**NAGARJUNA COLLEGE OF ENGINEERING AND**

**TECHNOLOGY**

(An Autonomous College under VTU, Belagavi)



# A Mini-Project Report On

**“Hand Gestures Recognition”**

Submitted In Partial Fulfillment for the Completion of the Course

Mini Project

In

**COMPUTER SCIENCE AND ENGINEERING**

Submitted by

Ms. Monika S.P 1NC20CS045

Ms. Nikitha Raj K.R 1NC20CS050

Under the guidance of

Mr. Kiran. J

Asst. Professor

Dept. of CSE, NCET

**DEPARTMENT OF COMPUTER SCIENCE & ENGNIEERING**

NAGARJUNA COLLEGE OF ENGINEERING AND

TECHNOLOGY

(An Autonomous College under VTU, Accredited by NAAC with “A”)

Maidugurki (V), Venkatagiri Kote (P), Devanahalli (T), Bengaluru-562164.

**2022-23**

## NAGARJUNA COLLEGE OF ENGINEERING AND TECHNOLOGY

(An Autonomous College under VTU, Belagavi)

Bengaluru-562164, Karnataka, India



# CERTIFICATE

This is to certify that the project work entitled “Hand

Gestures Recognition” is carried out by Ms. Monika S.P (1NC20CS045), Ms. Nikitha Raj K.R (1NC20CS050), bonafide students of Nagarjuna College of Engineering and Technology, an autonomous institution under Visvesvaraya Technological University, Belagavi, submitted in partial fulfillment of the requirement for the award of Bachelor of Engineering in Computer Science and Engineering during the academic year 2022-2023. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report deposited in the departmental library. The project report has been approved, as it satisfies the academic requirement in respect of Project work prescribed for the said degree

|  |  |  |
| --- | --- | --- |
| **Name & Signature of the Guide** | **Name & Signature of the HOD** | **Name & Signature of the Principal** |
| Mr. Kiran.J | Dr. Anil Kannur | Dr. B V Ravishankar |
| Assistant prof, Dept. of CSE | HOD, Dept. of CSE | Principal, NCET |
|  | External Viva-Voce |  |
| Name of the Examiner |  | Signature with date |

1.

2.

# ACKNOWLEDGMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without mentioning the names of the people who made it possible, whose consistent guidance and encouragement crowned our effort with success. We consider it is our privilege and duty to express our gratitude and respect to all those who guided us in the completion of this project work.

At the very outset, firstly we would like to be highly grateful to the college, Nagarjuna College of Engineering and Technology, for providing us with all the necessary help and grooming us to be Computer Science and Engineers.

We take this privilege to express our deep gratitude to Dr. B V Ravishankar, Principal, Nagarjuna College of Engineering and Technology for his constant support, and encouragement, and for providing all the required facilities.

We would like to thank our Project coordinator Mrs.Swathi S, Associate Professor, Department of Computer Science and Engineering, for her constant support and assistance at every stage.

It‟s our immense pleasure to thank our beloved guide Mr. Kiran J, Assistant Professor, Department of Computer Science and Engineering, for helping and guiding us to complete this project work.

# ABSTRACT

. Understanding the activities of human from videos is demanding task in Computer Vision. Identifying the actions being accomplished by the human in the video sequence automatically and tagging their actions is the prime functionality of intelligent video systems. This project is about identifying or predicting the activity someone is performing based on the sensor data recorded. Human activity

recognition is applicable in scenarios where we require knowledge of an individual's

intelligent video systems. This project is about identifying or predicting the activity someone is performing based on the sensor data recorded. Human activity

recognition is applicable in scenarios where we require knowledge of an individual's

activity in real time. The goal of activity recognition is to identify the actions and

objectives of one or more objects from a series of examination on the action of object

and their environmental condition. The major applications of Human Activity

Recognition vary from Content-based Video Analytics, Robotics, Human-Computer

Interaction, Human fall detection, Ambient Intelligence, Visual Surveillance, Video

Indexing etc. The Experimental Evaluation of various papers are observed efficiently

with the various performance metrics like Precision, Recall, and Accuracy

# TABLE OF CONTENTS

Acknowledgment i

Abstract ii

Chapter No. Title Page no.

1. Introduction
2. Literature Survey
3. Objectives
4. Requirement Specification
   1. Software Requirement
   2. Hardware Requirement
5. System Analysis
   1. Existing System
6. System Design
7. System Implementation
8. Testing
9. Screen Shots
10. Future Enhancement
11. Conclusion

## INTRODUCTION

Identifying the gesture of human by computer is a vital dynamic research topic. It has numerous implementations in practical environment control and indication language, robot command or Melody design. In this Machine learning project we are going to make a real-time hand gesture Recognizer using the Media pipe, Framework and Tensor flow in open CV and Python. Neural Network is familiar to Artificial Neural Network. Neural Network is subdivision of Machine Learning and core of Deep learning algorithms. The idea of Neural Network is encouraged by the human brain. It imitates the way that that biological neurons dispatch the signals to one another Neural Networks made up of node layers that has input layer, single layers and an output layer, additional layers.

