International Journal of Scientific Research & Engineering Trends
Volume 9, Issue 3, May-Jun-2023, ISSN (Online): 2395-566X

Gesture Based Drawing - Gesdraw

Asst. Prof Ms. Geeta N. Brijwani, Mr. Pranav Patel, Mr. Moinuddin Mansoori
Department of Computer Science
KC College,Mumbai, India
geeta.brijwani@kccollege.edu.in, pranavp778@gmail.com, moinmansuri1998@gmail.com

Abstract- Using a trackpad or a pen tab might be restricting the artistic flow of people. The system suggested in this research seeks to address this issue. The solution is an application to track the user gestures and relay the drawing as such using only a web camera for the detection. There is a MediaPipe model that has been utilized with CV2, to allow real-time gesture detection and capture and thus it allows free flow of creativity.

Keywords- Real-time, OpenCV, NumPy, MediaPipe, Python, CustomTkinter, Docx, Speech.

I. INTRODUCTION

There are several existing systems that allow users to draw on the screen or on a canvas on the computer using a trackpad. Such solutions seem very tedious as it is very uncomfortable to draw using a trackpad. Another solution is to draw using a pen and tablet. But such a solution involves investment from the user itself.

Gesdraw is a digital drawing canvas which utilizes camera, opencv and MediaPipe libraries to recognize and map the hand gestures. The user's finger is considered as the brush or the pen used to draw on the interface chosen by the user. The size of the brush can be modified, pen color can be changed by motion over built-in buttons, and even more functionalities can be involved. But how would one actually capture the hand movements? The answer would be employing the computer's camera to directly capture the movements, process them and apply it on the canvas. Hand tracking, more specifically finger tracking, technique is used as a tool with the computer acting as an external device similar to a keyboard and a mouse. It is used in various fields like Virtual Reality to sign language recognition and more.

The project combines computer vision techniques to figure out the user's hand movements, and plot them on whichever medium the user might be inclined to, all to create a unique and interactive user experience. The application can be used for a variety of purposes, such as creating digital art, brainstorming, or teaching and much more. The project involves building a program that can accurately detect and recognize position of the index finger, which serves as the guide to drawing. The same points have to be used to display colored points which make up the user's drawing.

The application has to have a user-friendly interface that allows users to easily navigate and use the drawing tools and reduce complexity otherwise the objective of the project might end up being missed. Additional features such as color selection, brush size selection, and eraser functionality have to be involved to enhance the user experience and provide more options and functionality to use. Gesdraw is an exciting opportunity to explore the possibilities of computer vision and graphics while creating a useful and engaging application for users.

II. LITERATURE SURVEY

There have been several advancements made in the field for gesture recognition which involved the development of the MediaPipe library by Google for the same. It has several algorithms written in Python, JavaScript and more supported languages. The algorithms include ones for object detection, image classification, and gesture recognition with pose tracking. There has been a custom library developed by Mr. Tom Schimansky in an effort to create a more efficient and user-friendly UI in python. It is the CustomTkinter library in Python, available on GitHub.

III. PROPOSED MODEL

The answer to the problem statement is rather easy to develop considering that several iterations have been developed. The Agile methodology will be implemented, specifically, the scrum framework wherein sprints will be utilizing short periods of development centered on a goal. The author has used the MediaPipe library, which contains the hand detector class, configured with respect to the tracking confidence and number of hands to track. The same class would also contain several functions that return the number of hands detected, number of fingers detected, and the finger landmarks or positions.

IV. METHODOLOGY

Since the algorithm for development is ready, the first task would be to develop specific implementations of the functions in the HandDetector class of the MediaPipe library for our problem statement. Fig 1 lists landmarks of all the points in the hand.

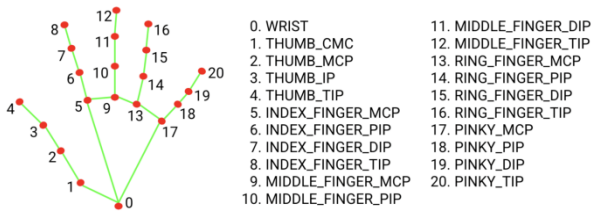

Fig 1. Hand Landmarks (Figure shows every landmark tracked by the algorithm along with their indices).

Functions would be developed to differentiate the number of hands, number of fingers that are up as well as the landmarks of each hand detected to be converted into pixel values for drawing.

Once the core is developed, the Main GUI would be developed that allows users to select different canvas to draw on. Next the individual options are developed where each one has common functionalities such as 3 color selection, eraser, voice assistant, saving options and clear option. The only difference among the options would be the medium selected by the user to draw on which can be one of the 3 - Blank canvas, drawing on the screen, and drawing on an existing image.

The last tasks would be to develop additional functionalities for Voice assistant, document saving feature and image saving features.
Also, creating folders to save the documents, for saving the images of user's work, and for saving the images used in the GUI can be beneficial for organizational purposes.

V. TECHNOLOGY ADOPTED

Python is a high-level, interpreted programming language that is based on OOPs (Object Oriented Programming). It allows for simpler code development along with capability extension via the large collection of libraries that have been created by its large community.

It has been increasingly used in Machine Learning, Artificial Intelligence, Data mining and other concepts. The entire solution would be created in Python. Various modules have been employed for the following uses:

1. CV2 :
Utilized to capture the user movements and display the canvas which will reflect the gestures. It will also be used to save the drawings in the image format. Frames captured by the webcam are converted into NumPy arrays. Also assists in addition of header images to the application.

2. Mediapipe:
It is the module that provides with multiple algorithms for hand tracking, body movement tracking, face mesh, face detection and more. It provides the algorithm implemented for the application to allow efficient tracking of hand movements.

