IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

VIRTUAL PEN USING COMPUTER VISION

[1]Mr. Shubham Deolekar, [2]Miss. Eshani Korgaonkar, [3]Miss. Amisha Madav

[1]Designation of 1st Author, [2]Designation of 2nd Author, [3]Designation of 3rd Author

[1]Information Technology,

[1]Finolex Academy of Management and Technology, Ratnagiri, India.

***Abstract*:** Virtual Pen System is a software designed to detect and capture the text or shape drawn in the air. The shape can be drawn using a normal pointer of blue colour. This application will help the artists to draw free-hand drawing and can be helpful for them around the world to draw some sketches or create their art work virtually and easily. It can also be used to draw text on canvas. The user can experience the joy of writing or drawing a 3D free space with an ease as the application is user friendly.

Index Terms - *python*, OpenCV, computer vision, marker.

I. INTRODUCTION

Writing or drawing something in the air is the process to write or draw something in a 3D space. Virtual pen using computer vision is an active research area which attempts to develop a software with the ability to automatically read the text or any symbol which is drawn in the air. This system refers to writing or drawing something in air by moving a hand with the help of a normal pen. It is widely applicable where traditional pen-up and pen-down writing systems are troublesome. Due to the simple writing style, it has a great advantage over the gesture-based system in today's world.

In this system, we have developed an air-drawing recognition system by the use of a normal pointer, the camera that tracks the brightest pixels of blue colour in the frame and displays the captured path by brightest pixels and then displays the input on canvas as well as a live capturing screen. Traditionally it becomes very difficult to write or draw something through a mouse on a desktop and seems to be very challenging for the user to give the information directly to the computer. Thus, there is a need for a Virtual text Recognition mechanism to analyse any sign or symbol which is drawn in the air and to save the input given by the user. It's user-friendly GUI at any time enables the user to access the services provided by the system with ease.

II. LITERATURE REVIEW

Trajectory based writing system [1] refers to writing a linguistic character or word in free space by moving a finger, marker or handheld device. It is widely applicable where pen-up and pen-down writing systems are troublesome. However, it is a challenging task because of the non-uniform characters and different writing systems. It also employed a 6D edition gesture alphanumeric character dataset and achieved 99.32% accuracy which is the highest to date.

Hand gesture recognition using PCA [2] dealt with the detection and recognition of hand gestures. Images of the hand gestures were taken using a Nokia N900 cell phone and matched with the images in the database and the best match was returned. It could help in video gaming by allowing players to interact with the game using gestures instead of a controller. However, such an algorithm needs to be more robust to account for the myriad of possible hand positions in three-dimensional space.

GFRECOG [3] A Generic Framework with Significant Feature Selection Approach for Face Recognition included two kinds of methods that are currently popular in developed face recognition patterns namely Eigenface method and Fisher face method. Facial image recognition Eigenface method is based on the reduction of face dimensional space using Principal Component Analysis (PCA) for facial features. The main purpose of the use of PCA on face recognition using Eigenface was formed by finding the eigenvector corresponding to the largest eigenvector of the face image. The area of the project face detection system with face recognition was image processing. The software requirement used by them was MATLAB software.

In Object Detection and Identification [4] the main aim was to detect all the instances of objects from a known class, such as people, cars or faces in an image. The image it detects is then reported with some pose information. This is as simple as the location of the object, location and scale, or the object defined in terms of a bounding box. It's methodology included Squeeze Net, InceptionV3, Dense Net, Squeeze NAS.

III. PROPOSED SYSTEM

We proposed a system using python programming language which has inbuilt live capturing video libraries like OpenCV, to make a system simple. For this system we are using vector methodology. We are using a jupyter notebook for compiling and executing source code of the system. The user has to first run the code to use this system. After executing the code this system will start and four windows will be displayed: Color detectors, Paint, Tracking, Mask. Using the "Color detectors" window we can set the color of the pointer we want to track. On the "paint" window we will see the drawing we have drawn. On the "tracking" window we will see live camera feedback and color selection options, clear screen option. We will draw in front of this window and the drawing will be visible on this window also. On the "mask" window we will see the brightest pixels of blue color. Then whatever gestures the user wants to draw in the air should be drawn with the help of a pen. The camera captures the text or image drawn in the air and tracks the brightest pixels in frame with the help of vector methodology. Then the brightest pixel path is displayed. Our system is a desktop system and can be used for various purposes like free hand drawing, digital signature, etc.

