**Mini Project Report on**



**MOTION GESTURE RECOGNITION**



**Submitted in partial fulfillment of the requirement for the award of the degree of**

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE & ENGINEERING**

**Submitted by:**

**RUUDRA AMOLA**  **Univ Roll No: 2016972**

***Under the Mentorship of***

**Dr. Parul Madan**

**Assistant Professor (CSE)**



**Department of Computer Science and Engineering**

**Graphic Era (Deemed to be University)**

**Dehradun, Uttarakhand**

**June 2023**



**CANDIDATE’S DECLARATION**

I hereby certify that the work which is being presented in the project report entitled

**“Motion Gesture Recognition”** in partial fulfillment of the requirements for the 6th semester of the Degree of Bachelor of Technology in Computer Science and Engineeringof the Graphic Era (Deemed to be University), Dehradun shall be carried out by the under the mentorship of **Dr. Parul Madan, Assistant Professor**, Department of Computer Science and Engineering, Graphic Era (Deemed to be University), Dehradun.



RUUDRA AMOLA 2016972

Name University Roll Number **signature**

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Chapter No.** | **Description** | **Page No.** |
| Chapter 1 | Introduction | **1** |
| Chapter 2 | Literature Survey | **3** |
| Chapter 3 | Methodology | **5** |
| Chapter 4 | Result and Discussion | **8** |
| Chapter 5 | Conclusion and Future Work | **10** |
|  |  |  |

**Chapter 1**

**Introduction**

**Problem Statement:**

The main purpose of this project is to build an application of motion gesture recognition technology. Further implemented few basic hand gestures using Open CV in Python.

**What is Motion Gesture Recognition Technology?**

Motion gesture recognition technology is a type of human-computer interaction (HCI) that allows users to interact with devices by making gestures with their hands or bodies. This technology is becoming increasingly popular as it offers a more natural and intuitive way to interact with devices.

There are two main types of motion gesture recognition technology: **contact-based** and **contactless**. Contact-based gesture recognition technology uses sensors that are worn on the body, such as accelerometers, gyroscopes, and magnetometers. These sensors track the movement of the body and can be used to recognize gestures such as waving, pointing, and clapping. Contactless gesture recognition technology uses cameras to track the movement of the hands and body. This technology is often used in devices such as smartphones and tablets.

**SOFTWARES AND TOOLS/MODULES USED:**

* **Language Used-** *Python* (Version 3.11.1)

Python is a high-level, general-purpose programming language that is widely used for web development, data analysis, artificial intelligence, scientific computing, and many other applications. It is known for its simplicity, readability, and flexibility, as well as its comprehensive standard library and large developer community. Python supports object-oriented, imperative, and functional programming styles, and it can be used to build standalone programs as well as to create scripts that can be run by other programs.

* **Modules Used-**

**A black screen with white text

Description automatically generated**

1. ***Cv2* module:** The cv2 module is a part of the Python programming language's OpenCV library ("Open Computer Vision"), which is a collection of tools for working with visual data. OpenCV is a popular library to read and write image files, access the camera on your computer or mobile device, or perform image processing operations.
2. ***NumPy* module:** NumPy is a library for the Python programming language, and it’s specifically designed to help you work with data. With NumPy, you can easily create arrays, which is a data structure that allows you to store multiple values in a single variable. NumPy also includes several functions that make it easy to perform mathematical operations on arrays.
3. ***math* module:** The math module in Python is a built-in module that provides a set of mathematical functions and constants. It is a standard module. The math module includes functions for performing basic mathematical operations, such as addition, subtraction, multiplication, and division. It also includes functions for performing more advanced mathematical operations, such as trigonometric functions, logarithmic functions, and exponential functions.

* **IDE used-** *VS Code* (Version1.8)

Visual Studio Code (VS Code) is a free and open-source code editor developed by Microsoft for Windows, Linux, and macOS. It is built on top of the open-source Monaco editor. VS Code is a powerful code editor that can be used for a wide variety of programming languages, including JavaScript, TypeScript, Python, C++, C#, Java, and PHP. It includes several features that make it a popular choice for developers.

**Chapter 2**

**Literature Survey**

A literature survey is a critical summary of what has been published on a topic by accredited researchers. It is a way to get a broad overview of the state of the art in a particular field and to identify key issues, trends, and gaps in the research. In this case, a literature survey for motion gesture recognition using OpenCV would involve reviewing the existing research on motion gesture recognition systems in general.

**Introduction**

Motion gesture recognition technology is a type of human-computer interaction (HCI) that allows users to interact with devices by making gestures with their hands or bodies. This technology is becoming increasingly popular as it offers a more natural and intuitive way to interact with devices.

