# School of Computer Science and Engineering

## ARTIFICIAL INTELIGENCE PROJECT REPORT

(Project Term Jan – May, 2023)

## *HAND GESTURE RECOGNITION*


**Submitted by**

| NAME OF STUDENTS | REGISTRATION NUMBER | ROLL NO |
|---|---|---|
| Rishu Raj | 12204901 | RK21SPA39 |
| Sofiya khan | 12113394 | RK21SPA03 |
| Yallakkagari Vinay Kumar | 12107392 | RK21SPA23 |

**Submitted to :** ***Dr. Gurleen Kaur Walia***


*Transforming Education Transforming India*

# INDEX

# 1) INTRODUCTION: -

## 1.1) Hand Gesture Recognition in AI

Hand gesture recognition is a technology that enables computers to interpret human hand gestures as commands. It has been an active area of research in artificial intelligence (AI) for several years. The technology has a wide range of applications, including gaming, virtual reality, sign language interpretation, and human-robot interaction.

This is an area with many different possible applications, giving users a simpler and more natural way to communicate with robot's/systems interfaces, without the need for extra devices. So, the primary goal of gesture recognition research applied to Human-Computer Interaction (HCI) is to create systems, which can identify specific human gestures and use them to convey information or controlling devices. For that, vision-based hand gesture interfaces require fast and extremely robust hand detection, and gesture recognition in real time.

This paper presents a solution, generic enough, with the help of machine learning algorithms, allowing its application in a wide range of human-computer interfaces, for real-time gesture recognition. Experiments carried out showed that the system was able to achieve an accuracy of 99.4% in terms of hand posture recognition and an average accuracy of 93.72% in terms of dynamic gesture recognition.

To validate the proposed framework, two applications were implemented. The first one is a real-time system able to help a robotic soccer referee judge a game in real time. The prototype combines a vision-based hand gesture recognition system with a formal language definition, the Referee CommLang, into what is called the Referee Command Language Interface System (ReCLIS).

In recent years, there have been significant advancements in hand gesture recognition technology, thanks to the increasing availability of powerful machine learning algorithms and the availability of large datasets. Convolutional neural networks (CNNs) have emerged as the most popular approach for hand gesture recognition, achieving state-of-the-art accuracy rates.

One of the key challenges in hand gesture recognition is the variability of hand shapes and movements across different users. To address this challenge, researchers have developed techniques that involve data augmentation, which involves artificially generating new data from existing data to increase the size of the training dataset. Another approach is to use transfer learning, which involves using pre-trained models to improve the accuracy of hand gesture recognition.

Despite these advancements, hand gesture recognition in AI still faces some challenges. One of the challenges is the need for large amounts of training data to achieve high accuracy rates. Another challenge is the need for real-time processing, which can be computationally expensive.

Overall, hand gesture recognition in AI has made significant progress in recent years and has a wide range of potential applications. As machine learning algorithms continue to improve and more data becomes available, we can expect to see even more accurate and sophisticated hand gesture recognition systems in the future.

### 1.2) History of Hand Gesture Recognition

The history of hand gesture recognition can be traced back to the early 1980s when researchers began exploring the use of computer vision techniques to interpret hand gestures. One of the earliest systems was developed by MIT researcher Rosalind Picard in 1991, which used a glove with sensors to recognize hand gestures.

In the following years, researchers continued to explore different approaches to hand gesture recognition, including using cameras and image processing algorithms. However, early systems were limited by the processing power of computers at the time and the lack of large datasets for training machine learning algorithms.

It wasn't until the mid-2000s that hand gesture recognition technology began to make significant progress. With the availability of more powerful computers and larger datasets, researchers were able to develop more accurate and sophisticated algorithms for hand gesture recognition.

