Hindi Vidya Prachar Samiti's

RAMNIRANJAN JHUNJHUNWALA COLLEGE

(AUTONOMOUS)

GHATKOPAR (W), MUMBAI – 400 086.

Certificate

This is to certify that ***Mr. Md Rahmatulla*** *Roll No* ***33*** of ***M.Sc. (I.T.)-P1*** class has completed the Project on ***Media Player Controlling by hand gestures*** in the Department of Information Technology during the academic year 2022 - 2023.

Professor In-Charge ***Co-ordinator of IT Department***

Prof. Bharti Bhole

College Seal & Date ***Examiner***

ACKNOWLEDGEMENT

The four things that go on to make a successful Endeavour are dedication, hard work, patience and correct guidance. Able and timely guidance not only helps in making an effort fruitful but also transforms the whole process of learning and implementing into an enjoyable experience

In particular, I would like to thank our principal **Dr. (Mr.) Himanshu Dawda** & R.J College.

I would like to give a very special honour and respect to our teacher, **Prof. Bharti Bhole** who took keen interest in checking the minute details of the research work and guided us throughout the same. A sincere quote of thanks to our Head of Department **Prof. Bharti Bhole** for Valuable guidance also the non-teaching staff for providing us software & their time. I appreciate outstanding co-operation by them, especially for the long Lab timings that we could receive.

A Special thanks to everyone who has helped us in making the research work.

Md Rahmat

ABSTRACT

The computer is becoming more important in our daily life with the development of ubiquitous computing. Computer applications require interaction between human and computer. This interaction needs to be unrestricted and it had made challengeable to traditional input devices such as keyboard, mouse, pen etc. Hand gesture is used in people's daily life frequently. Hand gesture is an important component of body languages in linguistics. They are more natural in interaction, compared with those devices mentioned above. Human computer interaction becomes easy with use of hand as a device. Use of hand gestures to operate machine would make interaction interesting. Gesture recognition has gained a lot of importance. Hand gestures are used to control various applications like windows media player, robot control, gaming etc. Use of gesture makes interaction easy, convenient and does not require any extra device. Vision and audio recognition can be used together. But audio commands may not work in noisy environment. In this paper the idea to use hand gestures to control windows media player.

INDEX

1. INTRODUCTION

Everyone is dependent to perform most of their tasks using computers. The major input devices are keyboard and mouse. But there are a wide range of health problems that affects many people, caused by the constant and continuous work with the computer. Direct use of hands as an input device is an attractive method for Human Computer Interaction Since hand gestures are completely natural form for communication so it does not adversely affect the health of the operator as in case of excessive use of keyboard and mouse. The User interface has a good understanding of human hand gestures. By using the gesture, Feelings and thoughts can also be expressed. Users generally use hand gestures to express their feelings and notifications of their thoughts. Hand gesture and hand posture are related to the human hands in hand gesture recognition. In this paper we are going to present an application which uses dynamic hand gestures as input to control the windows media player. We have considered single handed gestures and their directional motion defines a gesture for the application. In this application image acquisition is done using a Webcam. Some functions in windows media players are used more frequently and thus applying controls windows media player for those functions using predefined gestures. Fig. 1 shows the defined gestures according to the windows player control function.

2. AIMS AND OBJECTIVES

There are a lot of critical situations in the day-to-day lives of disabled people. We have come up with a small part of such life to make it easier using computer vision technology. In this paper, we have discussed a low system that uses hand gesture recognition technology to control the VLC media player. Among many computer visions based interactive systems, designing hand gestures and facial expression based HCI system retains to be a highly challenging task. Our main purpose is to find a nontangible way to interact with the computer.

Our project aims at modifying the existing video player controlled by human hand gestures by making use of Convex Hull and OpenCV.

- ✓ To minimize the use of keyboard and mouse in computer.
- ✓ To integrate gesture recognition features into any computer at a low cost.
- ✓ To help in the development of a non-tangible way to interact with the video player.