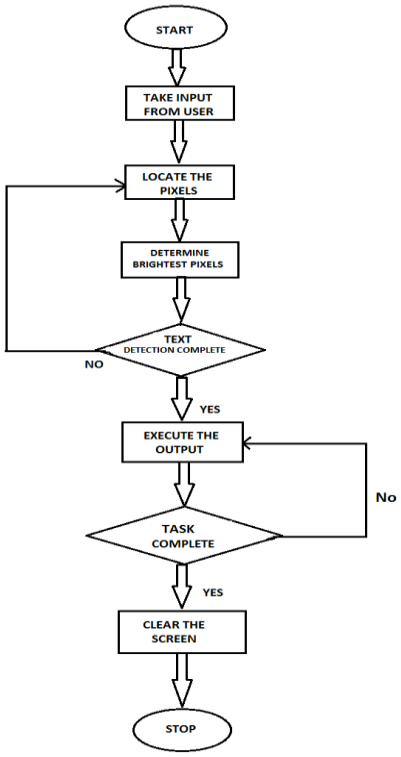

Fig 1. Block Diagram of the system

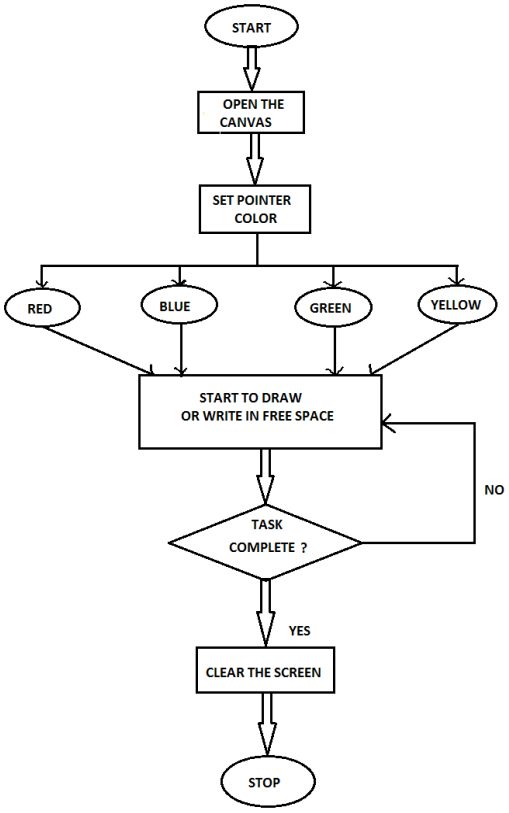

Fig 2. System startup flowchart

Eventually, the system provides various services to the user which can be used for different purposes : Easy drawing while teaching [1] Considering the current situation of global pandemic and due to lockdown online education is preferred everywhere, many times, teachers need to draw some figures or write something in order to explain a particular concept. Virtual pen can be used for teaching purposes by eradicating the limitation of the board, by providing a sufficient amount of space to draw. Free Hand Drawing for Artists [2] Artists around the world can use this system for creating their artwork with a joy of drawing virtually. [3]Making digital signatures, As the normal process of making digital signatures may seem to be somewhat hectic, Virtual pen proves to be an advantage in this process.

IV. EXPERIMENTATION

The basic system setup requirements include a well-functioned laptop/desktop with a web camera and a hand-held marker/pent. Laptop/desktop should have an i3 processor or higher with minimum 4gb RAM and minimum 2gb storage space. We are using an integrated laptop web camera with 1920*1080 resolution for the initial setup. Web cameras in this process will be used by the system for tracking the movements of the marker resulting in it to get the coordinates of the most visible brightest pixels. After the successful testing of the web camera, the marker will be set to work as a pointer or a pen which the user can use to draw or write in a free space. We are using python programming language and libraries like OpenCV, NumPy and deque data structure.