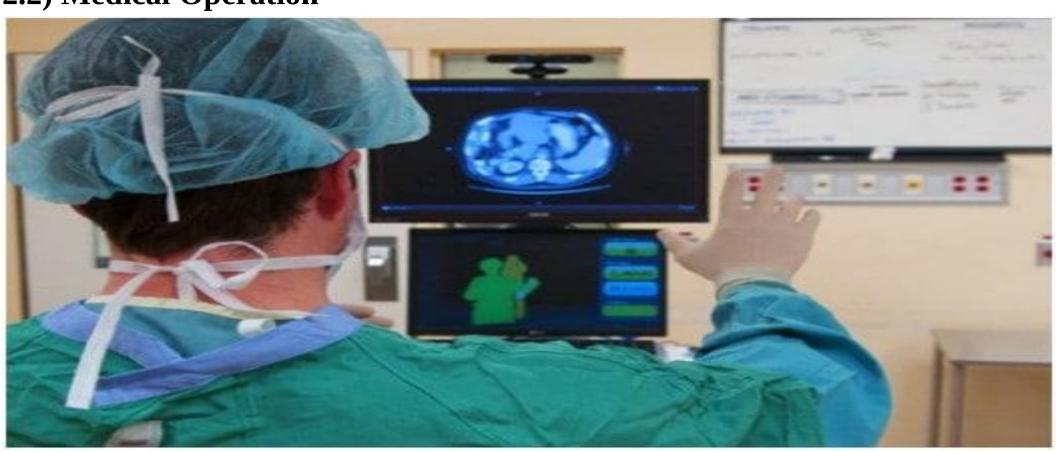
Since then, hand gesture recognition technology has continued to advance rapidly, with new techniques such as convolutional neural networks (CNNs) and transfer learning leading to even higher accuracy rates. Today, hand gesture recognition is used in a wide range of applications, from gaming and virtual reality to medical applications and security systems.

## 2) DIFFERENT APPLICATION OF AI IN HAND GESTURE RECOGNITION: -

**2.1) Talking to computer**

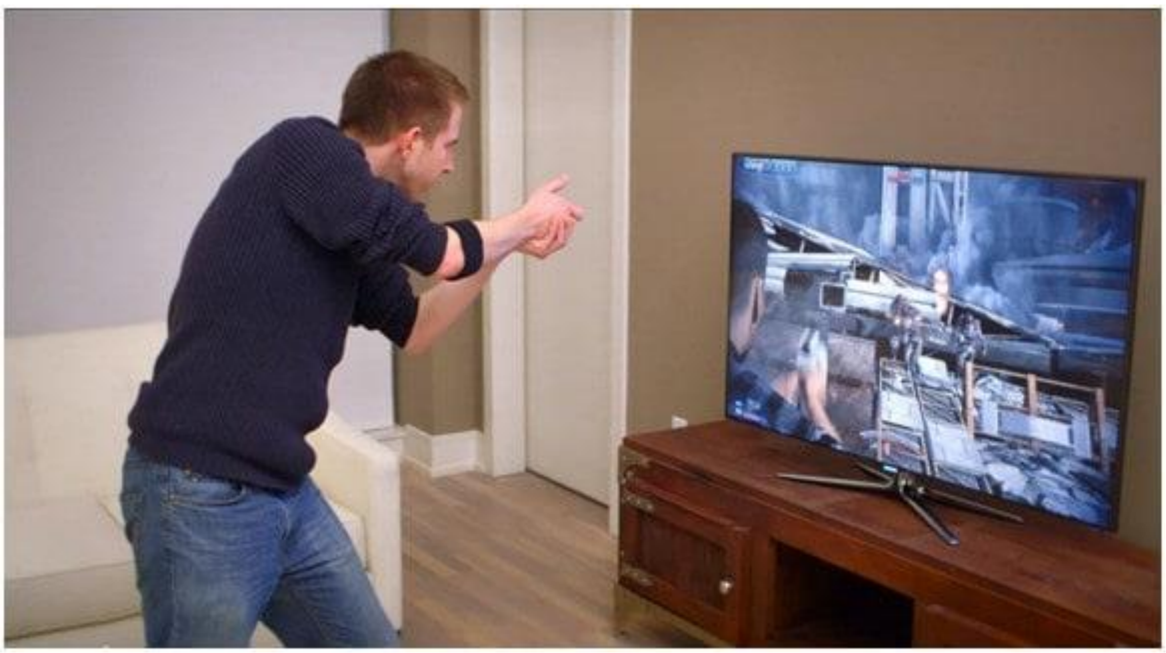
Imagine a world in which a person putting together a presentation can add a quote or move an image with a flick of the wrist instead of a click of a mouse. A future in which we can easily interact in virtual reality much as we do in actual reality, using our hands for small, sophisticated movements like picking up a tool, pushing a button or squeezing a soft object in front of us. This kind of technology is still evolving. But the computer scientists and engineers who are working on these projects say they believe they are on the cusp of making hand and gesture recognition tools practical enough for mainstream use, much like many people now use speech recognition to dictate texts or computer vision to recognize faces in photos.