3. SCOPE OF THE PROJECT

The Hand Gesture recognition is moving at tremendous speed for the futuristic products and services and major companies are developing technology based on the hand gesture system and that includes companies like Microsoft, Samsung, Sony and it includes the devices like Laptop, Hand held devices, Professional and LED lights. The use and adoption will become more cost effective and cheaper. It's a brilliant feature turning data into features with mix of technology and Human wave. Smart phones have been experiencing enormous amount of Gesture Recognition Technology with look and views and working to manage the Smartphone in reading, viewing and that includes what we call touch less gestures. In the medical fields Hand Gesture may also be experienced in terms of Robotic Nurse and medical assistance. As the Technology is always revolving and changing the future is quiet unpredictable but we have to be certain the future of Gesture Recognition is here to stay with more and eventful and Life touching experiences.

4. PROPOSED SYSTEMS

In this system we have used different image processing techniques, feature extraction and classification tool for recognizing the gesture in real time and appropriate command to the windows media player.

- ✓ Data acquisition: Done by in built webcam on the laptop.
- ✓ Segmentation: there are two types of techniques used. Skin detection model for detection of hand region.
  - Skin detection model for detection of hand region.
  - Approximate median technique for subtraction of background.
- ✓ Recognition phase: decision tree was used as a classification tool.
- ✓ Windows interaction: give the appropriate command to the windows player according to the recognised gesture.

5. SYSTEM REQUIREMENTS

Software Requirements

- Operating System: Windows 2000, XP, 7/8/8.1/10
- Python IDLE
- Libraries : OpenCV, Numpy, pyautogui

Hardware Requirements

- Processor: Any processor above 500 MHz
- RAM: 1GB
- Hard Disk: 100GB

6. ANALYSIS

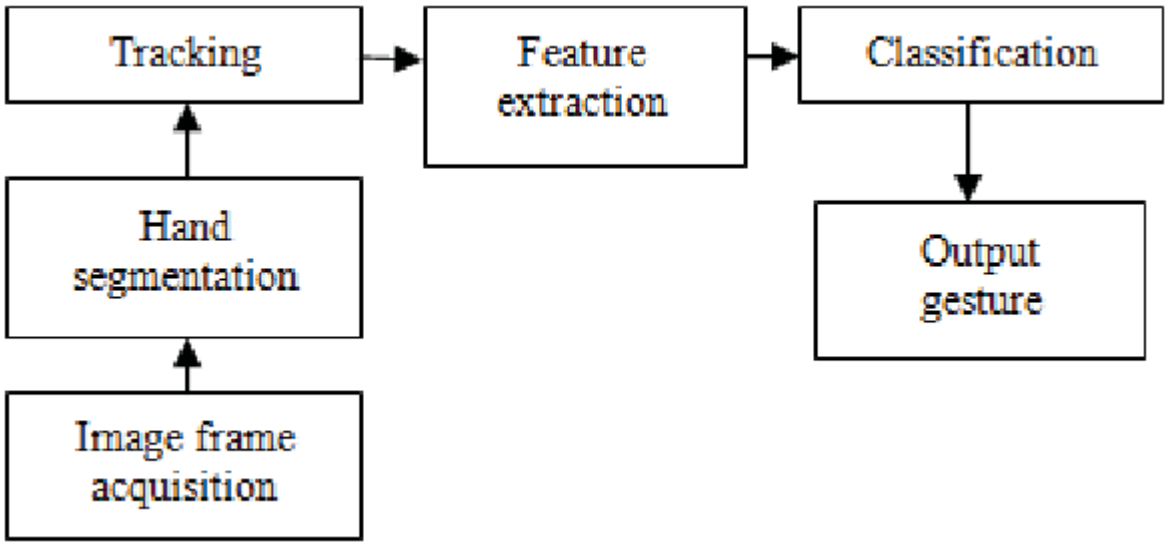


Fig. 1 : Basic Architecture of the Project

Figure 1 is the basic structure that we plan to achieve where the simple human hand gestures will be considered as a control input for the video player to perform actions based on the appropriate mapping of the gestures along with their meaningful control actions (play, pause, volume up, volume down, resume).

