**Algorithm for Gesture Detection Using Image Processing with Python .**

***Abstract* —**: Gesture is the most primitive technique for conversation amongst human being. Today in the time of current day innovation gesture recognition affects the world differently, from the physically challenged individuals to robotic manage to virtual fact conditions. In this paper, a deep convolution neural network is proposed to immediately classify hand gestures in pictures without any segmentation or detection stage that could discard the irrelevant not-hand area.

***Keywords*** *—* : Hand gesture recognition, Human Computer interaction interface etc.

1. INTRODUCTION

Hand gesture recognition system received great attention in the recent few years because of its manifoldness applications and the ability to interact with machine efficiently through human computer interaction. In this paper a survey of recent hand gesture recognition systems is presented. Key issues of hand gesture recognition system are presented with challenges of gesture system. Review methods of recent postures and gestures recognition system presented as well. Summary of research results of hand gesture methods, databases, and comparison between main gesture recognition phases are also given.

Gesture is the most primitive technique for conversation amongst human being. Today in the time of current day innovation gesture recognition affects the world differently, from the physically challenged individuals to robotic manage to virtual fact conditions. Human hand gestures provide the natural and advantageous approach of non-verbal conversation with the computer interface. Hand gestures are the considerable body moves that are actions of hands, arms or fingers.

Hand gesture recognizable proof levels from the static gesture with the complicated foundation or dynamic gestures that express the human feeling and communicate with computer or humans. The hand is specifically use as the contribution to the machine, for the verbal exchange reason for gesture identification there is no need of an intermediate medium. In this paper, a deep convolution neural network is proposed to immediately classify hand gestures in pictures without any segmentation or detection stage that could discard the irrelevant not-hand area.

1. LITERATURE REVIEW

A lot of technological fields have already implemented gesture technologies in Smart Homes, Cars and other Internet if things devices. A good example could be Project Soli by Google where Radar technology is used to track micro-motions .

1. **Appearance based approaches:**

In this approach, the visual appearance of the input hand image is modeled using the characteristic extraction, which are then compared to the feature extracted of the saved image. This technique has the advantage of actual time overall performance and less complicated as compared to that of 3D mannequin based approach.

1. **3D model based approaches**:

In this strategy for modeling and analysis of hand shape the 3D model is used. While doing the 2D projection the some information is lost, So intended a depth parameter is added in 3D model to make it more accurate 3D model are categorized into volumetric and skeletal models. These two models have the constraints while designing. Volumetric model is used in real time application as it deals with the 3D model base look of human hand. The primary constraint while designing this system is, it has the massive dimensionality parameters which cause the designer to work in three dimensional.

1. METHODOLOGY/EXPERIMENTAL

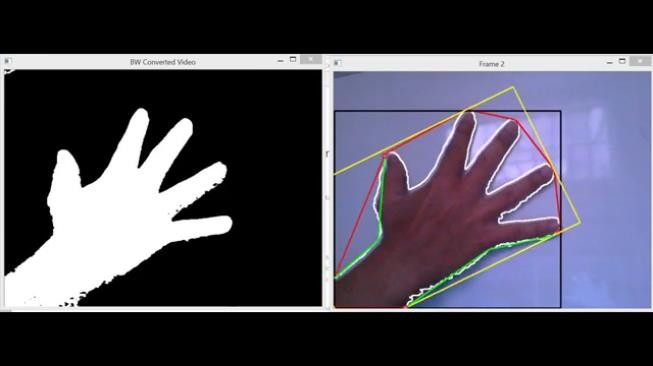
Hand gesture recognition system is used as human computer interaction medium as well as for paralyzed persons so it’s useful current technology. To achieve the goal of system several tools and technologies have been used. The study of tools and technologies and the previous technologies used are studied and researched thoroughly.

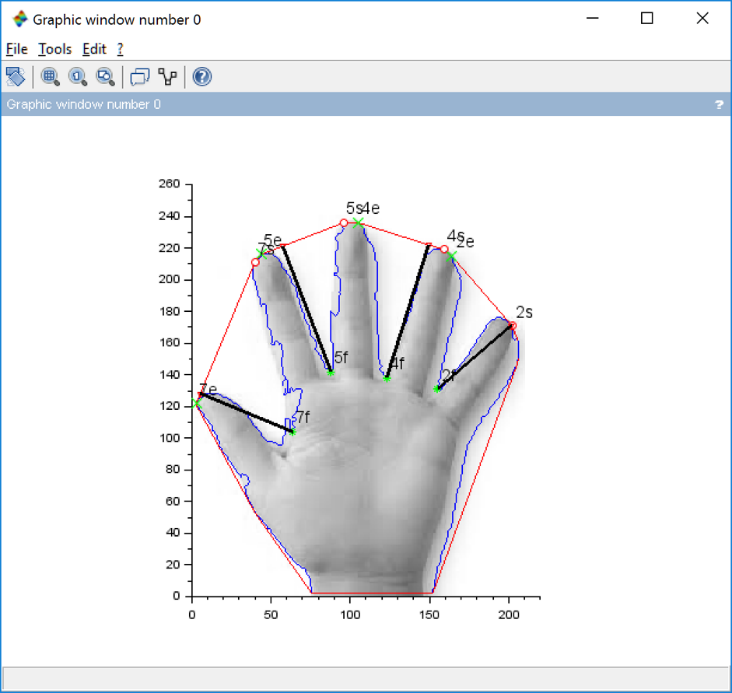
1. **Open cv**:

Open CV is the set of open source computer vision library and computer vision is the way of teaching intelligence to machine and making them see just like human. Opencv is an image processing library which was created by intel and later supported by willow garage and now maintained by idseez. It’s available for mac ,windows and linux. It works in C,C++,Python. It is open source and free as well as easy to use and install. OpenCV helps the deep learning frameworks like TensorFlow, Torch/PyTorch and Caffe.

1. **Computer vision:-**

Computer vision is the way of teaching intelligence to the machines and making them see the images or videos and extract the data from them just like humans. The goal of computer vision is to write computer programs that can interpret images or videos i.e. the programs can take images as an input and give the descriptive information about the image. This whole process occurs in several stages as first the image is taken as the imput and then processing of image occurs.



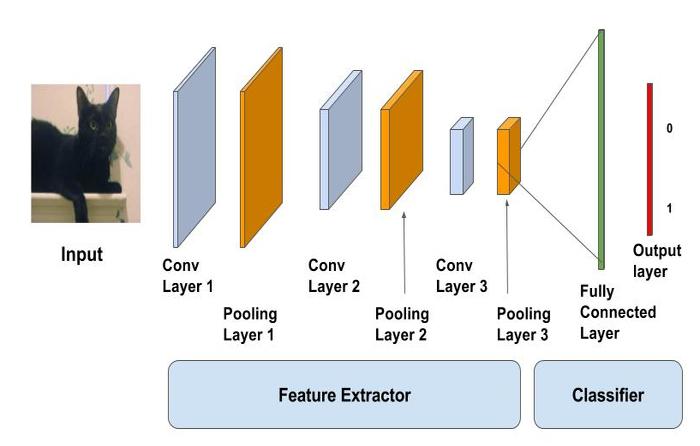


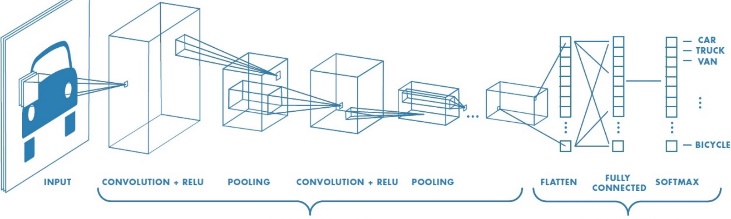
1. **Deep learning:-**

Deep learning is the machine learning technique which learns directly from the data. The data can be in any digital form as images, text ,sound as well as videos. Deep learning works on the heavy set of data. It basically classifies the type of information and then gives it as an output. There are many layers in the deep learning to make it efficient and powerful. There are mainly two layers in the algorithm used in deep learning. The first one is feature extraction layer which extract the features based on convolutional neural network and the one is classification layer which classifies the image and gives the information as the result.

1. **Convolutional neural network**:-

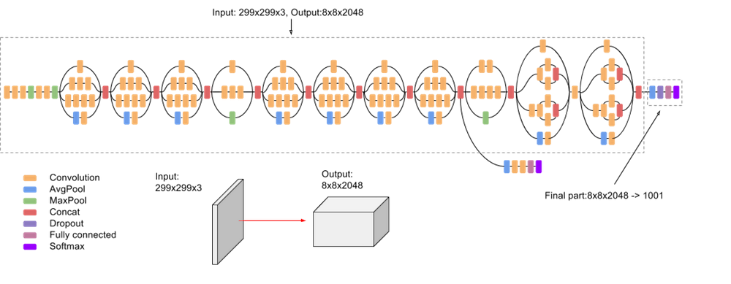
In deep learning, a convolutional neural network (CNN) is a type of deep neural networks, which deals with the set of data to extract information about that data. Like images, sounds or videos etc. can be used in the CNN for the data extraction. There are mainly three things in CNN. First one is local receptive field and then shared weight and biases and the last one is activation and pooling. In CNN, first the neural networks are trained using a heavy set of data so that the CNN can extract the feature of given input. When the input is given, first image preprocessing is done then the feature extraction occurs on the basis of set of data stored and then the classification of data is done and output is shown as the result.





