

## Janardhan Bhagat Shikshan Prasarak Sanstha’s

**CHANGU KANA THAKUR ARTS, COMMERCE AND SCIENCE**

## COLLEGE

**NEW PANVEL(Autonomous)**

## PROJECT ON

**Gesture Based presentation using OpenCV**

## DEVELOPED BY

**Ms. Vakhardar Siddhi Sanjay**

## UNDER THE GUIDANCE OF

**Prof. Ms. Vrinda Patil**

## ACADEMIC YEAR

### 2022-2023



**Department of Computer Science**

### CERTIFICATE

#### This is to certified that the project entitled

**“Gesture Baesd presentation using OpenCV”**

Is successfully completed by **Ms. Vakhardar Siddhi Sanjay,** Roll No: 54, ExaminationNo:**BSCCS5062** under the guidance of **Prof. Ms. Vrinda Patil** ,during the academic period of 13th June 2022 to2nd Nov 2022 as per the syllabus, fulfillment for the completion of the BCS degree in the Computer Science. It is also to certify that this is original work of the candidate done during academic year 2022 - 2023.

**Place:** Panvel

**Date:**

#### InternalExaminer Principal

**External Examiner Head of Department**



# ACKNOWLEDGEMENT

Itisindeedamatterofgreatpleasureandproudprivilegetobeabletopresentthisprojecton“**Gesture Based presentation using OpenCV**”

I would also like to express my deep regards and the gratitude towards the principal **Prof. Dr. Sanjay Patil.**

I respect and thank Head of the department **Prof. Mrs.Pratibha Jadhav**, for providing me an opportunity to do theproject work and giving me all the support and guidance.Also,Iwouldliketotenderoursincerethanktoalltheteachersfortheir co-operation.

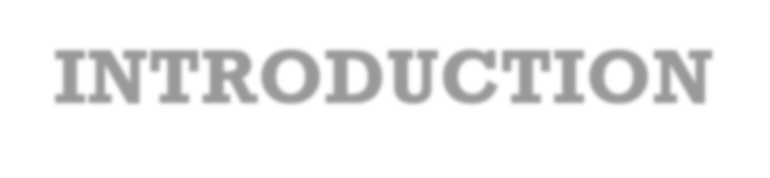
The completion of the project work is a mile stone in student ife and it execution is inevitable in the hands of guide. I am highly indebted the projects guide **Prof. Ms. Vrinda Patil** for his invaluable guidance and appreciation for givingform and substance to this report. It is due to his enduringefforts; patience and enthusiasm, which has given a sense of direction and purposefulness to this project and ultimately made it a Success.

I would Wish to thank the non - teaching staff and myfriendswhohavehelpedme.

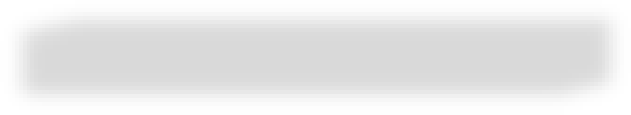
Really it is highly impossible to repay the dept of all thepeoplewhohavemeallthetimeinonewayortheotherdirectlyorindirectlyhelpedmeforperformingtheproject.

# INDEX

|  |  |  |
| --- | --- | --- |
| **Sr.No.** | **Title** | **PageNo.** |
| **01** | **Introduction** | **08** |
| **02** | **Proposed System** | **09** |
| **03** | Existing System | **09** |
| **04** | **Working Of System** | **10** |
| **05** | **Requirement Specification** | **14** |
| **06** | **System Design Details** | **16** |
| **07** | **System Implementation** | **19** |
| **08** | **Results** | **23** |
| **09** | **Conclusion and future Scope** | **27** |
| **10** | **Plagarism Report** | **29** |
| **11** | **References** | **31** |



**INTRODUCTION**



## Introduction:

Gestures are an important aspect of human interaction, both interpersonally context of man-machine interfaces. A gesture is a form of non-verbal communication in which visible bodily actions communicate particular messages, either in place of speech or together and in parallel with words.

Gestures include movement of the hands, face, or other parts of the body.

Hand gesture recognition is one obvious way to create useful and highly adaptive interface between machines and users. This project aims at exploring new way of human-computer interaction. In this project we track the user hand movements and appropriately produce intended actions on the computer.

The project is using python’s hand tracking module and open-cv package to determine hand gestures which will trigger some events like slide change in the presentation window. During this project for gesture recognition, the human body’s motions are read by computer camera.

The computer then makes use of this data as input to handle applications. the target of this project is to develop an interface which will capture human hand gesture dynamically and will control the presentation window.

## 2. Proposed System:

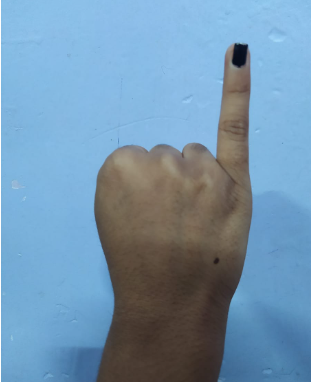
### Most gesture recognition methods usually contain three major stages. The first stage is the object detection. The target of this stage is to detect hand objects in the digital images or videos. Many environment and image problems are needed to solve at this stage to ensure that the hand contours or regions can be extracted precisely to enhance the recognition accuracy. Common image problems contain unstable brightness, noise, poor resolution and contrast. The better environment and camera devices can effectively improve these problems. However, it is hard to control when the gesture recognition system is working in the real environment or is become a product. Hence, the image processing method is a better solution to solve these image problems to construct an adaptive and robust gesture recognition system. The second stage is object recognition. The detected hand objects are recognized to identify the gestures. At this stage, differentiated features and effective classifiers selection are a major issue in most researches. The third stage is to analyze sequential gestures to identify users’ instructs or behaviors.

