Virtual Drawing Using OpenCV

Mr. K. Sairam[1], Kavadi Sadana[2], Muduthanapally Shravani[3], Patnam Sreelekha[4]

[1]Assistant Professor, Department of Computer Science and Engineering Nalla Narasimha Reddy Education Society's Group of Institutions Hyderabad, India

[2,3,4]Scholar, Department of Computer Science and Engineering Nalla Narasimha Reddy Education Society's Group of Institutions Hyderabad, India

Abstract

We know that during the time covid19 pandemic online learning has become very important and this technique is widely used all over the world. But there are many limitations for Online learning. One of the most important limitation is concentrating on the line which is taught. It is also important that the Teachers to mark the important words or lines. Using Virtual drawing algorithms and controlling system algorithms which contributes greatly enhancement progression of which can improving the UI between the men and machine in the various applications. So, using OpenCV, Media pipe, and Python we can compute a solution to this problem but we know that implementing Virtual Air Drawing Where it access user for create drawings by just moving our fingers, and which makes our project decent.

Keywords: OpenCV, Media Pipe.

I. INTRODUCTION

OpenCV (Open Source Computer Vision) - is a programming language library consisting of different types of functions mainly for computer vision. To explain in a simple language or in general way it is a library used for Image Processing. It is used mainly to do all the operations which are related to Images.

What it can do:

1. Read and Write Images.
2. Detection of faces and its features.
3. 3. Detection of different shapes such as circle, rectangle etc. in an image. E.g. Detection of coins in images.
4. Text recognition in images. e.g. Reading Number Plates.
5. Can modify the quality of an image or it's color.
6. Developing Augmented reality apps. OpenCV is a library for images. It roughly supports all main programming languages. Commonly used in python and C++. OpenCV can be used to read or write an image, for image modification. Convert colored to gray, binary, HSV etc. OPENCV is also an OPEN SOURCE. As soon as computers can successfully think for themselves, in a way that has the plasticity that the brain does AI with quantum computing- i.e. not route algorithms - and that computer generates meaningful thoughts then sure. As most of the modern arts depend on concept, emotion, etc. there is an emphasis on communication. The question stands: can computers create something that looks cubist referring to the wide range of cubist works. Derive something out of the initial artwork although the

algorithm complexity to create unexpected emergence that would be hard to anticipate- they would still be reliant on initial input

II. LITERATURE SURVEY

Details about the literature review that we observed on the topic Virtual Air Sketching are viewed in this section. The papers' contents are listed below.

The Economical air writing system converting finger movements to text using web camera - The system is being developed using fingertip detection and finger movement techniques. Fingertip is first detected using Python, OpenCV, and CNN techniques, and then its trajectory is tracked and shown on the screen. The tracking of the hands and the tips of the fingers is done using the Media pipe package. The movements of the LED-fitted fingers are recorded using a web camera, and the patterns are recognized using characters from the database. As it is simpler to track down the red color, is attached to the user's finger to speed up finger movement tracking. The precise character is found and displayed on the screen using the optical character recognition (OCR) technique. This OCR method makes use of a pre-built database that is filled with the entire English alphabet, from A to Z. This database is used to identify the English alphabet and compare it to the cropped black-and white image. A text editor, such as Notepad, displays the English character. The user's characters are processed one at a time by this system, and the process is looped until the user has finished typing. Software called MATLAB is used to program all intended operations and produce the desired result. By dividing the problem solution into various modules, the aforementioned methodology is put into practice. The method for automatic video indexing and video search in large lecture video archives is presented in this research. They have used key frame detection and automatic video segmentation to provide a visual roadmap for navigating the video material. Video is converted into textual data using OCR (Optical Character Recognition) on different frames of video and ASR (Automatic Speech Recognition) on audio tracks. Automated Lecture video indexing: In order to extract each individual slide frame with its own temporal scope and be regarded a video segment, they first detect the slide transitions from the visual screen. Then textual metadata is extracted from slide frames using video OCR analysis. The first step in the implementation phase is to capture a video and then separate it into sequences. Red-colored objects are tracked from this series of images, which includes 100 images. In this instance, it is assumed that the only red colored objects in the environment are those illuminated by the tracking LED light mounted on the finger ModelNeural Network Model.