3. Time:
Used to calculate the frame per second to estimate efficiency of the application. Also utilized for saving document with timestamp and keeping track of user session.

4. OS:
Allows one to access the file system to retrieve images from the folder as well as to save the images and documents.

5. NumPy:
To save the frames in the form of NumPy arrays for easy manipulation. To create the canvas and plot points and strokes on the same.

6. PYTTSX3:
Allows initializing the voice engine to reply to the user for various commands. Utilized in the voice assistant of the application.

7. Speech_Recognition:
Utilized in the voice assistant. Allows conversion of speech to text and also allows making up for the noise in environment and provides several important functions.

8. Datetime:
Used to create the timestamp for saving the document.

9. Random:
For random values to be utilized.

10. Docx:
To create a word document for each session and similarly add screenshots of the drawing in the same upon function call. Also allows formatting of the word document with the addition of headers and more.

11. Tkinter:
To design the GUI of the system and provide options for the same. The main screen of the application is created with the Tkinter module.

12. CustomTkinter:
Utilized on top of Tkinter in the main screen. To provide enhancement over the look and feel of the Tkinter application, with its sophisticated widgets.

13. Pillow:
To assist in the creation of the GUI. Allows addition of images in the image buttons.

VI. FLOW DIAGRAM

The entire application would work as follows:

- User starts the application which displays the main screen allowing for either of 3 options to be selected.
- Upon subsequent selection, the user can draw on the screen as required and can select several options such as changing color, eraser, clearing screen, etc.

- User can save the selection in a document or in an image format, and can hence choose to exit the application or continue further.

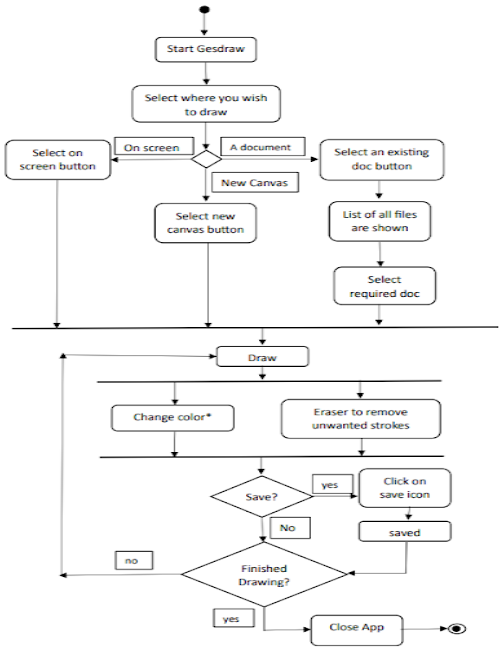

Fig 2. Flow Diagram of the entire application (Indicating the sequence of events that the user can follow).

VI. CONCLUSION

The aim of this project was easy conversion of thought to sketches for purposes such as art, or teaching, by the means of gesture recognition. As the developed system only utilizes a webcam in terms of hardware, it is also an economical solution. It has allowed easy drawing without the use of a mouse as such.

But there are also various shortcomings of the system, a few of them are:

- Accuracy of hand tracking.
- Capture and processing speed of images appears to be slow.
- Noise and other factors affecting the voice assistant.

The Gesdraw project has the potential for further innovations and solution, some of them are:

- Adding more features such as different brush sizes and shapes and a color wheel
- Improving the accuracy of hand movement detection and the camera frames per second
- Adding the ability to share the artwork on social media platforms.
- Adding the ability to draw on multiple canvases.
- Collaborative editing of the canvas amongst multiple users.
- Application of the technology in 3D modelling and animation.

ACKNOWLEDGMENT

We would like to Ms. Geeta Brijwani for providing her constant support and guidance and inputs in the making of the project. Also, we would also like to take the opportunity to thank Mr. Aditya Agarwal and Ms. Kirti Bhatt for providing their suggestions and advice for the making of this project. Lastly, we would like to thank the college for giving us the opportunity to do research and publish a paper in a good impact research journal.

REFERENCES

.

[1] P. Ramasamy, G. Prabhu, and R. Srinivasan, "An economical air writing system is converting finger movements to text using a web camera," 2016 International Conference on Recent Trends in Information Technology (ICRTIT), Chennai, pp. 1-6, 2016. (9, Regular).

[2] Saira Beg, M. Fahad Khan and Faisal Baig, "Text Writing in Air," Journal of Information Display Volume 14, Issue 4, 2013.

[3] Alper Yilmaz, Omar Javed, Mubarak Shah, "Object Tracking: A Survey", ACM Computer Survey. Vol. 38, Issue. 4, Article 13, Pp. 1-45, 2006.

[4] P. Ramasamy, G. Prabhu, and R. Srinivasan, "An economical air writing system is converting finger movements to text using a web camera," 2016 International Conference on Recent Trends in Information Technology (ICRTIT), Chennai, 2016, pp. 1- 6.

[5] S. Vikram, L. Li, and S. Russell, "Handwriting and gestures in the air, recognizing on the fly," in Proceedings of the CHI, vol. 13, 2013, pp. 1179–1184.

[6] Vladimir I. Pavlovic, Rajeev Sharma, and Thomas S. Huang, "Visual Interpretation of Hand Gestures for Human-Computer Interaction: A Review," IEEE Transactions on Pattern Analysis and Machine Intelligence, VOL. 19, NO. 7, JULY 1997, pp.677-695

[7] W. Makela, "Working 3D Meshes and Particles with Finger Tips, towards an Immersive Artists' Interface," Proc. IEEE Virtual Reality Workshop, pp. 77-80, 2005.

[8] 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

[9] Ruiduo Yang, Sudeep Sarkar, "Coupled grouping and matching for sign and gesture recognition", Computer Vision and Image Understanding, Elsevier, 2008.