### 3. Existing System:

### In recent decades, due to computer software and hardware technologies of continuous innovation and breakthrough, the social life and information technology have a very close relationship in the twenty-first century. In the future, especially the interfaces of consumer electronics products (e.g. smart phones, games and infota1inment systems) will have more and more functions and be complex. How to develop a convenient human-machine Interface (Human Machine Interaction/Interface, HMI) for each consumer electronics product has become an important issue. The traditional electronic input devices, such as mouse, keyboard, and joystick are still the most common interaction way. However, it does not mean that these devices are the most convenient and natural input devices for most users. Since ancient times, gestures are a major way for communication and interaction between people.

### 4.Working Of System:

This system takes input from webcam and detects one hand at a time. After detection of hand, system the checks for presence of gestures and if it senses gestures then based on the nature of the gesture appropriate action will be performed.



**Fig 1:Gesture for next slide**

When gesture in Fig 1 is detected,the system will respond by navigating to next slide.



**Fig 2:Gesture for previous slide.**

When gesture in Fig 2 is detected,the system will respond by navigating to previous slide.



**Fig 3:Gesture for pointer functionality.**

When gesture in Fig 3 is detected,the system will respond by providing a pointer on the slide to highlight what the user is pointing at.



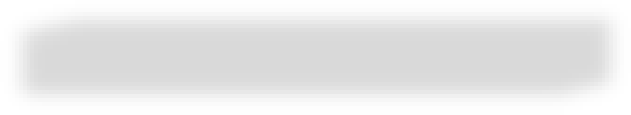
**Fig 4:Gesture for drawing on the slides.**

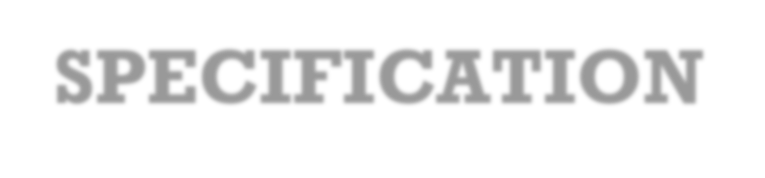
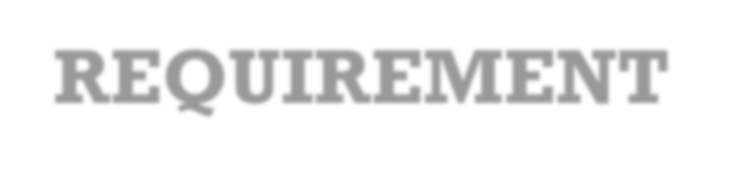
When gesture in Fig 4 is detected,the system will respond by providing a pointer that can draw on slides.

s

**Fig 5:Gesture for erasing the drawing from the slides.**

When gesture in Fig 5 is detected,the system will respond by erasing the drawn content on the slide.





**REQUIREMENT SPECIFICATION**

## 5.Requirement Specification:

**Requirement Analysis:**

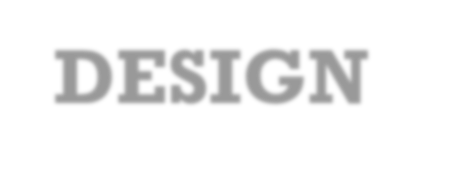
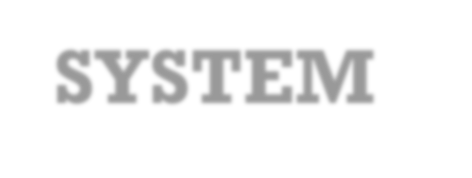
**Hardware Requirements (Used for development)**

* Operating System: Windows 10
* Processor: Intel I3
* System Type: 64-bit operating system, x64-basedprocessor
* Installed Ram: 8 GB
* Web cam (For real-time hand Detection)

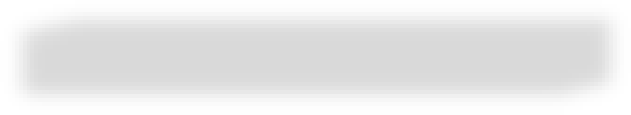
**Software Requirements:**

Software used to execute this project is:

* Python: Python is an interpreter, high-level, general-purpose programming language. Created by Guido van Rossum and first released in 1991.It supports multiple programming paradigms, including structured (particularly, procedural), object-oriented, and functional programming.
* OpenCV: OpenCV (Open Source Computer Vision Library) is a library of programming functions mainly aimed at real-time computer vision. Originally developed by Intel, it was later supported by Willow Garage then Itself.
* Numpy:Numpy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.