**2.2) Medical Operation**



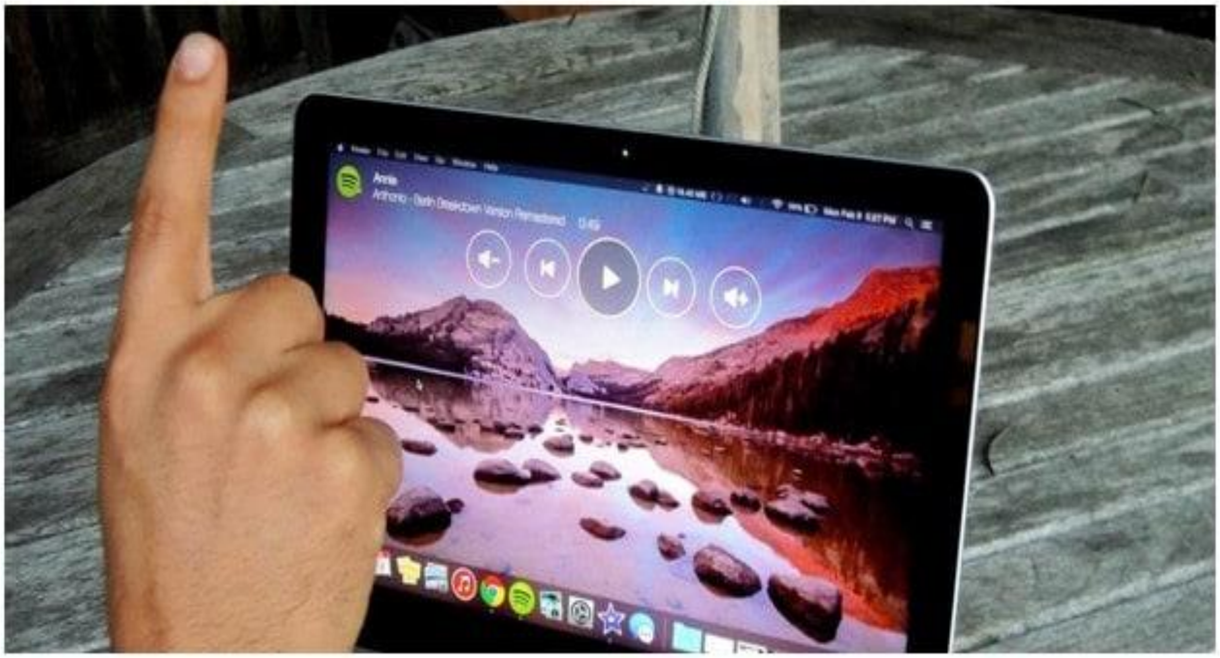
Gestures can be used to control the distribution of resources in hospitals, interact with medical instrumentation, control visualization displays, and help handicapped users as part of their rehabilitation therapy. Some of these concepts have been exploited to improve medical procedures and systems; for example, a technology which satisfied the "come as you are" requirement, where surgeons control the motion of a laparoscope by making appropriate facial gestures without hand or foot switches or voice input. Simply hand gestures into doctor-computer interfaces, describing a computer-vision system that enables surgeons to perform standard mouse functions, including pointer movement and button presses, with hand gestures that satisfy the "intuitiveness" requirement. Hand gesture recognition technology can also be used in medical applications, such as physical therapy, to monitor and track the movements of patients. This can help healthcare professionals to develop more effective treatment plans and improve patient outcomes.