**History**

The first attempts at motion gesture recognition technology date back to the early 1980s. However, it was not until the late 1990s that this technology began to gain widespread attention. This was due in part to the development of new sensors, such as accelerometers and gyroscopes, which made it possible to track the movement of the body more accurately.

In the early 2000s, motion gesture recognition technology began to be used in a variety of commercial products, such as the Microsoft Kinect and the Leap Motion Controller. These products showed the potential of motion gesture recognition technology for a wide range of applications, such as gaming, virtual reality, and home automation.

**Applications**

* **Gaming:** Motion gesture recognition technology is often used in gaming to control characters and objects in the game. This allows for a more natural and intuitive way to interact with the game.
* **Virtual reality:** Motion gesture recognition technology is a key component of virtual reality (VR) systems. VR systems use cameras to track the movement of the user's head and hands, which allows the user to interact with the VR environment in a natural way.
* **Home automation:** Motion gesture recognition technology can be used to control home automation devices such as lights, thermostats, and locks. This allows users to control their home environment with simple gestures.
* **Sign language recognition:** Motion gesture recognition technology can be used to recognize sign language. This allows people who are deaf or hard of hearing to communicate more effectively.

**Challenges**

Despite the advances that have been made in motion gesture recognition technology, there are still some challenges that need to be addressed. Some of the challenges include:

* **Accuracy:** The accuracy of gesture recognition systems can still vary depending on the environment and the user.
* **Robustness:** Gesture recognition systems need to be robust to noise and other environmental factors.
* **Interpretability:** Gesture recognition systems need to be able to interpret the meaning of gestures in a meaningful way.

**Future trends**

The future of motion gesture recognition technology is very promising. As the technology continues to develop, it is likely that we will see even more innovative and exciting applications for this technology. Some of the potential future trends in motion gesture recognition technology include:

* **Improved accuracy:** Gesture recognition systems are likely to become more accurate as the technology continues to develop.
* **Greater robustness:** Gesture recognition systems are likely to become more robust to noise and other environmental factors.
* **Improved interpretability:** Gesture recognition systems are likely to become better at interpreting the meaning of gestures in a meaningful way.
* **New applications:** As the technology continues to develop, we are likely to see new and innovative applications for motion gesture recognition technology.

**Chapter 3**

**Methodology**

This project mainly focuses on practical implementation of motion gesture recognition technology to track hand movement and recognize a gesture from that in front of the webcam of a specific hand shape and detects it.

It has a frame box which appears once the open CV module invokes the system webcam and then it recognizes the hand gesture from the web cam. The main library used in the project is OpenCV.

**About OpenCV:**

The concept of OpenCV was put forth by Gary Bradski which had the ability to perform on multi-level framework. OpenCV has several significant abilities as well as utilities which appears from the outset. The OpenCV helps in recognizing the frontal face of the person and creates XML documents for several areas such as the parts of the body.

The minimum requirements for the software would be python along with OpenCV and the required dataset. The minimum requirements for the hardware would be intel i3 or any processor above it and 4 core CPU. Operating systems of windows 10 will be sufficient and 8GB RAM required.

**Brief understanding of the Code:**

Hand gesture recognition system has developed excessively in the recent years, reason being its ability to cooperate with machine successfully. Gestures are considered as the most natural way for communication between humans and PCs in virtual framework. We often use hand gestures to convey something as it is non-verbal communication which is free of expression. In this project, our PC's camera records a live video, from which a preview is taken with the assistance of its functionalities or activities.

First, we capture the video and then define the region of interest. In the region of interest, it converts all the BGR colours to HSV colours.

hsv = cv2.cvtColor(roi, cv2.COLOR\_BGR2HSV)

The **HSV** (which stands for **Hue Saturation Value**) scale provides a numerical readout of your image that corresponds to the color names contained therein. Hue is measured in degrees from 0 to 360. The Hue in HSV represents the color, Saturation in HSV represents the greyness, and Value in HSV represents the brightness. Whenever we want to solve problems related to object detection, it is necessary to use HSV and find the range of HSV. The Hue, Saturation, and Value in HSV have their own range of values. The Hue range in HSV is [0,179], the Saturation range in HSV is [0,255] and the Value range in HSV is [0,255].

There is also an upper bound and lower bound range for a range of each color in HSV. The HSV or Hue, Saturation, and value of a given object provide better performance when compared to RGB or Red, Blue, and Green color space and hence it is used widely in the area of computer vision.