**SYSTEM DESIGN**

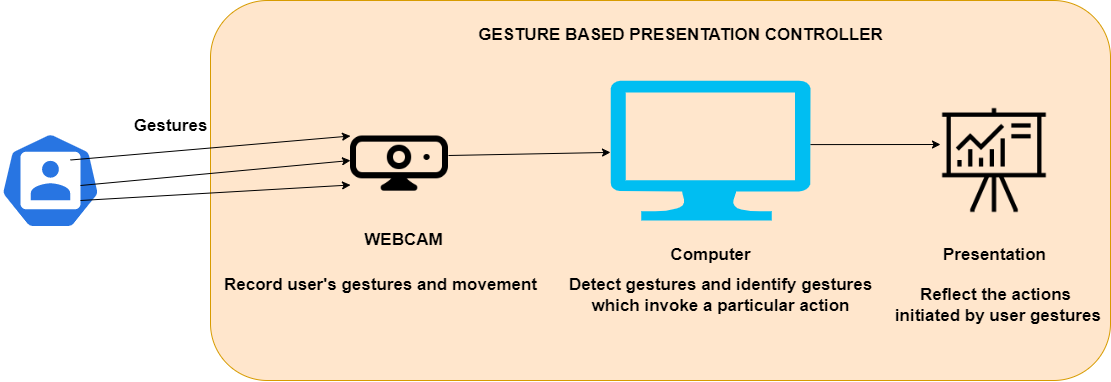


## System Design Details:

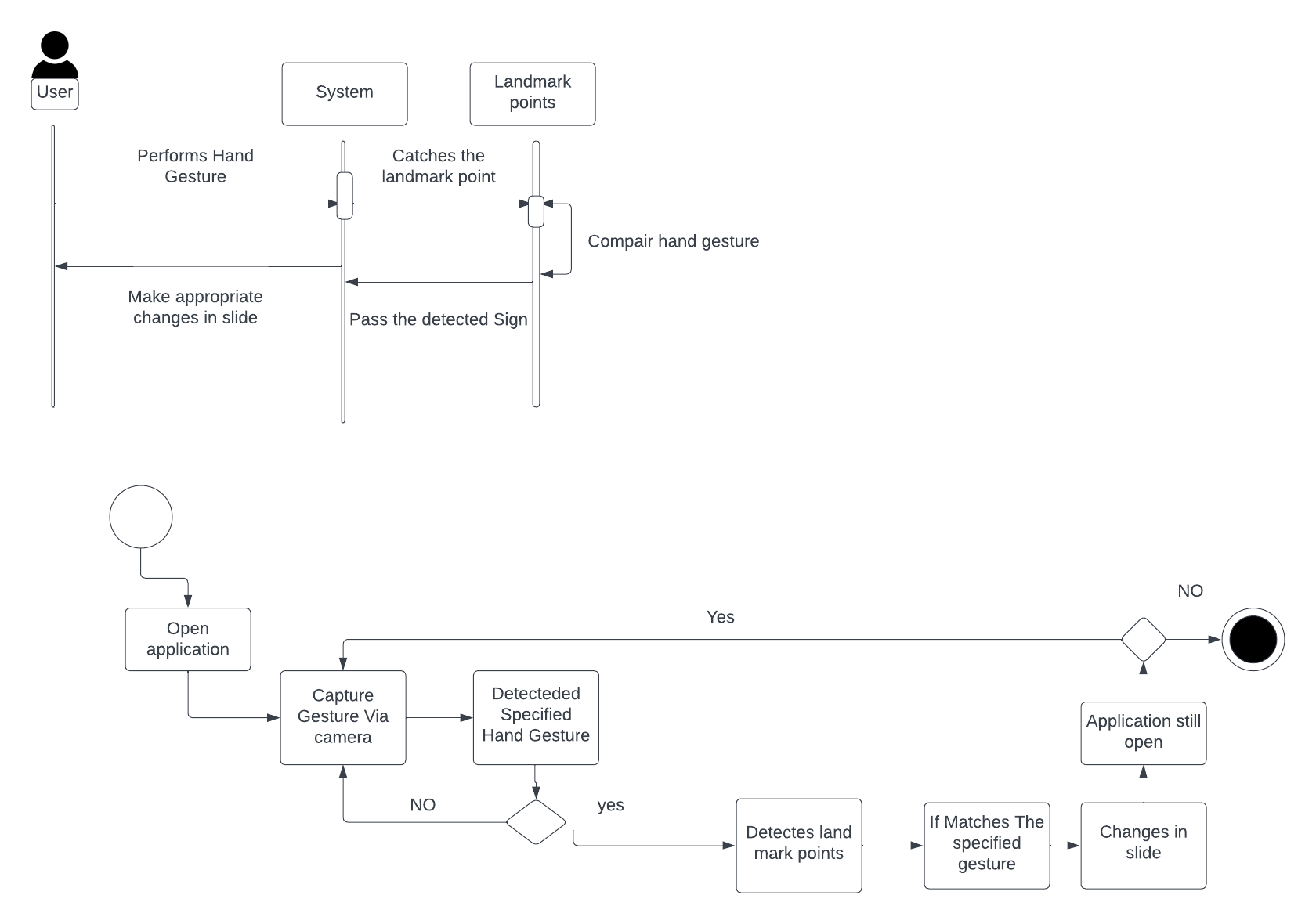
Use Case Diagram

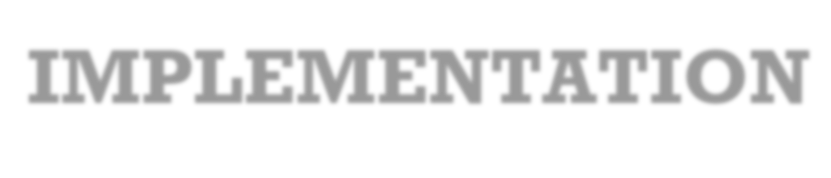
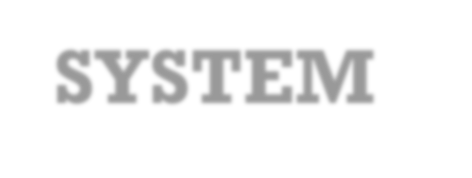
## .

Sequence Diagram

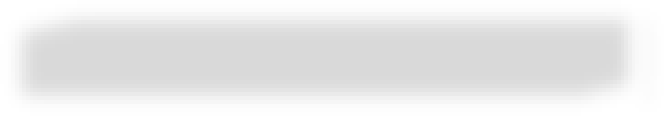


Activity





**SYSTEM IMPLEMENTATION**



## System Implementation:

from cvzone.HandTrackingModule import HandDetector

import cv2

import os

import numpy as np

# Variables

width, height = 1280, 720

gestureThreshold = 300

folderPath = "image"

#Camera Setup

Cap = cv2.VideoCapture (0)

cap.set (3, width)

cap.set (4, height)

detectorHand = HandDetector(detectionCon=0.8, maxHands=1)

#Variables

imgList = []

delay = 30

button Pressed = False

counter = 0

drawMode = False

imgNumber = 0

delayCounter = 0

annotations = [[]]

annotationNumber = -1

annotationStart = False

hs, ws = int(120 \* 1), int(213 \* 1)