### 2.3) Gesture-based Gaming control



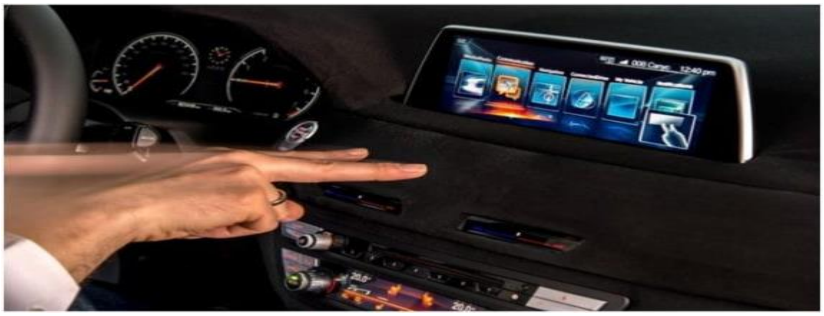
Computer games are a particularly technologically promising and commercially rewarding arena for innovative interfaces due to the entertaining nature of the interaction. Users are eager to try new interface paradigms since they are likely immersed in a challenging game-like environment. In a multi-touch device, control is delivered through the user's fingertips. Which finger touches the screen is irrelevant; most important is where the touch is made and the number of fingers used. In computer-vision-based, hand-gesture-controlled games, the system must respond quickly to user gestures, the "fast-response" requirement. In games, computer-vision algorithms must be robust and efficient, as opposed to applications (such as inspection systems) with no real-time requirement, and where recognition performance is the highest priority. Research efforts should thus focus on tracking and gesture/posture recognition with high-frame-rate image processing.

**2.4) Hand gesture to control the home appliances like MP3 player, TV: -**



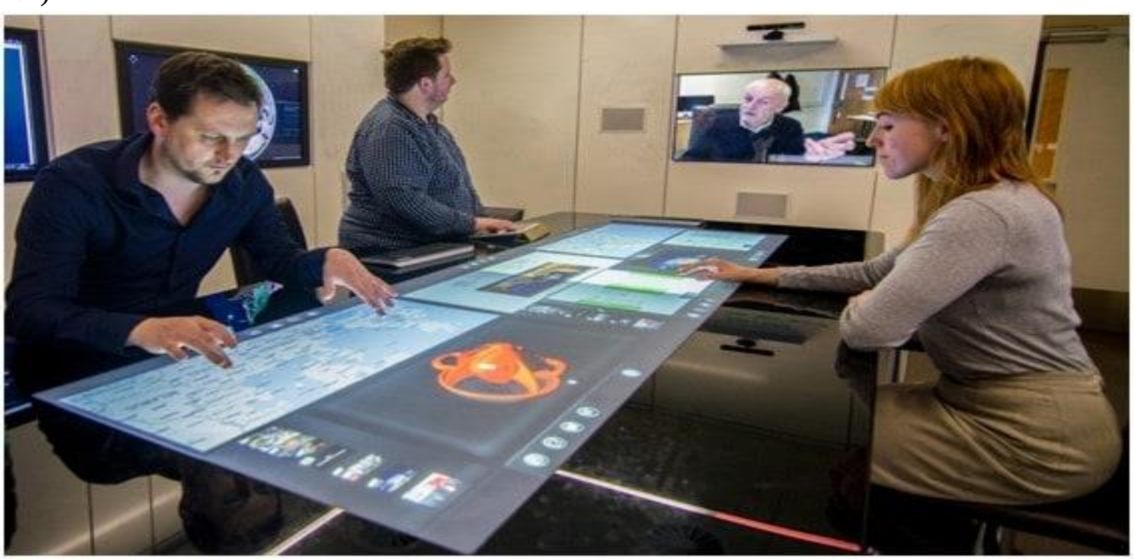
Hand gesture-based electronic device control is gaining more importance nowadays. Most electronic devices focus on the hand gesture recognition algorithm and the corresponding user interface. Hand Gesture Based Remote is a device to replace all other remotes used in households and perform all their functions. Normally in homes, remotes are used for appliances like TV, CD player, Air Conditioner, DVD Player and Music System. Remotes are also used for lights ON/OFF control, Door Opener, etc. All these devices can be controlled by one Universal Remote. Though the technology is synchronized for all remotes (Infrared Transmission and ON/OFF modulation in the range of 32-36 kHz), there is no agreed convention on code format for data transmission. Communication between remote and appliances is established by following a predefined code.