7. DESIGN

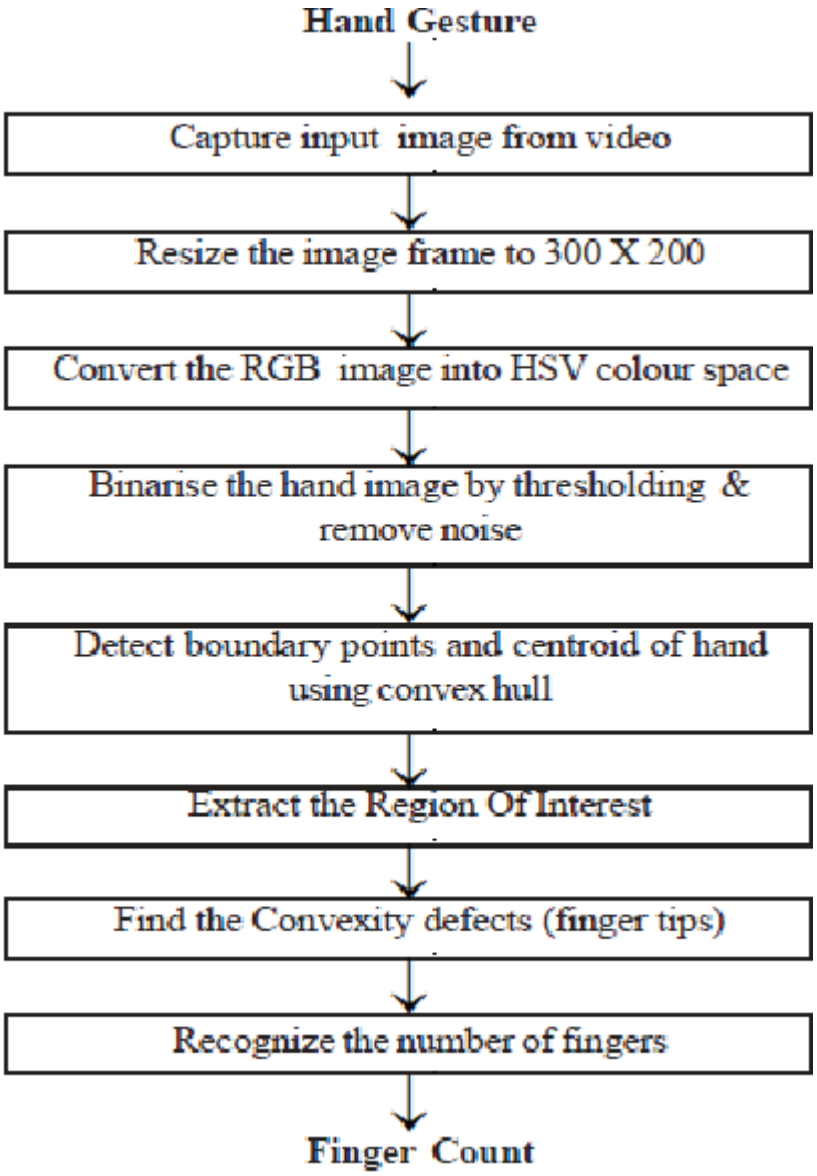


Fig. 2 : Flowchart of the Image Detection

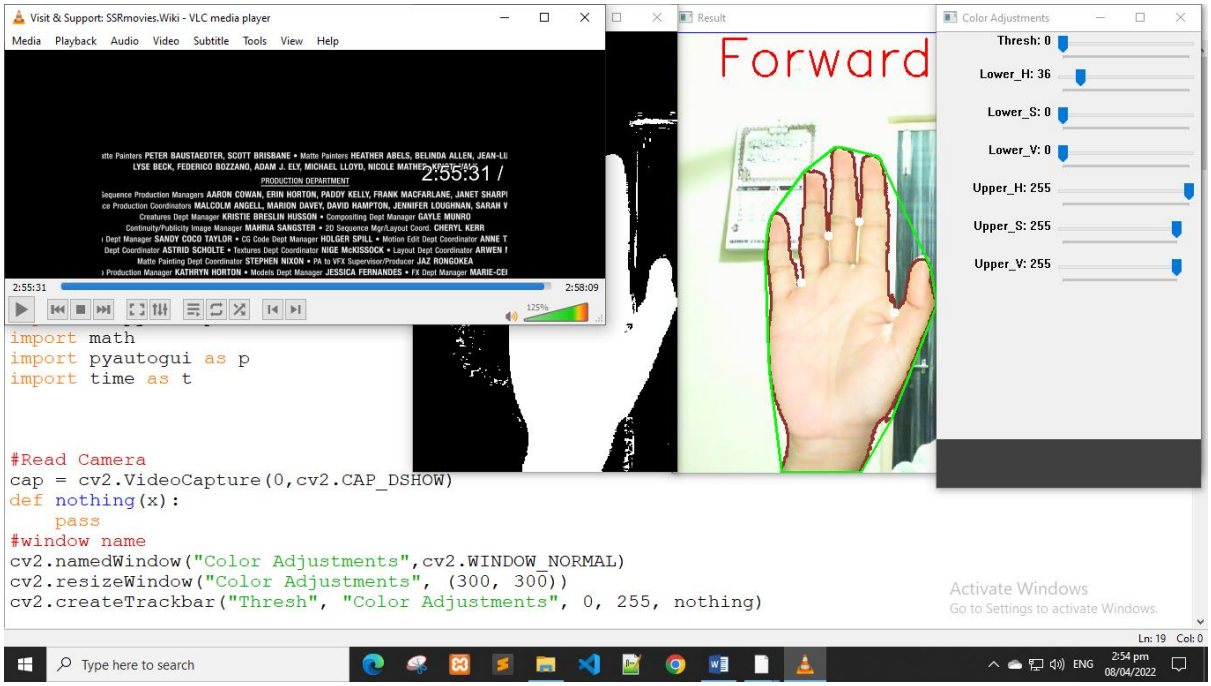
The steps for Image Processing is been represented using Figure 2 which consists of eight steps such as capturing, resizing, converting, thresholding, contouring, extracting, detecting then finally recognizing.

8. IMPLEMENTATION

The design for the proposed system is shown in Figure 3 The images and their masked labels are in .png format. The images are being captured from the live video stream and these images are further processed to achieve efficiency of the detection stage. In the detection stage, initially, contours are identified for the images and then these contours are used for the creation of the Convex Hull around the hand

region. Then these image outputs are passed as control commands to the VLC media player to perform the associated action.

It is clear that, using three dimensions matrix pixel values (RGB values) for all the images in it crucial to distinguish between these images. Semantic Segmentation technique where the objective is to find different regions in an image and tag its corresponding labels, for this first step includes segmenting only the hand region from the live video stream and identify different possible gestures. The convex hull cluster is of peaks that cover the region of the hand. Here, we must clear the principle of the convex set, which means all lines between any 2 points within the hull are entirely within it. After determining the gesture, the specific functioning is performed. The method of recognizing the movement is a dynamic process. After operating the specific command from the gesture, go back to the initial step to accept other images to be processed, and so on.