#Get the list presentation images

pathImages = sorted(os.listdir(folderPath), key=len)

print(pathImages)

while True:

# Import images

success, img = cap.read()

img = cv2,flip(img, 1)

pathFullImage = os.path.join(folderPath, pathImages[imgNumber])

imgCurrent = cv2.imread(pathFullImage)

hands, img = detectorHand.findHands(img)

cv2.line(img, (0, gestureThreshold), (width, gestureThreshold), (0, 255, 0), 10)

if hands and buttonPressed is False:

hand = hands[0]

cx, cy = hand["center"]

lmList = hand["lmList"] # List of 21 Landmark points

fingers = detectorHand.fingersUp(hand) # List of which fingers are up

xVal = int(np.interp(lmList[8][0], [width // 2, width], [0, width]))

yVal = int(np.interp(lmList[8][1], [150, height-150], [0, height]))

indexFinger = xVal, yVal

if cy <= gestureThreshold:

if fingers == [1, 0, 0, 0, 0]:

print("Left")

buttonPressed = True

if imgNumber> 0:

imgNumber -= 1

annotations = [[]]

annotationNumber = -1

annotationStart = False

if fingers == [0, 0, 0, 0, 1]:

print("Right")

buttonPressed = True

if imgNumber<len(pathImages) - 1:

imgNumber += 1

annotations = [[]]

annotationNumber = -1

annotationStart = False

if fingers == [0, 1, 1, 0, 0]:

cv2.circle(imgCurrent, indexFinger, 12, (0, 0, 255), cv2.FILLED)

if fingers == [0, 1, 0, 0, 0]:

if annotationStart is False:

annotationStart = True

annotationNumber += 1

annotations.append([])

print(annotationNumber)

annotations[annotationNumber].append(indexFinger)

cv2.circle(imgCurrent, indexFinger, 12, (0, 0, 255), cv2.FILLED)

else:

annotationStart = False

if fingers == [0, 1, 1, 1, 0]:

if annotations:

annotations.pop(-1)

annotationNumber -= 1

buttonPressed = True

else:

annotationStart = False

if buttonPressed:

counter += 1

if counter > delay:

counter = 0

buttonPressed = False

for i, annotation in enumerate(annotations):

for j in range(len(annotation)):

if j != 0:

cv2.line(imgCurrent, annotation[j - 1], annotation[j], (0, 0, 200), 12)

# Adding WebCam image on the Slides

imgSmall = cv2.resize(img, (ws, hs))

h, w, \_ = imgCurrent.shape

imgCurrent[0:hs, w - ws: w] = imgSmall

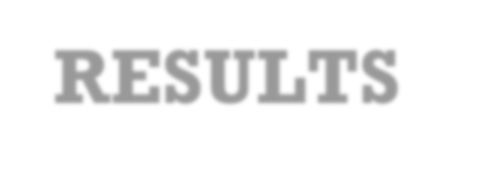
cv2.imshow("Slides", imgCurrent)

cv2.imshow("Image", img)

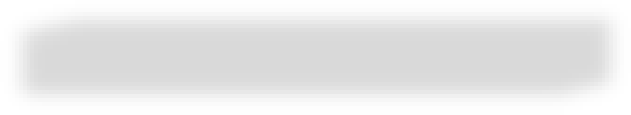
key = cv2.waitKey(1)

if key == ord('q'):