**2.5) Gesture control car Driving**



It’s as simple as it sounds: you use your hands to control the functions in your car and no longer have to look away from the road. Though, of course, there are some nifty technologies working behind the scenes to achieve this feat. One of them is state-of-the-art eye-tracking cameras that monitor where the driver’s eyes focus, and the second is 3D hand gesture recognition sensors that read the movement of their hand. So a driver could look with their eyes to choose the radio and then use a movement of their hand to change the station, without taking their eyes off the road. Hand movements used to control the system are ones we already use naturally on a daily basis; for example, raising or lowering the hand, pointing, swiping left or right, rotating clockwise or anti-clockwise, and pinching or spreading. This would enable you to perform actions such as scrolling through a phone contact list, changing the destination on your Sat-Nav, returning to a previous song or increasing the temperature in the vehicle.

2.6) **Communication**



Virtual reality and immersive reality systems are computer-generated environments that replicate a scenario or situation, either inspired by reality or created out of imagination. These reality systems, often called hybrid realities to allow the stimulation of the user’s physical presence via user interaction and movement to create an all-encompassing sensory experience. This may include the senses of sight, hearing, touch and even smell. The interaction of a user with a VR environment is limited to the use of various devices or VR head-mounted displays which often require the use of pointing devices. However, for virtual reality, commanding devices which can be manipulated unseen are much preferred for example voice commands, lip-reading, interpretation of facial expression and recognition of hand gestures.

## 3) IMPACT OF AI IN HAND GESTURE RECOGNITION: -

Artificial Intelligence (AI) has had a significant impact on hand gesture recognition technology in recent years. With the increasing availability of powerful machine learning algorithms and large datasets, researchers have been able to develop more accurate and sophisticated systems for interpreting human hand gestures.

One of the key advantages of using AI in hand gesture recognition is that it enables systems to learn from large amounts of data and improve their accuracy over time. Machine learning algorithms, such as convolutional neural networks (CNNs), can be trained on large datasets of hand gestures, allowing them to recognize a wide range of gestures with high accuracy rates.

Another advantage of using AI in hand gesture recognition is that it enables systems to adapt to different users and environments. By analyzing data from different users and environments, machine learning algorithms can learn to recognize variations in hand shapes and movements, improving the accuracy of the system.

AI has also enabled the development of real-time hand gesture recognition systems, which can interpret hand gestures in real-time and provide immediate feedback. This is particularly useful in applications such as gaming and virtual reality, where users need to interact with virtual objects in real-time.

Hand gesture recognition technology has a wide range of applications, including gaming, virtual reality, sign language interpretation, and human-robot interaction. With the increasing availability of AI-powered hand gesture recognition systems, these applications are becoming more sophisticated and user-friendly.

However, there are also some challenges associated with using AI in hand gesture recognition. One of the challenges is the need for large amounts of training data to achieve high accuracy rates. Another challenge is the need for real-time processing, which can be computationally expensive.

Despite these challenges, the impact of AI in hand gesture recognition has been significant, enabling more natural and intuitive interactions between humans and machines. As machine learning algorithms continue to improve and more data becomes available, we can expect to see even more accurate and sophisticated hand gesture recognition systems in the future.

One of the key advantages of using AI in hand gesture recognition is that it enables systems to learn from large amounts of data and improve their accuracy over time. Machine learning algorithms, such as convolutional neural networks (CNNs), can be trained on large datasets of hand gestures, allowing them to recognize a wide range of gestures with high accuracy rates. As more data becomes available, these algorithms can continue to learn and improve their accuracy, making them even more effective at recognizing hand gestures.

Another advantage of using AI in hand gesture recognition is that it enables systems to adapt to different users and environments. By analyzing data from different users and environments, machine learning algorithms can learn to recognize variations in hand shapes and movements, improving the accuracy of the system. This is particularly important in applications such as sign language interpretation, where different users may have different signing styles.

AI has also enabled the development of real-time hand gesture recognition systems, which can interpret hand gestures in real-time and provide immediate feedback. This is particularly useful in applications such as gaming and virtual reality, where users need to interact with virtual objects in real-time. Real-time hand gesture recognition can also be used in human-robot interaction, enabling more natural and intuitive interactions between humans and robots.