1. **Inception v3**:-

Transfer learning is a machine learning approach which makes use of a pre-trained neural network. There are mainly 2 phases in this method. First one is Feature extraction phase which uses convolutional neural network and another one is Classification phase with fully-connected network. For example, the image recognition model recognized as Inception-v3 consists of two parts: classification layer and softmax layers



The result is an algorithm that can be used to identify and manipulate hand gestures through the use of various types of sensors and cameras.

Some image of the final program are shown below

1. LIMITATIONS

In this section, drawbacks of some discussed methods are explained: Orientation histogram

method applied in have some problems which are; similar gestures might have different

orientation histograms and different gestures could have similar orientation histograms, besides

that, the proposed method achieved well for any objects that dominate the image even if it is not

the hand gesture. Wrong object extraction problem raised if the objects larger than the

hand. The performance of recognition algorithm decreases when the distance greater than 1.5

meters between the user and the camera. Besides that, its variation to lighting condition changes

and unwanted objects might overlap with the hand gesture.

In the system is variation to

environment lighting changes which produces erroneous segmentation of the hand region.

1. FUTURE SCOPE

Hand gestures recognition system has been applied for different applications on different

domains, as mentioned in including; sign language translation, virtual environments, smart

surveillance, robot control, medical systems etc. overview of some hand gesture application areas

are listed below.

* 1. Sign Language Recognition.
  2. Robot Control
  3. Graphic Editor control
  4. Virtual Environments.
  5. Numbers Recognition
  6. Television and Electronics Control
  7. 3D Modelling.

#define area of hull and area of hand areahull = cv2.contourArea(hull) areacnt = cv2.contourArea(cnt)

import cv2

import numpy as np import math

1. CODE

#find the percentage of area not covered by hand in convex hull

arearatio=((areahull-areacnt)/areacnt)\*100

#find the defects in convex hull with respect to hand hull = cv2.convexHull(approx, returnPoints=False) defects = cv2.convexityDefects(approx, hull)

cap = cv2.VideoCapture(0)

while(1):

try: #an error comes if it does not find anything in window as it cannot find contour of max area

#therefore this try error statement

ret, frame = cap.read() frame=cv2.flip(frame,1)

kernel = np.ones((3,3),np.uint8)

#define region of interest roi=frame[100:300, 100:300]

cv2.rectangle(frame,(100,100),(300,300),(0,255,0),0) hsv = cv2.cvtColor(roi, cv2.COLOR\_BGR2HSV)

# define range of skin color in HSV

lower\_skin = np.array([0,20,70], dtype=np.uint8) upper\_skin = np.array([20,255,255], dtype=np.uint8)

#extract skin colur imagw

mask = cv2.inRange(hsv, lower\_skin, upper\_skin)

#extrapolate the hand to fill dark spots within mask = cv2.dilate(mask,kernel,iterations = 4)

#blur the image

mask = cv2.GaussianBlur(mask,(5,5),100)

#find contours

\_,contours,hierarchy= cv2.findContours(mask,cv2.RETR\_TREE,cv2.CHAIN\_APP ROX\_SIMPLE)

#find contour of max area(hand)

cnt = max(contours, key = lambda x: cv2.contourArea(x))

#approx the contour a little

epsilon = 0.0005\*cv2.arcLength(cnt,True) approx= cv2.approxPolyDP(cnt,epsilon,True)

#make convex hull around hand hull = cv2.convexHull(cnt)

# l = no. of defects l=0

#code for finding no. of defects due to fingers for i in range(defects.shape[0]):

s,e,f,d = defects[i,0]

start = tuple(approx[s][0]) end = tuple(approx[e][0]) far = tuple(approx[f][0]) pt= (100,180)

# find length of all sides of triangle

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)

s = (a+b+c)/2

ar = math.sqrt(s\*(s-a)\*(s-b)\*(s-c))

#distance between point and convex hull d=(2\*ar)/a

# apply cosine rule here

angle = math.acos((b\*\*2 + c\*\*2 - a\*\*2)/(2\*b\*c)) \*

57

# ignore angles > 90 and ignore points very close to convex hull(they generally come due to noise)

if angle <= 90 and d>30:

l += 1

cv2.circle(roi, far, 3, [255,0,0], -1)

