direction. The DOF of the setup is 2 as it rotates about x-axis and y-axis.

Circuit

The USB Webcam is connected to the PC via the USB cable through which the video data is transmitted to the PC so that the frames can be read by the OpenCV library, by the virtue of which it can recognize a face of a person with the help of a pretrained face recognition model known as ‘Haarcaascade’. After recognizing the face, the landmarks are calculated and transmitted to the Arduino program already uploaded into its ROM, via serial communication. Two of the servos are connected to the Arduino to control the angle of rotation of the camera, respectively.

The ground and the power of all the elements are connected with the Arduino’s gnd and 5v terminal, respectively. The servos signals are received from the pin 5 and 6 of the Arduino.

Code Explanation

<u>Arduino Code:</u>

- Firstly, we have included the servo library for controlling the angle of rotation and declared to two servo’s objects (horizontal and vertical).
- Declare all the required variables by the whole code.
- Void setup () - here we setup the servo signal pins and write initial angles. We also declare
- Void loop () - First, it takes the serial inputs of the landmarks of the face from the python program via pyserial library and some serial communication function included in the serial library of the Arduino program.

- Then it calculates the angle which must be fed to both the servos by using the relation between the landmarks received and the angle of the required. The max range of the angle of rotation of the servo motor has been set to 180 degrees.

- Then it also checks for the angle greater than that of 180 degrees or less than 0 degrees and if those conditions are set to be true then the position of the servo is to be set 0 degrees.

- Then finally after calculating the position/angle of the servo the angle is fed to both the servos with a delay of 100 milliseconds for some required stability.

<u>Python code:</u>

- Firstly, import all the required libraries like OpenCV (cv2) and pyserial.

- Then, the Arduino serial connection is set up and the pretrained face recognition model is imported, and the video capturing is started.

- Inside while loop the frame is extracted from the captured video and the image flipped to avoid the mirror effect. Finally, the cascade model is applied or implemented to each frame with the selected algorithm and some modified parameters.

- Inside the for loop the coordinated are calculated and sent to the Arduino program serially via the afore mentioned library – pyserial, using arduinoSerial.write command.

- Then the video camera's output is displayed frame by frame on a new window and there is also an option of quitting the program by pressing q once.

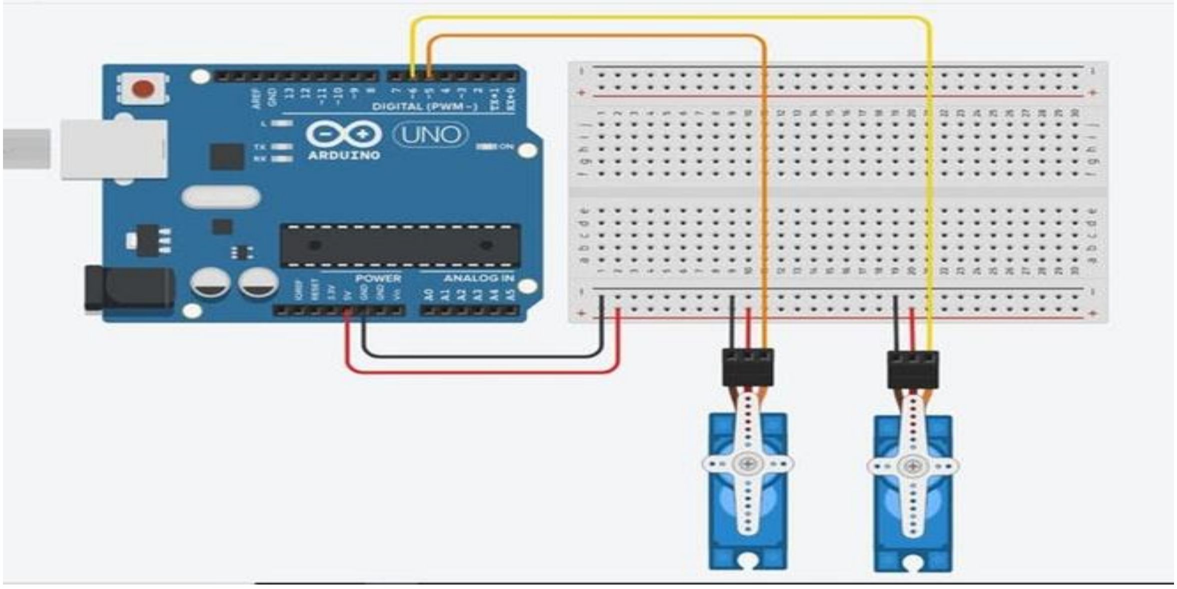

Fig – 3: Circuit Diagram

Results and Discussions

The final outcome of this project is that we can make a camera or any other thing to follow the motion or become motion sensitive. Thus, we can make it to follow any person's motion by recognizing the face of him/her. We have therefore successfully used the knowledge of DOF, Arduino architecture, Working of Servos and Ultrasonic sensors and some coding to achieve the task of following one's motion.

Such devices can be applicable in various field or domain of works. In industry or process automation, Defense applications like Drones or in Precise and stealth Military robots or in basic day to day security applications.

Conclusions and Future Prospects

As far as this project is concerned I would like to call it to a close and start with a bigger, better and a more realistic project in the next PBL, as the university will be functioning at least in hybrid mode from the next semester and if time permits I would like to combine the project of PBL-1 and PBL-2, so as to enable it to toggle between the 2 modes of using camera to detect a person's face or to detect an object in the vicinity of the ultrasonic sensors.

References

- Doxygen overview. (n.d.). Retrieved from https://docs.opencv.org/4.x/d4/db1/tutorial_documentation.html
- Face Tracker Using OpenCV and Arduino. (n.d.). Retrieved from https://create.arduino.cc/projecthub/shubhamsantosh99/face-tracker-using-opencv-and-arduino-55412e
- Face Tracking OpenCV, Python, & Arduino. (2020, July 09). Retrieved from https://www.learnrobotics.org/blog/face-tracking-opencv/
- Face Tracking Using Arduino. (n.d.). Retrieved from https://create.arduino.cc/projecthub/WolfxPac/face-tracking-using-arduino-b35b6b
- From mind to design in minutes. (n.d.). Retrieved from https://www.tinkercad.com/dashboard?type=circuits&collection=designs

Acknowledgement

The success and outcome of this project required a lot of guidance and assistance from many people, and I am extremely fortunate to have got this all along the completion of our project work. Whatever I have done is only due to such guidance and assistance and we would not forget to thank them. I respect and thank Dr Manish Rawat (Associate Professor Mechatronics Engineering) for giving we an opportunity to do this project work and providing all the support and guidance which made me complete the project on time, His dynamism, vision, sincerity and motivation have deeply inspired me. He has taught me the methodology to carry out the research and to present the research works as clearly as possible. It was a great privilege and honour to work and study under his guidance. I am extremely grateful to him for providing such a nice support and guidance. I am grateful because I managed to complete this project within the time given by my professor. I am extending my thanks to the Department of Mechatronics, Manipal University, Jaipur for their support during my research work. I also thank all the staff of the university for their kindness. I am extremely grateful to my parents for their love, prayers, caring and sacrifices for educating and preparing me for my future. Finally, I would like to express our gratitude to our friends and respondents for support and willingness to spend some time with us.