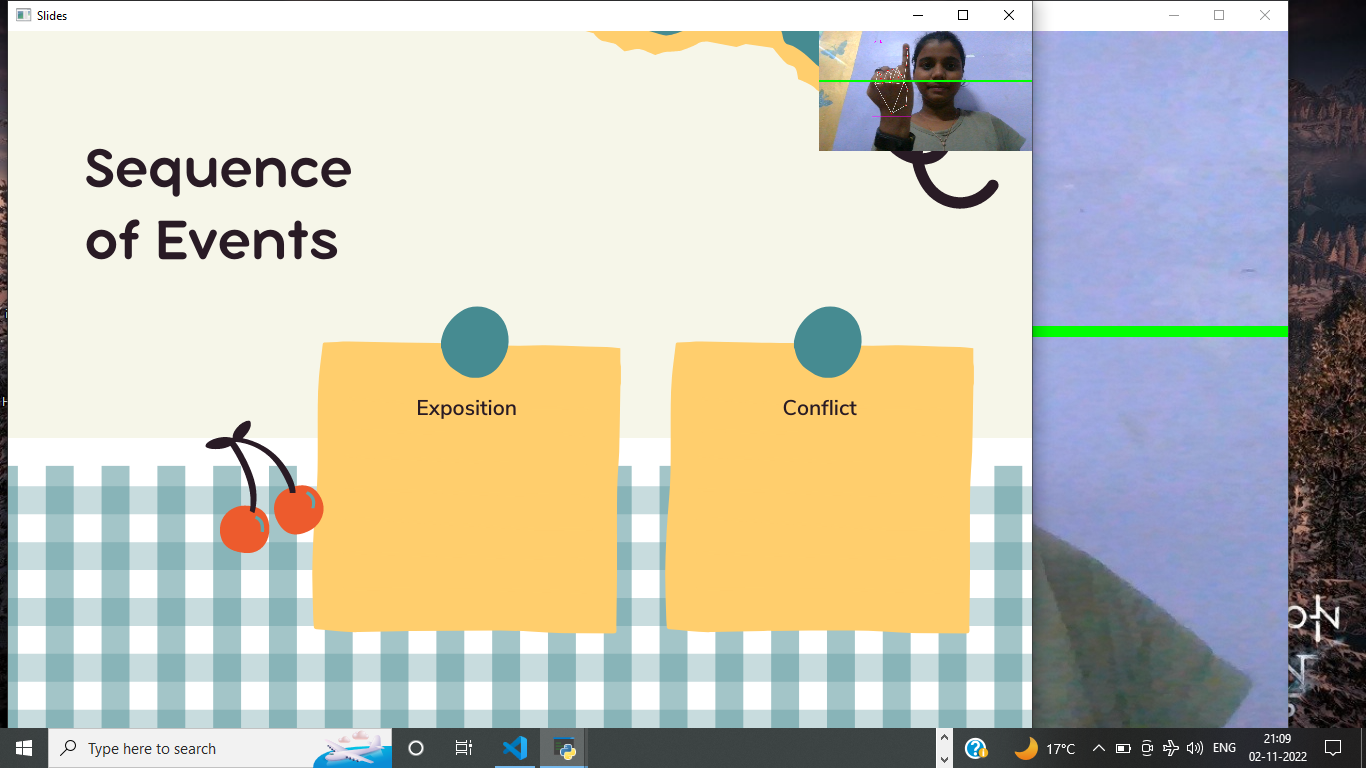
break



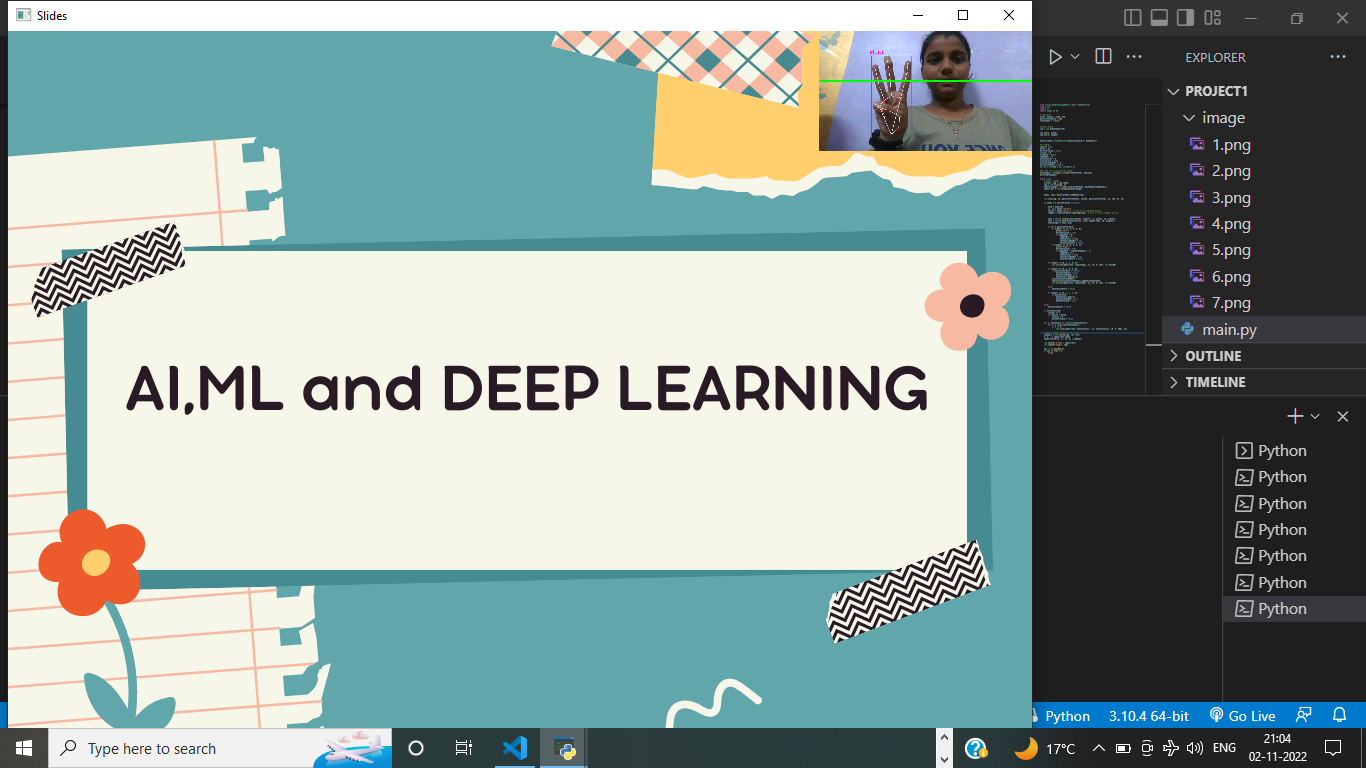