'Virtual Text Recognition using Computer vision' will work on vector methodology. At the initial stage, the web camera used in the capturing process will track the most visible brightest pixels. The frame rate will be approximately 30 fps, i.e., the system will be capturing 30 pairs of coordinates in one second. The system will start to map the coordinates of these brightest pixels, which will then be given to the line function accordingly. As the line function receives the coordinates the process will be executed and the line will be drawn. The main function of the system is to execute the text written or the image drawn in the free space by the user on the screen. Since, the design of many systems dealt with the recognition of hand gestures and allowed writing in a free space which eventually eradicated the flaws in traditional writing. Virtual Text Recognition using Computer vision extends the scope, where along with the process of writing in a 3D space user are also given an access to a variety of system services like getting the pdf copy of the text written, free hand drawing, image cropping and making digital signatures. This enables the system to be used in various fields like teaching, photography etc.

We have made use of NumPy, cv2 libraries and dequeue data structure for the project. Track bars are created for adjusting the marker colours set by values i.e., optimal and max value. We have created a total of 6 track bars: Upper hue, Upper saturation, Upper value, Lower hue, Lower saturation, Lower value. Using these trackbars we can set the colour of the brightest pixel to be tracked, in this system we are tracking blue pixels so we have set the track bar values using setValues() function accordingly. The values are 160, 255, 255, 64, 72, 49 respectively. Different arrays are used to handle the colour points of different colours using dequeue, the max length of this deque structure is 1024; assigning the index values set to zero. Kernel is defined at the initial stage for masking. After setting the colour values of the colours i.e., Red, Blue, Green, Yellow, the colour index value should be set to zero.

After executing the code 4 windows will be displayed: Colour detectors - for trackbars, Paint - for white canvas, Tracking - for live camera feedback, Mask - for detecting the brightest pixel. For the painting window setup, the value as for the white canvas is set up as up to 255. We use cv2.rectangle() function for adding the boxes named Clear, Red, Blue, Green and Yellow all-in sequence. We have added names to these rectangles using cv2.puttext() function. We will capture live camera feedback using cv2.capture(0) function. As we are using an integrated web camera we have passed '0' to the capture function. Now turning to the live capture phase, the while loop is set for whether the live camera feedback is available or not, then the frame will be flipped. HSV is a very convenient way to work with images so we are converting BGR values to HSV using cv2.COLOR_BGR2HSV () function. The position of the trackbar is set including the upper and lower HSV values for identifying the marker being present between the range of the mentioned values. Frames are used in live video clips including the four-color rectangles.

In order to get a clear image of the marker we need to remove the impurities in the environment that disturbs the accuracy of the colour which can be done by Masking by performing the operations like erode, morphology and dilate by the range being set. ref. fig 3.4. As we are tracking blue pixels so in the Mask window the position of only blue pixels is visible and the other surrounding is blacked. In the stage, of finding the contours i.e., continuous line segment the centre is assigned to None. If the contours are formed the length is set to one. During the sorting cycle of the contours the biggest contour among all will be considered as the pointer and enclosed with a circle. The centre of the detected contour is then detected and checked whether any of the buttons above the screen is clicked or the cursor is hovered to. Following the condition, if the centre comes to be less than 65 the cursor detects the area as a rectangle either clear or any of the colour rectangles. If the colour index points greater than 65, we are still in the painting section of the canvas. Setting up the points conditional values a line is drawn. Lastly the program will be terminated releasing the camera and all the resources for a better efficiency. We can press "Q" on the keyboard to close the system.