## 4) Future Scope: -

The future scope of the impact of AI in hand gesture recognition is vast, with the potential to revolutionize many aspects of our lives. With the increasing availability of powerful machine learning algorithms and large datasets, researchers are developing more accurate and sophisticated systems for interpreting human hand gestures. One of the key areas for future development is the use of AI in sign language interpretation. Sign language is a complex and nuanced language, with many variations in hand shapes and movements. AI-powered hand gesture recognition systems have the potential to improve the accuracy and speed of sign language interpretation, making it more accessible for people who are deaf or hard of hearing.

Another area for future development is the use of hand gesture recognition technology in healthcare. Hand gesture recognition can be used to monitor and track the movements of patients, helping healthcare professionals to develop more effective treatment plans and improve patient outcomes. For example, hand gesture recognition technology can be used in physical therapy to monitor a patient's movements and provide real-time feedback on their progress. In the field of human-robot interaction, AI-powered hand gesture recognition systems have the potential to enable more natural and intuitive interactions between humans and robots. This could have significant implications for industries such as manufacturing and logistics, where robots are increasingly being used to perform tasks traditionally performed by humans.

AI-powered hand gesture recognition technology also has the potential to transform the way we interact with smart devices in our homes. By using hand gestures to control devices such as lights, thermostats, and entertainment systems, we can create a more natural and intuitive way of interacting with technology. This could make smart homes more accessible and user-friendly for people with mobility issues or those who want to avoid touching surfaces for hygiene reasons.

The future scope of AI in hand gesture recognition also includes the development of more sophisticated virtual reality environments. Hand gesture recognition can be used to interact with virtual objects and navigate through virtual environments, providing a more immersive and interactive experience for users. This could have significant implications for industries such as gaming, education, and training. As machine learning algorithms continue to improve and more data becomes available, we can expect to see even more accurate and sophisticated hand gesture recognition systems in the future. These systems will be able to recognize a wider range of hand gestures with higher accuracy rates, making them even more effective at interpreting human gestures.

However, there are also some challenges associated with the future development of AI in hand gesture recognition. One of the challenges is the need for large amounts of training data to achieve high accuracy rates. This can be particularly challenging in applications such as sign language interpretation, where there is a wide range of signing styles and variations in hand shapes and movements. Another challenge is the need for real-time processing, which can be computationally expensive.

In conclusion, the future scope of the impact of AI in hand gesture recognition is vast, with the potential to transform many aspects of our lives. From sign language interpretation and healthcare to human-robot interaction and smart homes, AI-powered hand gesture recognition technology has the potential to make our lives more accessible, user-friendly, and interactive. As machine learning algorithms continue to improve and more data becomes available, we can expect to see even more accurate and sophisticated hand gesture recognition systems in the future.

## 5) Conclusion: -

Hand gestures are a powerful way for human communication, with lots of potential applications in the area of human computer interaction. Vision-based hand gesture recognition techniques have many proven advantages compared with traditional devices.

However, hand gesture recognition is a difficult problem and the current work is only a small contribution towards achieving the results needed in the field. The main objective of this work was to study and implement solutions that could be generic enough, with the help of machine learning algorithms, allowing its application in a wide range of human-computer interfaces, for online gesture and posture recognition. To achieve this, a set of implementations for processing and retrieving hand user information, learn statistical models and able to do online classification were created. The final prototype is a generic solution for a vision-based hand gesture recognition system, which is able to integrate posture and gesture classification and that, can be integrated with any human-computer interface.

The implemented solutions, based on supervised learning algorithms, are easily configured to process new hand features or to learn different hand postures and dynamic gestures, while creating statistical models that can be used in any real-time user interface for online gesture classification. For the problem of hand posture classification, hand features that give good classification results were identified, being at the same time simple in terms of computational complexity, for use in any real-time application. The selected features were tested with the help of the Rapid Miner tool for machine learning and data mining. That way, it was possible to identify a learning algorithm that was able to achieve very good results in terms of pattern classification, and that was the one used in the final solution. For the case of dynamic gesture recognition, the choice fell on Hidden Markov Models, due to the nature of the data, gestures, which are time-varying processes. This type of models has proven to be very effective in other areas of application, and had already been applied successfully to the problem of gesture recognition. The evaluation of the trained gestures with the implemented prototypes proved that, it was possible to successfully integrate static and dynamic gestures with the generic framework and use them for human / computer interaction.