The air canvas application using OpenCV and NumPy in python: This database is used to identify the English alphabet and compare it to the cropped black-and white image. A text editor, such as Notepad, displays the English character. The user's characters are processed one at a time by this system, and the process is looped until the user has finished.

III. RESEARCH GAP

KEYBOARD

There are several methods to do this, one of which is to utilize the keyboard, a conventional and popular technique to show.

SPEECH-TO-TEXT

The second technique uses a programmer called speech-to-text that operates specific device along with audio. Through voice recognition, the programmer does this recognition.

TOUCH SCREEN

Another option is to use a touchscreen. You may interact with the computer screen using your rather of a mouse and keyboard.

IV. PROPOSED SYSTEM

• In this proposed project we are going to create a virtual painter using AI.
• The main objective is to, first track our hand and get its landmarks and then use the points to draw on the screen.
• Two fingers are used for selection and one finger for drawing. All this is done in real time.
• OpenCV library are used to track hand position in real-time to draw on the screen using the index finger.

V. METHODOLOGY

In this proposed project we are going to create a virtual painter using AI. The main objective is first to track our hand and get its landmarks and then use the points to draw on the screen. Two fingers (that is middle finger and the index finger) are used for selection and one finger (index finger) for drawing. All this is done in real time. Media pipe and OpenCV libraries are used to track hand position in real-time to draw on the screen using the index finger.

Features:
1. Able to draw by holding your index finger up.
2. Erase by holding both index and middle finger up.
3. Change colors by selecting the desired color using selection mode.

Approach:
Import libraries and modules necessary:
1. Media Pipe is a framework mainly used for building audio, video or any time series data. With the help of this we can build very impressive pipelines for different media processing functions. Some of its major applications are -Multihand tracking, face detection, object detection and tracking etc.
2. OpenCV is used to track an object of interest and allows the user to draw by moving the object, which makes it easy to draw simple things.
3. NumPy is a library for the Python programming language, adding support for large, multi- dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.
4. Time library is used to show framerate when we draw in real-time.

The steps involved are:
1. **Import header images -** The headers allow us to select the desired color using which we want to draw or select the eraser tool using which we can erase whatever is written on the screen. After importing headers, we have to run the webcam and overlay the images onto the webcam. Since the user is drawing in the opposite direction, the image has to be flipped horizontally which makes it easier to read and draw.
2. **Find hand landmarks -** The landmarks of hand which includes landmark points for tips of the fingers and has to be found. This is done using the Hand Tracking module. Detection confidence has to be given where high confidence is used to have good drawing experience and not commit any mistakes. The default confidence value is 0.5 but in this project the value is set to 0.85. The next step is to get

the landmark values and set the drawing mode to false as we do not want to draw here.

3. **Check which fingers are up -** We have to check which finger is up as we have to draw when one finger is up which is the index finger and select only when two fingers are up. This allows us to easily move around the canvas without painting. So, when two fingers are up, nothing is drawn on the screen and if we want to draw, one finger has to be used i.e., the index finger. This allows us to easily navigate through canvas. Here we check if the tip of thumb is on left or right hand and then for fingers we check if the tip of finger is above the other landmark which is two steps below it or not. If it is below, then the hand is closed else it is open.

4. **Selection mode -** Check if we are in selection mode. If yes i.e., two fingers are up, then we have to select the color using which we want to draw or select eraser if we want to erase something. Here, we set a rectangle as the visual indicator for selection mode. If the user is in selection mode check what the user is clicking on whether on brush or eraser. After selecting something from the header, change the draw color to the color selected and if eraser is selected use black color.

5. **Drawing mode -** Check if we are in drawing mode. If yes i.e., one finger is up(index finger), draw from starting point to ending point where the user moves his index finger. NumPy is used to draw over canvas on which the image will be saved. Here, we set a circle as the visual indicator for drawing mode.