After that I have defined the skin color range in HSV anything that is of skin color will be taken as one or white and any other color will be taken as zero or black. Then I have just dilated and blurred the image to decrease any amount of noise present in the image.

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

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

After that I have found **contours** in that image. **Contours** are just the outlines of any area that is being shown that region of interest. **Contours** are defined as the line joining all the points along the boundary of an image that are having the same intensity. Contours come handy in shape analysis, finding the size of the object of interest, and object detection. OpenCV has findContours() function that helps in extracting the contours from the image.

contours,hierarchy=cv2.findContours(mask,cv2.RETR\_TREE,cv2.CHAIN\_APPROX\_SIMPLE)

Then I have defined the **convex hull**. A convex hull of an object is the minimum boundary that can completely enclose or wrap the object (or contour of that object). Here, Convex hull is this green outline around my hand.

Then I have taken the area ratio, every gesture will have its own unique area ratio that is the basis on which gestures are differentiated.

After that **defects** in convex hull are to be found. Defects are regions that are not covered by the hand in the convex hull. Any deviation of the contour from the convex hull is known as convexity defect. OpenCV comes with a ready-made function to find this, cv2.convexityDefects(). It takes as input the contour and corresponding hull indices and returns an array containing convexity defects as output. It returns an array where each row contains these values - [ start point, end point, farthest point, approximate distance to farthest point ].

 #Print corresponding gestures which are in their ranges

        font = cv2.FONT\_HERSHEY\_SIMPLEX

        if l==1:

            if areacnt < 2000:

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

            else:

                if arearatio < 12:

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

                elif arearatio < 17.5:

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

                else:

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

        elif l==2:

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

        elif l==3:

              if arearatio<27:

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

              else:

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

        elif l==4:

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

        elif l==5:

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

        elif l==6:

            cv2.putText(frame, 'please re-position the hand', (0,50), font, 2, (0,0,255), 3, cv2.LINE\_AA)

        else:

            cv2.putText(frame, 'please re-position the hand', (10,50), font, 2, (0,0,255), 3, cv2.LINE\_AA)

If there is nothing in the box or the area covered by hand is too less, then I am displaying “put hand in the box”. We can differentiate three based on area ratio covered to “okay” or “three”. If there are more than 6 defects, then reposition the hand. This may be due to some error or lightning effects of the room.

Then I have just displayed both the python boxes and by pressing escape key the code execution ends.

**Chapter 4**

**Result and Discussion**

A screenshot of a computer

Description automatically generated with medium confidenceA screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated with medium confidenceGraphical user interface, website

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidenceA screenshot of a computer

Description automatically generated with medium confidence

The above screenshots show that the gestures are shown through the webcam using the hands and those gestures are recognized showing specified values in the window.

The system is built to recognize a total of 8 hand gestures out of which 6 have been displayed above in the screen shots.

Motion gesture recognition technology is a rapidly evolving field with a wide range of potential applications. As technology continues to develop, we are likely to see even more innovative and exciting applications for this technology in the future.

The main recognition part is done based on the area of the hand contour that appears in the region of interest and the angle between the fingers.

**Chapter 5**

**Conclusion and Future Work**

Developing recognition systems which are efficient for working under different conditions is tough, but it is more possible because these hurdles exist in the real-world environment. Hand gesture recognition is very significant for human-computer interaction. Gesture recognition appears in many real-life applications, e.g., gesture-based gaming control, medical applications, controlling home appliances, in IoT devices, sign language translations and many more.

**Advantages:**

The advantages of the face recognition system include more natural intuitive interaction, faster task processing, automation of the identity, breach of privacy, enhanced security, more flexibility in systems and many more in day-to-day life.

**Disadvantages:**

Some of the disadvantages are inaccuracy due to the varied working environments of the user, motion gesture systems can be sensitive to noise and other environmental factors too.  The meaning of gestures can be ambiguous. This can make it difficult for gesture recognition systems to interpret the meaning of gestures in a meaningful way.

The future of motion gesture recognition technology is very promising. As the technology continues to develop, it is likely that we will see even more innovative and exciting applications for this technology. Some of the potential future trends in motion gesture recognition technology include:

* **Improved accuracy:** Gesture recognition systems are likely to become more accurate as the technology continues to develop.
* **Greater robustness:** Gesture recognition systems are likely to become more robust to noise and other environmental factors.
* **Improved interpretability:** Gesture recognition systems are likely to become better at interpreting the meaning of gestures in a meaningful way.
* **New applications:** As the technology continues to develop, we are likely to see new and innovative applications for motion gesture recognition technology.

**THANK YOU!**