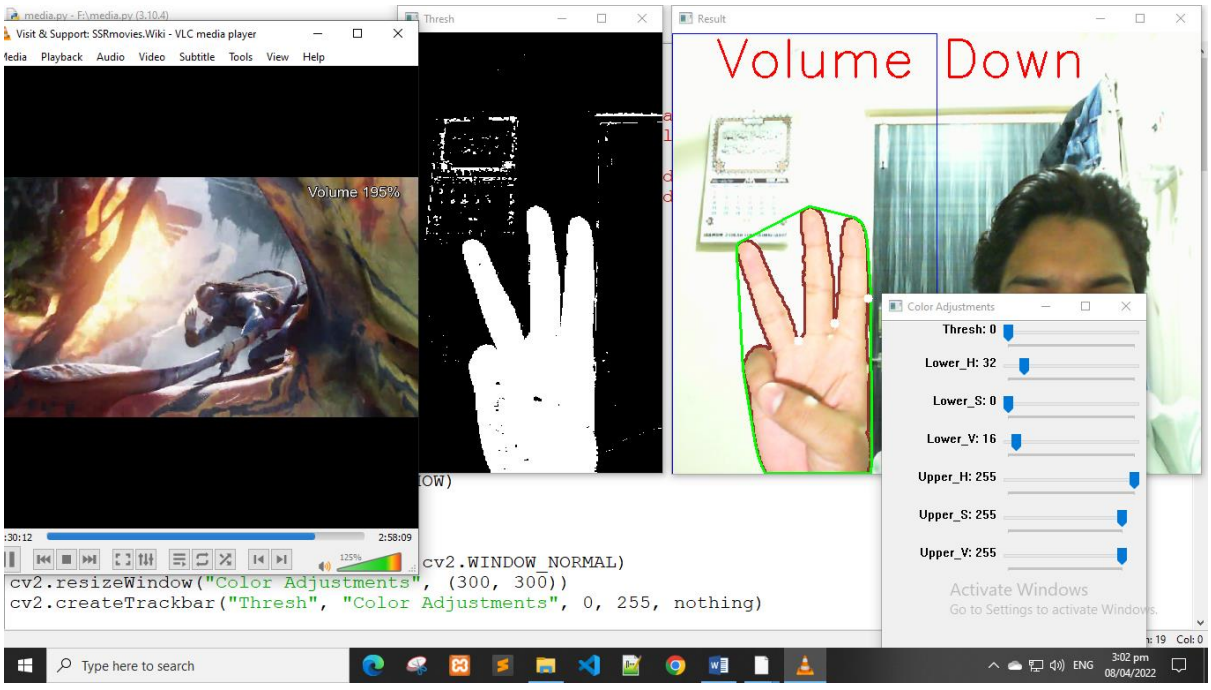


After this the gesture pictures were segregated into labelled folders as play, pause, volume up, volume down and resume appropriately. Exploratory Image details Analysis is an approach for analysing details related to images to summarize their main characteristics, often with visual methods. Much complex visualization can be achieved with matplotlib and usually, there is no need to import other libraries. This is used for gaining a better understanding of data aspects like: -main features of data -variables and relationships that hold between them -

identifying which variables are important for our problem Next we plan to input this processed images as input into the VLC media player and perform the necessary actions.

9. RESULTS

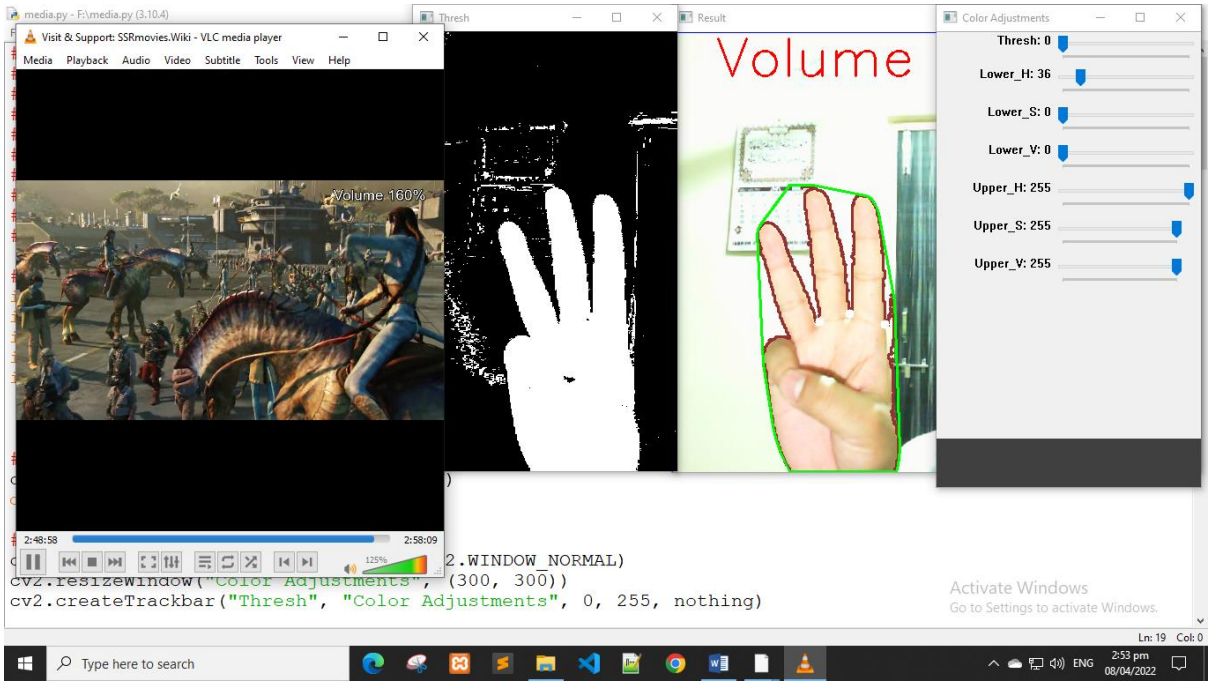
Depending on the contour defects and making use of OpenCV techniques the gestures are identified and classified into different gestures and there are visual threshold and color representations in the result. During the live video, streaming frames are extracted, processed, and converted into proper images along with region segmentation then these image outputs are supplied as input into the VLC video player to perform the associated action command related to VLC. Here we have implemented and restricted our project only for 4 gestures further scope would be to increase the number of gesture recognition and commands for the VLC video player.