### LITERATURE SURVEY

Gesture are dominant means of communication amid humans .In this article vision based approach using Indian sign language. Using Indian sign language interpreter job opportunities can be provided for deaf and dumb in government sector, it sectors, private sectors. Education for deaf and dumb people can be given easily and effectively using Indian Sign Language (ISL).

HAND GESTURE RECOGNITION USING SIGN LANGUAGE

Hand plays a very important role in sign language. Hand gesture is difficult to recognize as it has many connected joints and links. Study of Skeleton model is very much necessary for developing any king of Hand Gesture Recognition

System. Recognizing gesture is not simple task as it includes features such as

* Object Detection
* Object description
* Motion Modelling
* Motion Analysis
* Pattern Recognition
* Machine Learning
* Psycholinguistic Studies

As stated by WHO in the world popular 5% is having a disability of speaking and hearing. The communication between deaf and mute and normal person has been challenging task. The gesture of disabled person is detected using 3D auto meter and it process the database through node MCU and Raseberrypi .The Results showed that 99.7% accuracy was there in CNN. In this article study of VBHGR using Indian Sing language. Standard Sign language vary from country to country. Standard Sign language cannot be written in voice communication. Indian deaf and dumb group use Indian sign Language.

AN EFFICIENT FRAMEWORK FOR INDIAN SIGN LANGUAGE RECOGNITION USING WAVELET TRANSFORM.

The ISLR (Interpreting and sign language resources) system is introduced to pattern recognition technique and it has Two modules i.e., Feature Extraction by image processing and classification. In feature extraction an initial set of raw data is divided and it reduced to more manageable groups. So when you process it will be easier Discrete Wavelet Transform (DWT) is used to extract information from the images and also used to recognize the sign language.

When we done experimentally the result of proposed hand gesture recognition system achieves maximum 99.23% classification accuracy when we use cosine distance classifier.

HAND GESTURE RECOGNITION USING PCA (PRINCIPLE COMPONENT ANALYSIS)

There are many forms of work have been done by using PCA and Eigen image approach. In this Paper author presented some schemes by using a database driven hand gesture recognition based on skin color model approach and threshold approach. There are two types of gestures Dynamic gesture and Static gesture. Dynamic gestures is used for hand movement trajectory and to change the shape of the hand and it is more complex gesture recognition approach And static gesture requires only the processing a single image at the input of the classifier PCA is a method which handles images even having noise. The Model proposed does not evaluate images with uncontrolled background and also the model works with Static gestures .In future of this system can be upgraded to work for uncontrolled background and the efficiency of PCA can be accelerated.

## OBJECTIVES

The area of the project is to create a technique to identify the hand signal using Media pipe library, Open CV library and Tensor Flow library. Only a special hardware is needed for higher level recognition.

1. Develop a computer vision
2. Test the computer application
3. Record the outcomes of the project

## SYSTEM REQUIREMENT AND SPECIFICATION

SOFTWARE REQUIREMENTS

* Python-Software
* Open CV
* Media pipe
* Tensor Flow
* Database
* Python Libraries