**RESULTS**



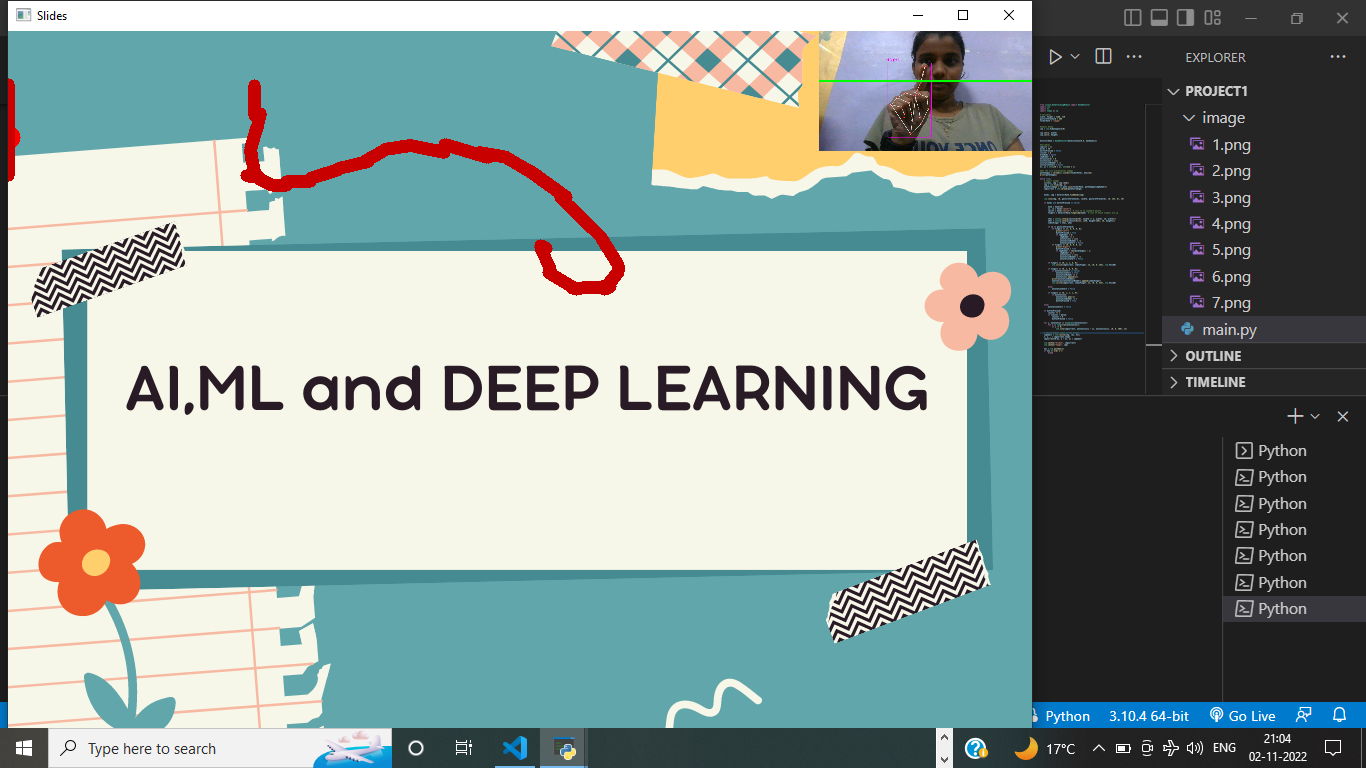
## Results:



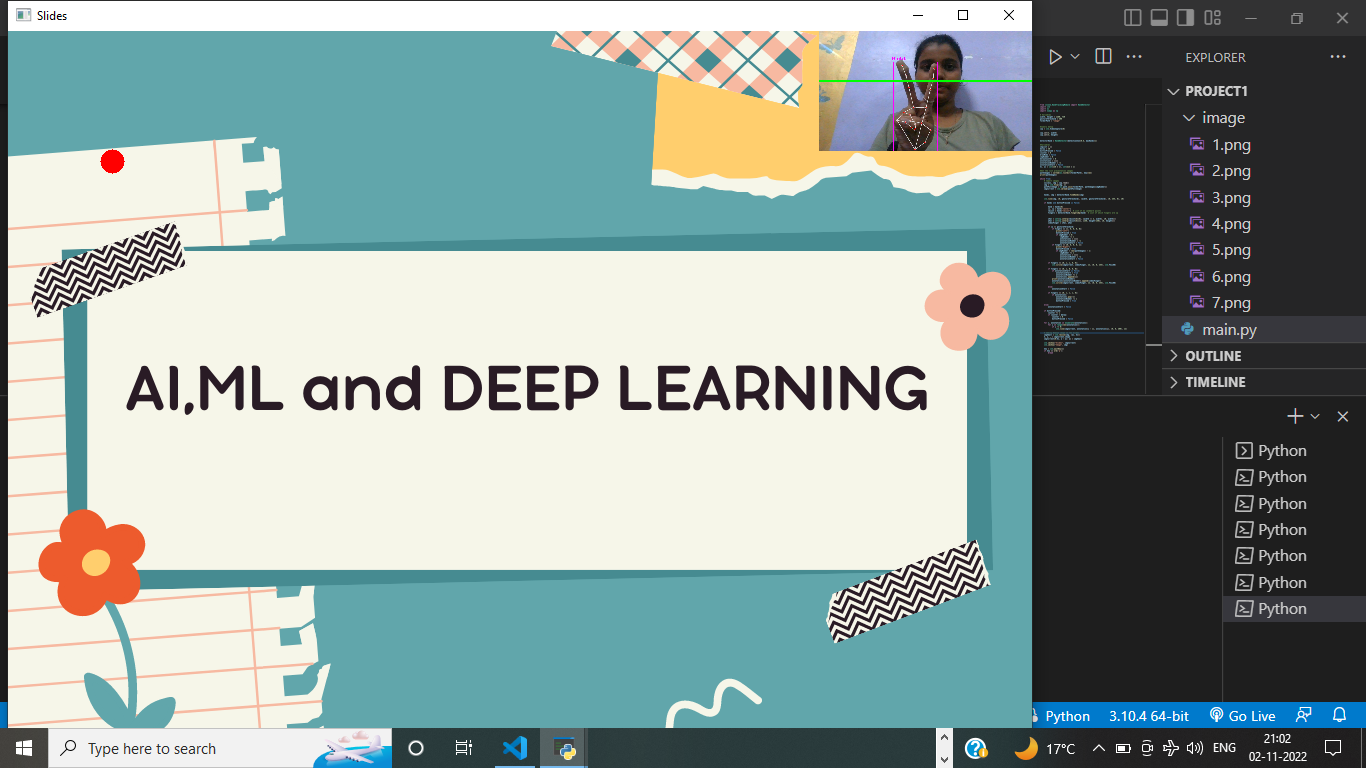
**Fig 1 : For going to next slide**



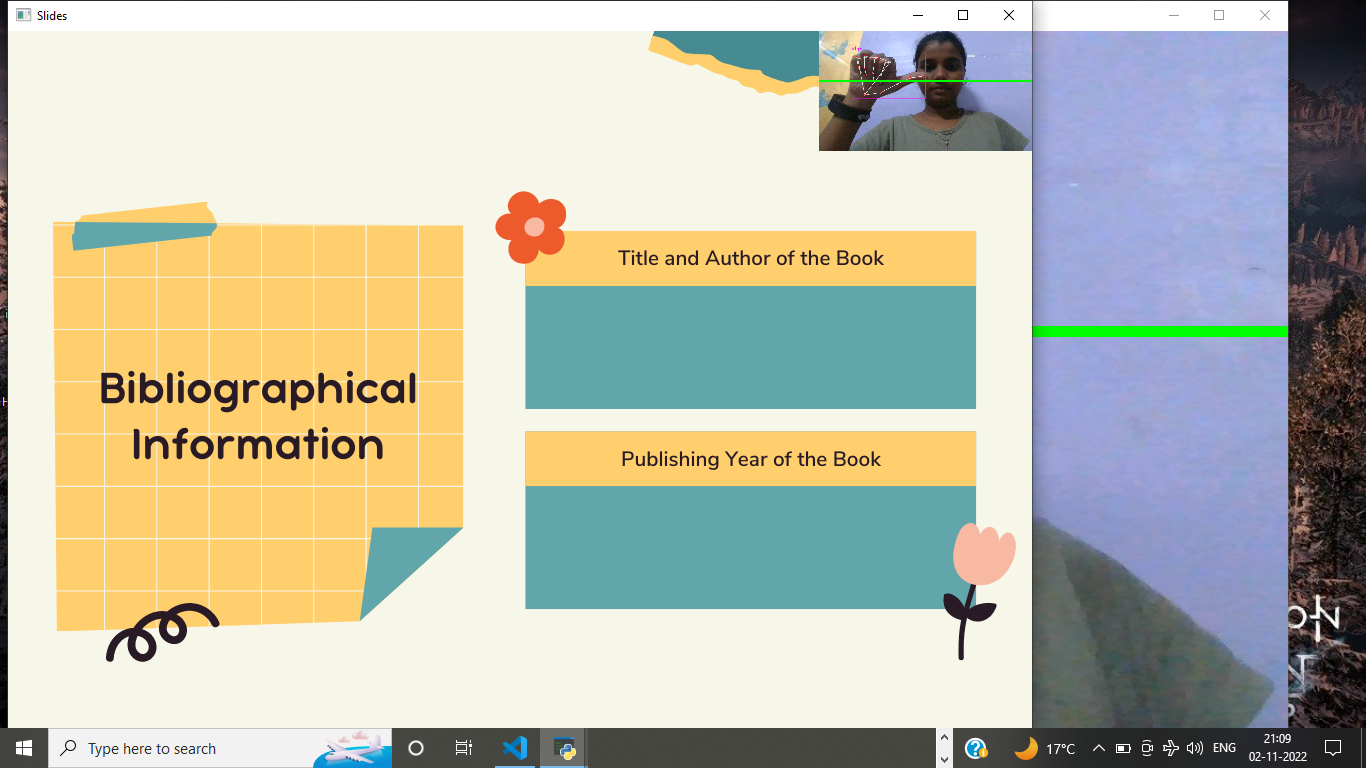
**Fig 2 : For Erasing elements**



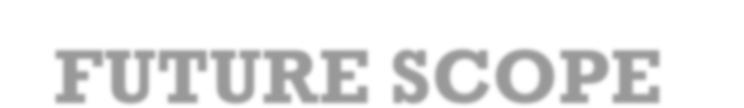
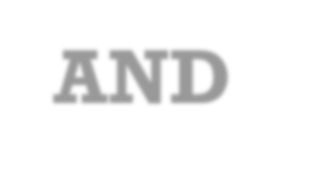
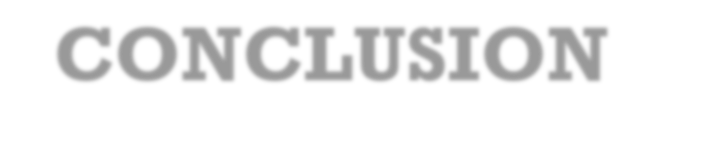
**Fig 3 : Pointer for writing**

****

**Fig 4 : For creating pointer**

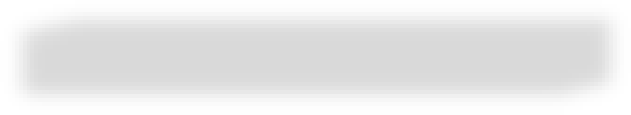


**Fig 5 : For landing on Previous slide**



**CONCLUSION AND**

**FUTURE SCOPE**



## Conclusion and Future Scope:

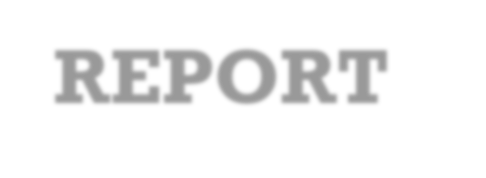
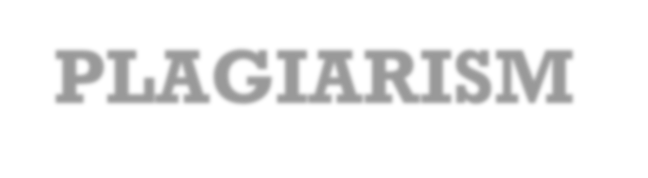
### Conclusion

There are several areas for the future research suggested by the result of the above work. For navigating slides, a wireless remote might be the optimal solution. In this research, we showed that content-centric gestures provide the clearest motivation for such systems, in contrast to previous work that has focused exclusively on using gestures for presentation navigation.