The system is successful in detecting the hand gesture for play and recognizing the action to be performed, so the corresponding action of playing the video is functioned.

The system is successful in detecting the hand gesture for pause and recognizing the action to be performed, so the corresponding action of pausing the video is functioned.

Media Player Controlling by hand gestures



The system is successful in detecting the hand gesture for Volume down and recognizing the action to be performed, so the corresponding action of decreasing the volume for the video is functioned.

The system is successful in detecting the hand gesture for Volume up and recognizing the action to be performed, so the corresponding action of increasing the volume for the video is functioned.

The system is successful in detecting the hand gesture for Resume and recognizing the action to be performed, so the corresponding action of resuming the video is functioned.

10. SOURCE CODE

#Step - 1 -Import Libraries and capture camera

```
import cv2
import numpy as np
import math
import pyautogui as p
import time as t


#Read Camera
cap = cv2.VideoCapture(0,cv2.CAP_DSHOW)
def nothing(x):
    pass
#window name
cv2.namedWindow("Color Adjustments",cv2.WINDOW_NORMAL)
cv2.resizeWindow("Color Adjustments", (300, 300))
cv2.createTrackbar("Thresh", "Color Adjustments", 0, 255, nothing)
```

Media Player Controlling by hand gestures

#Step - 2 -Convert frames Into hsv

```
hsv = cv2.cvtColor(crop_image, cv2.COLOR_BGR2HSV)
 #detecting hand
 l_h = cv2.getTrackbarPos("Lower_H", "Color Adjustments")
 l_s = cv2.getTrackbarPos("Lower_S", "Color Adjustments")
 l_v = cv2.getTrackbarPos("Lower_V", "Color Adjustments")

 u_h = cv2.getTrackbarPos("Upper_H", "Color Adjustments")
 u_s = cv2.getTrackbarPos("Upper_S", "Color Adjustments")
 u_v = cv2.getTrackbarPos("Upper_V", "Color Adjustments")
```

#Step - 3 -Track hand on color basis

```
 lower_bound = np.array([l_h, l_s, l_v])
 upper_bound = np.array([u_h, u_s, u_v])
```

#Step - 4 -Create mask on the basis of color and filter actual color

```
 #Creating Mask
 mask = cv2.inRange(hsv, lower_bound, upper_bound)
 #filter mask with image
 filtr = cv2.bitwise_and(crop_image, crop_image, mask=mask)
```

#Step - 5 -Invert pixel value and then enchance the result for better output

```
 mask1  = cv2.bitwise_not(mask)
 m_g = cv2.getTrackbarPos("Thresh", "Color Adjustments") #getting track bar value
 ret,thresh = cv2.threshold(mask1,m_g,255,cv2.THRESH_BINARY)
 dilata = cv2.dilate(thresh,(3,3),iterations = 6)
```

#Step - 6 -Find Contours for specific colored object

```
#findcontour(img,contour_retrival_mode,method)
 cnts,hier = cv2.findContours(thresh,cv2.RETR_TREE,cv2.CHAIN_APPROX_SIMPLE)


 try:
    #print("try")
```

#Step - 7 -Find Max area contour and draw it on live feed

```
# Find contour with maximum area
     cm = max(cnts, key=lambda x: cv2.contourArea(x))
     #print("C==",cnts)
     epsilon = 0.0005*cv2.arcLength(cm,True)
     data= cv2.approxPolyDP(cm,epsilon,True)

     hull = cv2.convexHull(cm)

     cv2.drawContours(crop_image, [cm], -1, (50, 50, 150), 2)
     cv2.drawContours(crop_image, [hull], -1, (0, 255, 0), 2)
```