#draw lines around hand cv2.line(roi,start, end, [0,255,0], 2)

l+=1

#print corresponding gestures which are in their ranges font = cv2.FONT\_HERSHEY\_SIMPLEX

if l==1:

if areacnt<2000:

cv2.putText(frame,'Put hand in the box',(0,50), font, 2, (0,0,255), 3, cv2.LINE\_AA)

else:

if arearatio<12:

cv2.putText(frame,'0',(0,50), font, 2, (0,0,255),

3, cv2.LINE\_AA)

elif arearatio<17.5:

cv2.putText(frame,'Best of luck',(0,50), font, 2, (0,0,255), 3, cv2.LINE\_AA)

else:

cv2.putText(frame,'1',(0,50), font, 2, (0,0,255),

3, cv2.LINE\_AA)

elif l==2:

cv2.putText(frame,'2',(0,50), font, 2, (0,0,255), 3, cv2.LINE\_AA)

elif l==3:

if arearatio<27:

cv2.putText(frame,'3',(0,50), font, 2, (0,0,255),

3, cv2.LINE\_AA)

else:

cv2.putText(frame,'ok',(0,50), font, 2, (0,0,255),

3, cv2.LINE\_AA)

elif l==4:

cv2.putText(frame,'4',(0,50), font, 2, (0,0,255), 3, cv2.LINE\_AA)

elif l==5:

cv2.putText(frame,'5',(0,50), font, 2, (0,0,255), 3, cv2.LINE\_AA)

elif l==6:

cv2.putText(frame,'reposition',(0,50), font, 2,

(0,0,255), 3, cv2.LINE\_AA)

else :

cv2.putText(frame,'reposition',(10,50), font, 2,

(0,0,255), 3, cv2.LINE\_AA)

#show the windows cv2.imshow('mask',mask) cv2.imshow('frame',frame)

except:

pass

k = cv2.waitKey(5) & 0xFF if k == 27:

break

cv2.destroyAllWindows() cap.release()

1. CONCLUSION

In this paper varios methods are discussed for various gesture recognition and an algorithm as well as a software is presented to identify some of the common gestures including detection of numbers and hand signs.

The selection of specific algorithm for recognition depends on the application needed. In this work

application areas for the gestures system are presented. Explanation of gesture recognition issues,

detail discussion of recent recognition systems are given as well. Summary of some selected

systems are listed as well.

1. ACKNOWLEDGMENT

We are Thankful to members of DESH department who cooperated with us and assisted us through every hurdle. We are grateful to Prof. Dr. CM Mahajan Sir, HOD DESH for his steady commitment and backing, last but not the least we would like to thank Ms. Sharda Mhase for giving us this opportunity and guiding us throughout the process.

1. REFERENCES
   1. WEBSITES

[HTTPS://GOGUL09.GITHUB.IO/SOFTWARE/HAND-](https://gogul09.github.io/software/hand-gesture-recognition-p1) [GESTURE-RECOGNITION-P1](https://gogul09.github.io/software/hand-gesture-recognition-p1)

[HTTP://WWW.NITC.AC.IN/ELECTRICAL/IPG/PYTHON/O](http://www.nitc.ac.in/electrical/ipg/python/opencv-python-tutroals.pdf) [PENCV-PYTHON-TUTROALS.PDF](http://www.nitc.ac.in/electrical/ipg/python/opencv-python-tutroals.pdf)

[HTTPS://DOCS.OPENCV.ORG/TRUNK/D1/DC5/TUTORIA](https://docs.opencv.org/trunk/d1/dc5/tutorial_background_subtraction.html) [L\_BACKGROUND\_SUBTRACTION.HTML](https://docs.opencv.org/trunk/d1/dc5/tutorial_background_subtraction.html)

[HTTPS://YOUTU.BE/V-XCMSYLZJA](https://youtu.be/v-XcmsYlzjA)

* 1. RESEARCH PAPERS

[HTTPS://WWW.RESEARCHGATE.NET/PUBLICATION/29](https://www.researchgate.net/publication/291051102_Hand_Gesture_Recognition_A_Literature_Review) [1051102\_HAND\_GESTURE\_RECOGNITION\_A\_LITE](https://www.researchgate.net/publication/291051102_Hand_Gesture_Recognition_A_Literature_Review) [RATURE\_REVIEW](https://www.researchgate.net/publication/291051102_Hand_Gesture_Recognition_A_Literature_Review)

[HTTPS://LINK.SPRINGER.COM/CHAPTER/10.1007/3-](https://link.springer.com/chapter/10.1007/3-540-46616-9_10) [540-46616-9\_10](https://link.springer.com/chapter/10.1007/3-540-46616-9_10)

[HTTPS://IEEEXPLORE.IEEE.ORG/ABSTRACT/DOCUMEN](https://ieeexplore.ieee.org/abstract/document/671007) [T/671007](https://ieeexplore.ieee.org/abstract/document/671007)