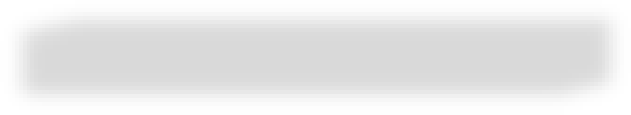
The drawbacks of previously implemented methods like circular profiling method using single hand are overcome since distance transform method can use both the hands. This does not require any training phase to identify a hand gesture hence does not require storage of images in database to recognize the hand gestures. Furthermore, they are suitable for real-time applications and solve the issues of time delay between the spotting and the recognition tasks

### FutureScope:

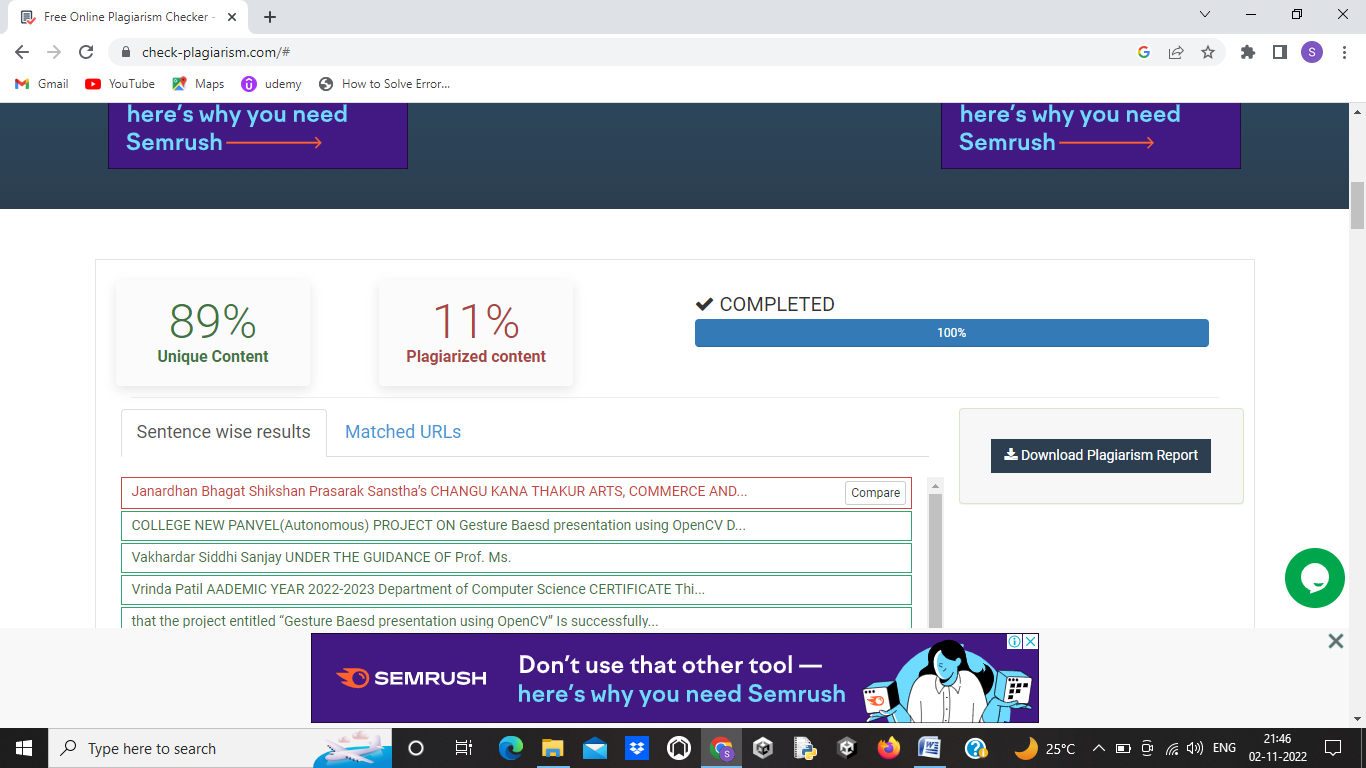
* Thesystemcanbeimplementedonanysmalldevicelikeraspberrypi.
* Motionofhandor Gestures is limited or specific task but we can add more features to do more work.

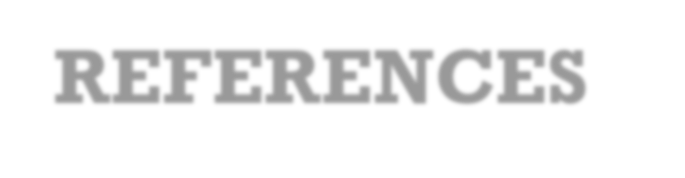


**PLAGIARISM REPORT**

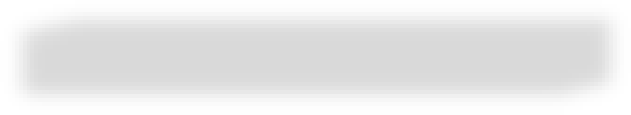


1. **Plagarism Report:**





**REFERENCES**



## References:

[1]Kamarulazhar Daud “Hand gesture recognition on python and opencv” OPSCIENCE

https://iopscience.iop.org/article/10.1088/1757-899X/1045/1/012043

September 07 2022.

[2] Syed Abdul Gaffar Shakhadri “Building a Hand Tracking System using OpenCV” Blog

https://www.analyticsvidhya.com/blog/2021/07/building-a-hand-tracking-system-using-opencv/ July 08 2022

[3]Pratik Bhatt, Deval Mehta, Nirav Faraswami “Apparatus And Method For Gesture Recognition” G.H.Patel College Of Engineering And Technology,V.V.Nagar Gujarat Technological University, Ahmedabad May, 2012

[4] Mohd. Baqir Khan, Kavya Mishra, Mohammed Abdul Qadeer “Gesture Recognition using Open-CV” IEEE

https://ieeexplore.ieee.org/document/8418531/ November 13 2017

### Websites:

* <https://towardsdatascience.com/sign-language-recognition-using-deep-learning-6549268c60bd>
* <https://ieeexplore.ieee.org/document/7916786>
* <https://stackoverflow.com/>