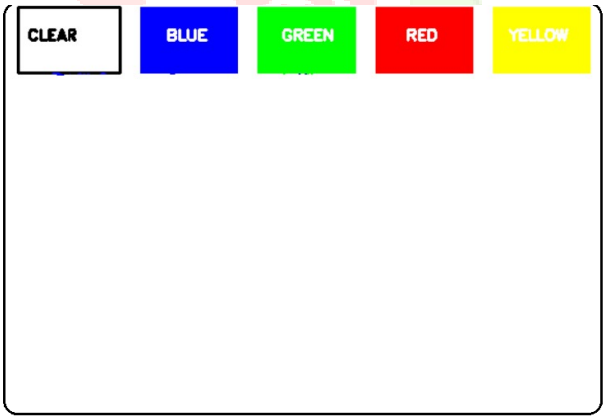

Fig 3.1. Painting Canvas window

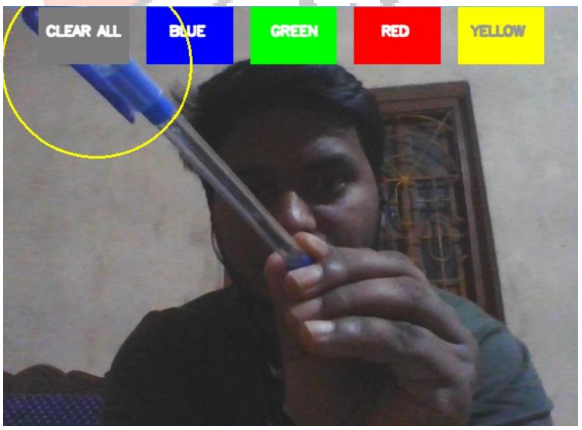

Fig 3.2. Live video stream (before drawing)

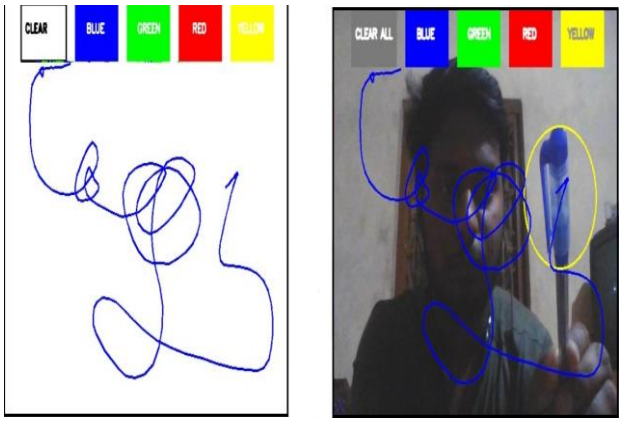

Fig 3.3. Live video stream and canvas (after drawing)

Fig 3.4. Mask of a pointer

V. CONCLUSION

The Virtual Text Recognition Using Computer Vision System is designed to detect the text or image drawn in air. Users have to use an LED to write or draw something in the air. As our system is not an android system, desktop or laptop is required. This system can be used in various fields like education, image cropping, etc. Our system is completely user friendly and components are at an affordable price so anyone can use this system easily. The accuracy of the colour detection unit can be affected if the environment contains the same colour as that of the led pointer.

REFERENCES

[1] "Trajectory-Based Air writing recognition using Deep Neural network and Depth sensor" Md.ShahinurAlam, Ki-Chul Kwon, Md.Ashraful Alam,Md.Y.Abass ,Shariar Md Imtiaz and Nam Kim, Department of Computer Science and Engineering, BRAC University Dhaka, Bangladesh, India

[2] "Hand Gesture Recognition using principal component analysis- Sunil Kumar,Tanu Srivastava and Raj shree singh-Department of Computer science and Engineering Mody University of Science & Technology,Lakshmangarh,Rajasthan,India

[3] GFRecog: A Generic Framework with significant Feature selection Approach for Face Recognition- Riyadh Rahef Nuiaa, Bahaa Kareem Mohammed,Seif Ali- International Journal of Engineering & Technology

[4] Object Detection and Identification-Mukkamala Rohith Sri Sai, Sainagesh Veeravalli , KL University