HARDWARE REQUIREMENT

* RAM: 8GB
* Hard disk: 2GB
* Processor: I3 or above

## SYSTEM ANALYSES

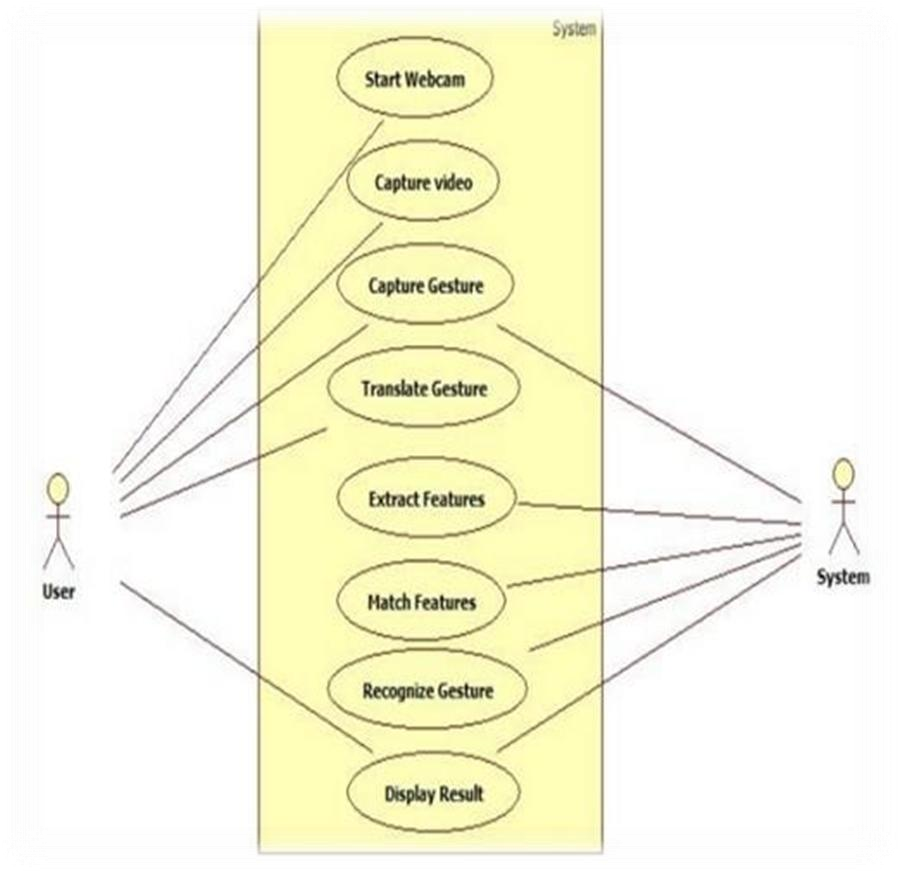
EXISTING SYSTEM

Image processing and deep learning applications is utilized on egocentric hand gesture recognition and fingertip detection to face up challenges. Nevertheless the image Processing based applications have dependency on background hand shape color, therefore tend to malfunction in complex and diverse scenarios. Moreover the applications that use convex hull technique for gesture recognition and fingertip detection have impulsive drawbacks. For example, in spite of the fact that they can recognize the gesture and detect fingers but they are unable to group fingers and thus cannot inform which fingertips have been detected.

## SYSTEM DESIGN

USE CASE DIAGRAM

A use case defines a goal-oriented set of interactions between external entities and the system under consideration. The external entities, which interact with the system, and it acts such as user and systems for our Application. A set of use cases describe the complete functionality of the system at a particular level. There are different entities involved in making interaction with the System. This System is based on Human Computer Interaction so it basically include the user, computer and it is connected to both digitally that is web camera which is connected to System. When the execution of the program starts, it firstly invoke the web camera to capture RGB image of the hand. The extract the features of hand it detect hand key point then it recognize the hand gestures.



METHODOLOGY

Image Acquisition

from camera

Hand Region

Segmentation

Hand detection

and tracking

Hand Posture

recognition

Classified

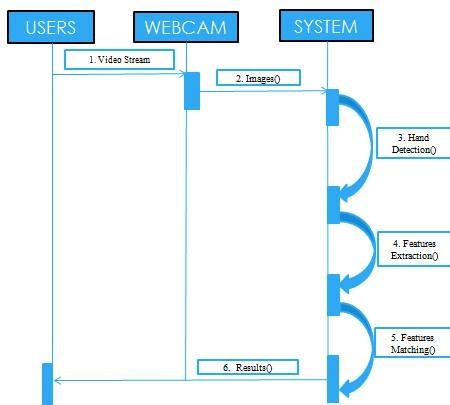
Gesture

Display as text

SEQUENCE DIAGRAM

Sequence diagram are an easy way of describing the behavior of a system by viewing the interaction between the system and the environment. A sequence diagram shows an interaction arranged in a time sequence. A sequence diagram has two dimensions: vertical dimension represents time; the horizontal dimension represents the objects existence during the interaction. The shown Sequence diagram explain the flow of the program. As this System is based on Human Computer Interaction so it basically include the user, computer and the medium to connect both digitally that is web camera. As the execution of the program starts, it firstly invoke the web camera to take RGB image of the hand. Then the image is segmented and filtered to reduce the noise in the image .After the removal of the noise the hand gestures are detected.

Sequence Diagram



## SYSTEM IMPLEMENTION

Steps to solve the project:

* Import necessary packages  Initialize models.
* Read from Webcam
* Detect hand key points.
* Recognize hand gestures

Step 1-Import necessary packages:

To build this Hand gesture Recognition project, we will need four packages.