VI. IMPLEMENTATION

1. COLOR TRACKING

By understanding the HSV Shading space for Color Tracking Further-more following, the little hued object at the fingertip. By approaching the pictures from the webcam is to be changed over to the HSV shading space for recognizing the hued object at the tip of finger.
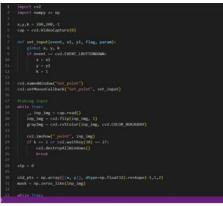
2. TRACKBARS

With the trackbars arrangement we will find the real time esteem from the trackbars and make range this reaches NUMPY structure which utilizes to be passed in the capacity This capacity returns the mask on the hued object. This mask is a high contrast picture with white pixels at the situation of the ideal tone.

3. CONTOUR DETECTION

By recognizing position of colored item as a fingertip by shaping a circling over it and we are playing out with MORPHOLOGICAL procedure on the Mask, to make it liberated from contaminations and to distinguish shape without any problem. This is Contour Detection.

4. FRAME PROCESSING

Following the fingertip and drawing focuses at each position for air material impact That is Frame Processing. Frame processing plays an important role because the higher the frame processing higher will be the accuracy.

```
1   import cv2
2   import numpy as np
3
4   x,y,k = 200,200,-1
5   cap = cv2.VideoCapture(0)
6
7   def set_input(event, x1, y1, flag, param):
8       global x, y, k
9       if event == cv2.EVENT_LBUTTONDOWN:
10          x = x1
11          y = y1
12          k = 1
13
14  cv2.namedWindow("Set_point")
15  cv2.setMouseCallback("Set_point", set_input)
16
17  #taking input
18  while True:
19      _, inp_img = cap.read()
20      inp_img = cv2.flip(inp_img, 1)
21      grayImg = cv2.cvtColor(inp_img, cv2.COLOR_BGR2GRAY)
22
23      cv2.imshow("_point", inp_img)
24      if k == 1 or cv2.waitKey(30) == 27:
25          cv2.destroyAllWindows()
26          break
27
28  stp = 0
29
30  old_pts = np.array([[x, y]], dtype=np.float32).reshape(-1,1,2)
31  mask = np.zeros_like(inp_img)
32
33      while True:
```

VII. RESULTS

VII. CONCLUSION

As we develop this system, we offer a very straightforward and cost-effective solution for one of the burgeoning areas of translating finger movements to text. This system is designed in such a way that it can recognize the English alphabet drawn in the air and translate it into text using just a basic Webcam. Any computer that has a Web Camera and the necessary configurations can have the system installed. This system can undoubtedly be seen as the starting point for the more recent invention of translating finger movement to text and can act as a benchmark for future improvements in the same field.

IX. REFERENCES

1. Prof. S.U. Saoji, Nishtha, Akash Kumar Choudhary, Bharat Phogat, "Air canvas application using OpenCV and NumPy in python" in IRJET, (Deemed to be University) College of Engineering, Pune, Volume: 08 Issue: 08, e-ISSN: 2395-0056, p-ISSN: 2395-0072, Aug 2021

2. Pavithra Ramasamy, Prabhu. G, Dr. R. Srinivasan, "An economical Air Writing System Converting finger movements To Text Using web Camera" in Fifth International Conference on Recent trends in Information Technology, 978-1-4673- 9802-2, 2016.

3. Sumedhi Bansode, Saurabhi Varkhad, Shivam Waghmare, Prof. Shubhangi Suryawanshi, "Computer Vision Based on Virtual Sketch Using Detection" in IJRASET, DR D Y Patil institute of Technology, Pimpri, Pune, Volume 10 Issue V, ISSN: 2321-9653, IC Value: 45.98, SJ Impact Factor: 7.538, May 2022.

4. AsstProf.Jahnavi Naik S, K Sai Sumanthi Reddy, Abhishek Reddy, AbhinandanHegde, Lakshmi Prashanthi Reddy, "Virtual Air Canvas App using OpenCV and Mediapipe in Python" in IJARIIE, Dayananda Sagar Arcade.