Volume 3, Issue , 2024
PP 45-52

International Journal of Futuristic Innovation in Arts, Humanities and Management (IJFIAHM)

AIR CANVAS APPLICATION USING OPENCV AND NUMPY IN PYTHON

1st Mr. Pratik Nikhar
pratiknikhar121@gmail.com
Department of Computer Science and Engineering , Shri Sai College of Engineering and Technology, Chandrapur India

2nd Mr. Lowlesh Yadav
lowlesh.yadav@gmail.com
Assistant Professor
Department of Computer Science and Engineering , Shri Sai College of Engineering and Technology,Chandrapur India

3rd Mr. Neehal Jiwane
neehaljiwane@gmail.com
Assistant Professor
Department of Computer Science and Engineering
Shri Sai College of Engineering and Technology, Chandrapur India

ABSTRACT:- Assistant Professor, Department of Computer Science and Engineering , Shri Sai College of Engineering and Technology,Chandrapur India

In recent years, air writing has emerged as an attractive and challenging research area in image processing and pattern recognition. This has the potential to significantly advance automation processes and improve human-machine interfaces in a variety of applications. Numerous studies have been conducted to develop new techniques and methods that can reduce processing time while providing higher detection accuracy. Object tracking is an important task in computer vision. The popularity of object tracking techniques is increasing due to the availability of faster computers, cheaper and better video cameras, and the demand for automated video analysis. In general, video analysis methods include three main steps: detecting objects, tracking their movement from frame to frame, and analyzing their behavior. Object tracking includes four different topics: choosing a suitable representation of an object, choosing a tracking feature, object detection, and object tracking. Object tracking algorithms are widely used in various applications such as real-world automatic surveillance, video profiling, and vehicle navigation. The goal of the project is to develop a motion to text converter that can be used as software for smart wearable devices. It will allow users to type from the air by following the path of their finger using computer vision. The generated text can be used for various purposes such as sending messages and emails. This project is a reporter of occasional gestures and will be a powerful means of communication for the deaf. It is an effective communication method that reduces the use of mobile devices and laptops by eliminating the need for typing.

Keywords : Object Tracking , Computer Vision , Object Detection .

INTRODUCTION:

The digital era has brought a shift from traditional writing to digital art. Digital art is the expression and transmission of art forms in a digital format and relies on modern science and technology to create its distinctive characteristics. Traditional art, on the other hand, refers to art forms created before the advent of digital art. It can be divided into fine arts, sound arts, audiovisual arts, and audiovisual imaginary arts, which includes literature, painting, sculpture, architecture, music, dance, drama, and other works of art. Digital and traditional art are related and interdependent. Social development is driven by the needs of human life rather than human will. The same goes for art. In today's world, digital and traditional art coexist symbiotically. Therefore, it is essential to systematically understand the basic knowledge about the differences between digital and traditional art. The traditional way of writing involves pen and paper or chalk and blackboard methods. The basic goal of digital art is to build a hand gesture recognition system for digital writing. Digital art includes many ways of writing, such as using a keyboard, touch screen, digital pen, stylus, electronic gloves, etc. In this system, we use hand gesture recognition with machine learning algorithms using Python programming to create a natural human-machine interaction. As technology advances, there is a growing need to develop natural "Human Computer Interaction (HCI)" systems to replace traditional systems.

LITERATURE REVIEW:

A] we proposed a system that uses the Kinect sensor to detect hand shape using depth information and color information. However, even with the Kinect sensor, detecting hand movements is still a difficult problem. The Kinect's sensor only has a resolution of 640x480, which makes it difficult to track small objects like fingers.

B] we proposed a method to connect the LED to the user's finger and track the finger using a web camera. The drawn characters are compared with the characters in the database and the alphabet matching the drawn pattern is returned. This method requires a pointed red LED

light source to be attached to your finger. It also assumes that there is no red object other than the LED light in the focus of the webcam.

C] In an advanced partition table interface approach for interaction was proposed. This system uses a video projector and a charge-coupled device (CCD) camera to make desktop applications available to users. In this system, each hand does a different job. The left hand is used to select radial menus and the right hand is used to select objects for manipulation. This is achieved using an infrared camera. However, fingertip determination is computationally expensive, so this system defines a fingertip search window.

PROBLEM DEFINITION:

This project aims to address several important social issues, including:

a) Hearing Impairment: People with hearing loss use sign language to communicate. However, most people in the world cannot understand their feelings and emotions without the intervention of an interpreter.

b) Excessive use of smartphones: Smartphones are portable and easy to use, but they can cause accidents, depression, distraction, and other diseases that humans have yet to discover. Negative cases include life-threatening events.

c) Paper waste: We waste a lot of paper by doodling, writing, drawing, etc. The basic facts are that it takes an average of 5 liters of water to produce one sheet of A4 paper, 93% of the writing comes from trees, and I can list 50% of the commercial uses. Garbage is paper, 25% of landfills are paper and the list goes on. Wasting paper by using water and trees has a negative impact on the environment and creates a lot of waste.Air Writing can quickly solve these problems. It is useful as a communication tool for people with hearing impairments. Annotated text can be presented using AR or converted to speech. You can write quickly in the air and continue your work without distraction. Plus, you don't need paper to write in the air. Everything is stored electronically. "

MAJOR CHALLENGES:

1. Detection Of Fingertip:Existing systems are limited to fingerprint recognition and cannot recognize highlights, colors or other objects. Identifying and characterizing objects such as fingers from RGB images without using depth sensors is a big challenge.

2. Realtime Controlling System:Changing a system from one state to another using hand gestures in real time requires a lot of precision in the code. In addition, users must learn many behaviors to effectively control the application.