#import necessary packages using Python open CV Import cv2 import numpy as np import mediapipe as np

import tensorflow.keras.models import load\_model Step 2-Intialize models:

Initialize Mediapipe:

mpHands = mp.solutions.hands mpdraw = mp.Solutions.drawing\_utils

* Mp.solution.hand module performs the hand recognition algorithm. So we can create the object and store it in mp.Hands.
* Using Mp.Hands. Hands method we configured the model. The first argument is max\_num\_hands, that means maximum number of hands will be detected by the model in single frame. Mediapipe can detect nultiple hands in single frame,but we will detect one hand in this project
* Mp.solutions.drawing\_utils will draw the detected key points ,so that we don‟t need to draw them manually.

Initialize Tensorflow:

#load the gesture recognition model

model = load\_model(„mp\_hand\_gesture‟)

#load class names f = open(„gesture.names‟,‟r‟) classNames = f.read().split(„\n‟) f.close()

print(ClassNames)

* Using load\_model function we load the tensorflow pretrained model.
* Gesture.names file contains the name of the gesture classes. First we use python‟s inbuilt open function and read the file.

OUTPUT

[„okay‟,‟peace‟,‟thumbsup‟,thumbsdowm‟,‟callme‟,‟stop‟,‟rock‟,‟live long‟,‟fist‟,‟smile‟]

The model can recognize 10 different gestures.

Step 3:Read franes for a webcam:

#intialize the webcam cap =cv2.videocapture(0)

while True:

#Read each frame from webcam

\_, frame = cap.read()

x,y,c = frame.shape

#flip the frame vertically Frame = cv.flip(frame,1) #Show the final output cv.imshow(“Output”,frame) if cv2.waitkey(1) == ord(„q): break

#release the webcam and destroy all windows cap.release()

cv2.destroyAllwindows()

* We create a video Capture object and pass an argument “0”. It is the camera ID of the system. In this case, we have one webcam connected to the system.
* The cap. read() function reads each frame from the webcam.
* Cv2. flip()function flips the frame.
* Cv2.im show() shows the frame on new open cv window.
* The cv2.wait key()function keeps the window open until the key „q‟ is pressed.

Step 4: Detect hand key points

Frame gb = cv2.cvtcolor(frame, cv2.COLOR\_BGR2RGB)

#get hand landmark prediction result = hand.process(framergb)

className = „ „ #post process the result if result.multi\_hand\_landmarks:

landmarks = [] for handslms in result.multi\_land\_marks: for lm in handslms.landmark:

#print(id,lm) lmx = int(lm.x \* x) lmy = int(lm.y \* y)

landmarks. append([lmx,lmy]) #Drawing landmarks on frames mpDraw.draw\_landmarks(frame,handlms,mpHand,HAND\_CONNECTIONS)

* Mediapipe works with RGB images but Opencv reads images in BGR format. So,using cv2.cvtCOLOR()function we convert the frame to RGB format.
* The process function takes RGB frame and returns class.
* Then we check if any hand is detected or not, using result.multi\_hand\_landmarks method
* After that, we loop through each detection and store the co-ordination on a list called landmarks.
* Here image height(y) and width(x) are multiped with the result because the model returns a normalized result. This means each value in the result is between 0 or 1.
* Finally we use mpDraw.draw\_landmarks() function to draw all the landmarks in the frame.

Step 5: Recognize hand gestures:

prediction = model.predict([landmarks]) print(prediction) classID = np .argmax(prediction)

#Show prediction on the frame

Cv2.putText(frame,classname,(10,50),cv2.FONT\_HERSEY\_SIMPLEX,1,

(0,0,255) , 2, cv2.LINE\_AA)

* The model.predict() function takes a list of landmarks and returns an array contains 10 prediction classes for each landmark.
* np.argmax() returns the index of the maximum value in the list
* after getting the index we can simply take the class mname from the className.list.
* Then using the cv2\_puttext function we show the detected gesture into the frame.

## TESTING

In extraction process hand gesture first input image is read by the web camera on the client side. Then it detect the hand key points from the input which is read by the web camera and also detect the hand movement .Then gestures of the hand is detected by the system. The detected hand gesture is shown below.

## SCREEN SHOTS

FUTURE ENHANCEMENT

The system could be trained for one or two gestures to recognition than all at once and made ready for testing .Some changes in the code can be done to implement the efficient gesture recognition. To make algorithm to work for all skin types and all light conditions which is difficult to do all together. Using Center of Gravity (COG) to manage the orientation factor can make the system more suitable for real time applications. If the code is developed in VC.net the speed of the preprocessing could be more. Some work can be done in detecting accurate and faster which could help deaf and dumb.

CONCLUSION

The project is outlining a hand gesture recognition system that is designed to be recognize the hand gesture in real time there are different types of hand gesture recognition have been implemented in this system it is helpful to the deaf and dump people it only require web camera to capture the input image and the system will recognize the gestures