Media Player Controlling by hand gestures

#Step - 8 -Find Convexity detect for counting Values and Apply Cosin method

```
    # Find convexity defects
    hull = cv2.convexHull(cm, returnPoints=False)
    defects = cv2.convexityDefects(cm, hull)
    count_defects = 0
    #print("Area==",cv2.contourArea(hull) - cv2.contourArea(cm))
    for i in range(defects.shape[0]):
        s,e,f,d = defects[i,0]

        start = tuple(cm[s][0])
        end = tuple(cm[e][0])
        far = tuple(cm[f][0])
        #Cosin Rule
        a = math.sqrt((end[0] - start[0]) ** 2 + (end[1] - start[1]) ** 2)
        b = math.sqrt((far[0] - start[0]) ** 2 + (far[1] - start[1]) ** 2)
        c = math.sqrt((end[0] - far[0]) ** 2 + (end[1] - far[1]) ** 2)
        angle = (math.acos((b ** 2 + c ** 2 - a ** 2) / (2 * b * c)) * 180) / 3.14
        #print(angle)
        # if angle <= 50 draw a circle at the far point
        if angle <= 50:
            count_defects += 1
            cv2.circle(crop_image,far,5,[255,255,255],-1)

    print("count==",count_defects)
```

#Step - 9 -Bind hand gestures with keyboard keys.

```
    # Print number of fingers
    if count_defects == 0:

        cv2.putText(frame, " ", (50, 50), cv2.FONT_HERSHEY_SIMPLEX, 2,(0,0,255),2)
    elif count_defects == 1:

        p.press("space")
        cv2.putText(frame, "Play/Pause", (50, 50), cv2.FONT_HERSHEY_SIMPLEX,
2,(0,0,255), 2)
    elif count_defects == 2:
        p.press("up")

        cv2.putText(frame, "Volume UP", (5, 50), cv2.FONT_HERSHEY_SIMPLEX,
2,(0,0,255), 2)
    elif count_defects == 3:
        p.press("down")

        cv2.putText(frame, "Volume Down", (50, 50), cv2.FONT_HERSHEY_SIMPLEX,
2,(0,0,255), 2)
    elif count_defects == 4:
        p.press("right")

        cv2.putText(frame, "Forward", (50, 50), cv2.FONT_HERSHEY_SIMPLEX,
2,(0,0,255), 2)
    else:
        pass
```

```
    except:
        pass
```

#Step -10 -Enjoy your output

```
    cv2.imshow("Thresh", thresh)
    #cv2.imshow("mask==",mask)
    cv2.imshow("filter==",filtr)
    cv2.imshow("Result", frame)

    key = cv2.waitKey(25) &0xFF
    if key == 27:
        break
cap.release()
cv2.destroyAllWindows()
```

11. PLANS FOR FUTURE

As per our plans, we have completed our project implementation. Next, we may aim to come with more gestures and their associated actions to be performed for building a completely modified VLC media player. A considerable amount of research yet needs to be done in this field as we need to take latency into consideration. The final aim is to reduce the project response time in real life as even a second delay could cause different control to be functioned.

12. CONCLUSION

The gesture would serve as the direct command for operations such as play or pause the video based on the user's gestures onto the screen. So people don't have to learn machine-like skills which are a burden most of the time, but by contrast, people need only to remember a set of gestures to control the video playback. The Hand Gesture recognition is moving at tremendous speed for the futuristic products and services and major companies are developing a technology based on the hand gesture system and that includes companies like Microsoft, Samsung, Sony and it includes the devices like Laptop, Handheld devices, Professional and LED lights. Smartphones have been experiencing an enormous amount of Gesture Recognition Technology with look and views and working to manage the Smartphone in reading, viewing and that includes what we call touch less gestures.

13. REFERENCES

1. Youtube Tutorial : https://www.youtube.com/watch?v=9zqgvsfbz8M
2. Github Source : https://github.com/askitlouder/Media-Player-Controlling-by-HandGestures-using-OpenCV-and-Python
3. Research Gate : https://www.researchgate.net/publication/49587944_A_Vision_based_Hand_Gesture_Interface_for_Controlling_VLC_Media_Player