3. Pen Up And Down Motion:This system uses an RGB camera to record the user's writing from above. However, due to the lack of depth sensing, the system cannot track the vertical movement of the pen. As a result, the entire path of the fingertip is traced and the resulting image is illogical and not recognized by the model.

METHODOLOGY:

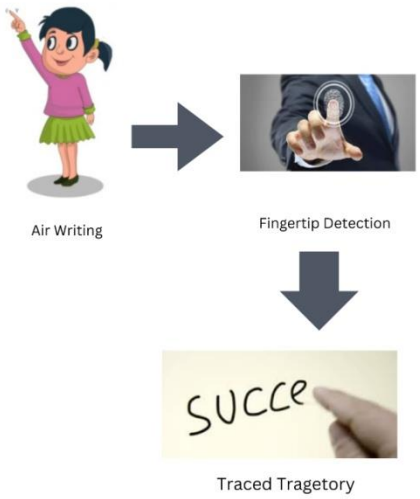

Fig 1: Workflow Of The System

1. Environment Setup: The first step is to set up the required environment for developing the Air Canvas application. This involves installing Python, OpenCV, and NumPy on the system.

2. Webcam Initialization: The next step is to initialize the webcam to capture live video frames. This can be done using OpenCV's VideoCapture() function.

3. Canvas Creation: A blank canvas is created using NumPy arrays. The canvas is initialized with zeros and has the same dimensions as the video frame.

4. Marker Color Detection: The user selects a color for the marker that they will use to draw on the canvas. The program detects this color using OpenCV's inRange() function.

5. Marker Movement Tracking: The program tracks the movement of the marker in real-time using OpenCV's findContours() function. The contours of the marker are detected and used to draw on the canvas.

6. Canvas Display: The canvas is displayed on the screen using OpenCV's imshow() function.

7. Canvas Saving: The user can save the canvas as an image file using OpenCV's imwrite() function.

the methodology involves capturing live video frames from a webcam, detecting the color of a marker, tracking its movement, and drawing on a blank canvas using NumPy arrays. The canvas is displayed on the screen in real-time, and the user can save it as an image file for later use.

ALGORITHM OF WORKFLOW:

Fig 2: Front End Of The System

1. Importing the required libraries: The first step is to import the required libraries such as OpenCV, NumPy, time, etc.
2. Setting up the camera: The next step is to set up the camera and capture the video stream using the VideoCapture() function in OpenCV.
3. Create a window: Using OpenCV's namedWindow() function, a window is created to display the video stream captured by the camera.
4. Initialize variables: Variables are initialized to store previous point, current point and font color.
5. Create a mask: The cv2.inRange() function in OpenCV is used to create a mask to identify the objects to be tracked.
6. Find contours: Contours are found using OpenCV cv2.findContours() function.
7. Filtering contour lines: Contour lines are filtered based on area and length using cv2.contourArea() and cv2.arcLength() OpenCV functions.
8. Draw on canvas: The filtered contour is drawn on the canvas using OpenCV's cv2.line() function.

9. Update variables: The previous point is updated with the current point and the current point is reset.

10. Canvas display: Use the OpenCV imshow() function to display the canvas in a separate window.

11. Exit the program: Press the "q" key to exit the program.

CONCLUSION:

The described system could revolutionize traditional writing methods by eliminating the need to hold a mobile phone to take notes. You can easily do the same on the go, and it's especially useful for people with disabilities and the elderly who have trouble using a keyboard. The system can also be used to control IoT devices and allows for aerial mapping. This will be a great application for smart wearables that will help people interact better with the digital world. Augmented reality can bring your text to life.However, the system has limitations that may be improved in the future. First, using a handwriting recognition engine instead of a character recognition engine allows users to type word by word and type faster. Second, instead of using fingertips, hand gestures with pauses can be used to control the system at runtime in real time. Third, the air typing system should only listen to the master's control movements and should not be deceived by others. I also used an EMNIST data file which is not a suitable air signature data file. Future object recognition algorithms such as YOLO v3 will improve the accuracy and speed of fingertip recognition. In the future, advances in artificial intelligence will improve the efficiency of air typing.

REFERENCES :

1. Y. Huang, X. Liu, X. Zhang, and L. Jin, Based On the Pointing Gesture Egocentric Enteraction System: Dataset, Approach, and Application in 2016 Computer Vision and Pattern Recognition Workshops (CVPRW), Las Vegas

2. X. Liu, Y. Huang, X. Zhang, and L. Jin. A cascading CNN pipeline for real-time fingertips Self-centered video recognition", Corr, 2015.

3. Yuan-Hsiang Chang, Chen-Ming Chang, "Automatic Hand-Pose Detection Tracking System Using Video Serise", INTECH, Croatia, 2010.

4. Omar Javed and Mubarak Shah, "Object Tracking: ACM Computer Survey. Vol. 38, Issue. 4, Article 13.

5. EshedOhn-Bar, Mohan ManubhaiTrivedi, "Hand Gesture Recognition In Real-Time For Automotive Interfaces," IEEE Transactions on Intelligent Transportation Systems, VOL. 15, NO. 6, December 2014, pp 2368-2377.

6. Yusuke Araga, Makoto Shirabayashi, Keishi Kaida, Hiroomi Hikawa, "Real Time Gesture Recognition System Using Posture Classifier and Jordan Recurrent Neural Network", IEEE World Congress on Computational Intelligence, Brisbane, Australia, 2012.

7. Lowlesh Nandkishor Yadav, “Predictive Acknowledgement using TRE System to reduce cost and Bandwidth” IJRECE VOL. 7 ISSUE 1 (JANUARY- MARCH 2019) pg no 275-278.

8. Saoji, S., Dua, N., Choudhary, A. K., & Phogat, B. (2021). Air canvas application using Opencv and numpy in python. IRJET, 8(08).