## 6) CODE: -

**RECOGNITION**

```
import cv2
from cvzone.HandTrackingModule import HandDetector
from cvzone.ClassificationModule import Classifier
import numpy as np
import math

cap = cv2.VideoCapture(0)
detector = HandDetector(maxHands=1)
classifier = Classifier("Model/keras_model.h5", "Model/labels.txt")

offset = 20
imgSize = 300
```

```
folder = "Data/C"
counter = 0

labels = ["A", "B", "C","Looser","All the best","Call"]

while True:
    success, img = cap.read()
    img = cv2.flip(img,1)
    imgOutput = img.copy()
    hands, img = detector.findHands(img)
    if hands:
        hand = hands[0]
        x, y, w, h = hand['bbox']

        imgWhite = np.ones((imgSize, imgSize, 3), np.uint8) * 255
        imgCrop = img[y - offset:y + h + offset, x - offset:x + w + offset]

        imgCropShape = imgCrop.shape

        aspectRatio = h / w

        if aspectRatio > 1:
            k = imgSize / h
            wCal = math.ceil(k * w)
            imgResize = cv2.resize(imgCrop, (wCal, imgSize))
            imgResizeShape = imgResize.shape
            wGap = math.ceil((imgSize - wCal) / 2)
            imgWhite[:, wGap:wCal + wGap] = imgResize
            prediction, index = classifier.getPrediction(imgWhite, draw=False)
            print(prediction, index)

        else:
            k = imgSize / w
            hCal = math.ceil(k * h)
```

```
            imgResize = cv2.resize(imgCrop, (imgSize, hCal))
            imgResizeShape = imgResize.shape
            hGap = math.ceil((imgSize - hCal) / 2)
            imgWhite[hGap:hCal + hGap, :] = imgResize
            prediction, index = classifier.getPrediction(imgWhite, draw=False)

        cv2.rectangle(imgOutput, (x - offset, y - offset - 50),
                      (x - offset + 90, y - offset - 50 + 50), (0, 255, 0), cv2.FILLED)
        cv2.putText(imgOutput, labels[index], (x, y - 26), cv2.FONT_HERSHEY_COMPLEX,
1.7, (255, 255, 255), 2)
        cv2.rectangle(imgOutput, (x - offset, y - offset),
                      (x + w + offset, y + h + offset), (255, 0, 255), 4)

        cv2.imshow("ImageCrop", imgCrop)
        cv2.imshow("ImageWhite", imgWhite)

    cv2.imshow("Image", imgOutput)
    cv2.waitKey(1)
```

**DATA COLLECTION**

```
import cv2
from cvzone.HandTrackingModule import HandDetector
import numpy as np
import math

cap = cv2.VideoCapture(0)
detector = HandDetector(maxHands=1)

offset = 20
imgSize = 300

folder = "Data/FK"
counter = 0
```

```
while True:
    success, img = cap.read()
    img = cv2.flip(img,1)
    hands, img = detector.findHands(img)
    if hands:
        hand = hands[0]
        x, y, w, h = hand['bbox']

        imgWhite = np.ones((imgSize, imgSize, 3), np.uint8) * 255
        imgCrop = img[y - offset:y + h + offset, x - offset:x + w + offset]

        imgCropShape = imgCrop.shape

        aspectRatio = h / w

        if aspectRatio > 1:
            k = imgSize / h
            wCal = math.ceil(k * w)
            imgResize = cv2.resize(imgCrop, (wCal, imgSize))
            imgResizeShape = imgResize.shape
            wGap = math.ceil((imgSize - wCal) / 2)
            imgWhite[:, wGap:wCal + wGap] = imgResize

        else:
            k = imgSize / w
            hCal = math.ceil(k * h)
            imgResize = cv2.resize(imgCrop, (imgSize, hCal))
            imgResizeShape = imgResize.shape
            hGap = math.ceil((imgSize - hCal) / 2)
            imgWhite[hGap:hCal + hGap, :] = imgResize

        cv2.imshow("ImageCrop", imgCrop)
        cv2.imshow("ImageWhite", imgWhite)

    cv2.imshow("Image", img)
```

```
key = cv2.waitKey(1)
if key == ord("s"):
    counter += 1
    cv2.imwrite(f'{folder}/Image_{counter}.jpg', imgWhite)
    print(counter)
```

**OUTPUT OF THE CODE**