IPEC Journal of Science & Technology, Vol. 02 (01), June 2023
ISSN: 2583-3286(Online)

Digitized Interaction: A Gesture-Controlled Whiteboard System with OpenCV, MediaPipe and NumPy

Swapna Singh, Hiya Shukla, Keshav Sharma, Harsh Tyagi, Jaya Prasad
Department of Computer Science and Engineering,
Inderprastha Engineering College, Ghaziabad, Uttar Pradesh, India

© The Author(s), under exclusive license to publication division, IPEC Journal of Science & Technology, 2023

Abstract: **This research paper introduces an innovative approach to developing a gesture-controlled whiteboard system using the open-source libraries OpenCV, MediaPipe, and NumPy. The proposed system leverages computer vision techniques to accurately track and interpret finger movements captured by a camera, enabling users to write on a virtual whiteboard without the need for physical markers. This paper details the technical implementation of the system, including the integration of OpenCV for real-time hand detection, MediaPipe for hand landmark estimation, and NumPy for efficient data processing. Writing is a mode of communication that enables us to articulate our ideas and convey information in a tangible form. Today, typing and writing are the predominant means of documenting information. Writing involves the formation of letters or words using a pen, pencil or even a finger, on a surface such as paper or a touch screen. In recent times, wearable devices have emerged that are capable of detecting and interpreting our actions through gesture recognition, which is a computing process that utilizes mathematical algorithms to comprehend human gestures.**
To monitor the movement of the fingers, computer vision is often employed. In addition to recording information, this technology can also be used to perform various tasks such as sending emails or text messages. Moreover, it has the potential to be an immensely useful tool for the hearing-impaired community, as it provides an alternative method of communication that does not rely on sound. With the aid of this technology, individuals who are deaf or hard of hearing can effectively communicate with others, thereby enhancing their overall quality of life.

Keywords **- Gesture recognition, Air writing, wearable device, Mediapipe, numpy, OpenCv, computer vision.**

.

I. INTRODUCTION

Throughout history, the way humans write has undergone significant transformations. Writing was first invented around 2000 BC by neolithic people who began by etching on walls

before switching to stones. Subsequently, cloth replaced stones before paper became the dominant medium for communication. With the advent of QWERTY keyboards, there has been a shift towards digital writing. Electronic devices are becoming increasingly popular, and traditional writing implements such as pen and paper are slowly being phased out.

The need to develop human-machine interactions has grown rapidly in response to the widespread use of augmented and virtual reality. Applications that use hand gestures have gained popularity over the years. Several systems for hand gesture recognition have been developed, including Automotive interfaces, the Economical Air Writing system, and Handwriting recognition in Free Space. However, handwriting in air requires more than just gesture recognition. It involves detecting, tracking, and tracing fingertip movements. Pavithra and Prabhu developed a fingertip detection system using LED lights, which captures movement and displays the alphabet on a screen using Optical Character Recognition (OCR). However, such methods that require the use of devices have some limitations.

To eliminate the need for cell phones for note-taking, a proposed system that employs fingertip detection and finger movement techniques has been developed. The system uses Python, OpenCV, and Convolutional Neural Network (CNN) techniques. The fingertip is first detected, and then its trajectory is traced and displayed on the screen. The MediaPipe package is used for hand tracking, while Pynput, Autopy, and PyAutoGUI packages are used for moving around the computer screen, performing various functions such as left-clicking,

Date of Submission: 8 May 2023
Date of Acceptance: 20 June 2023
Corresponding Author: Keshav
(Email: 1900300100111@ipec.org.in**)**

ISSN: 2583-3286(Online)

right-clicking, and scrolling. The proposed AI virtual mouse model has shown a high level of accuracy and can function well in real-world applications using only a CPU, without the need for a GPU.

II. PROBLEM STATEMENT

One of the major challenges with the existing system is that it only works with fingers and cannot differentiate between other objects such as highlighters, paints, or relatives. Identifying and distinguishing a finger from an RGB image without a depth sensor is a difficult task. Furthermore, the system has difficulty detecting the up and down movements of the pen as it only uses one RGB camera. As a result, the finger path is abstract and not easily recognized by the model. Real-time hand touch is required to change positions, which requires careful coding and the user must be familiar with various movements to control the system effectively.

The project aims to solve some important social issues. One such issue is the communication barrier faced by hearing-impaired individuals in their everyday lives. While listening and speaking are taken for granted by many, people with disabilities who rely on sign language struggle to communicate with those who do not understand it. This system will serve as a tool for communication for the deaf and will allow their online text to be displayed in augmented reality.

Another issue is the overuse of smartphones, which can lead to accidents, stress, distractions, and other illnesses. While smartphones are portable and easy to use, their excessive use can lead to terrifying events in life. Additionally, waste paper is a common problem. A4 paper production requires a significant amount of water, and 93% of paper sources come from trees. Furthermore, 50% of commercial waste is paper, and 25% of landfills are composed of paper waste. This waste harms the environment through water and tree use, and it produces tons of waste. On-air writing can quickly solve these problems as it eliminates the need for physical paper and allows for more efficient communication

III.. LITERATURE SURVEY

Real-time hand gesture recognition is a crucial area of research that has led to the development of various techniques for different applications. One such system developed by Shomi Khan, M. Elieas Ali, and Sree Sourav Das uses a skin color identification algorithm to translate American Sign Language (ASL) from real-time video into text. However, identifying the hand can be a challenge as skin tone and hand form can vary from person to person. To overcome this, the system employs two neural networks. The Scalable Color Descriptor (SCD) neural network is the first algorithm used to identify skin pixels in the image, and the Hand Gesture Recognition (HGR) neural network extracts the features. The features are extracted by two distinct algorithms: Finding the fingertip and Pixel segmentation algorithm.

In addition to ASL translation, some systems can accomplish mouse actions, such as moving the cursor, clicking left and right with hand gestures, using computer-vision-based real-time dynamic hand gestures. S. Belgamwar and S. Agrawal have developed a new human-computer interaction (HCI) technique that integrates a camera, an accelerometer, a pair of Arduino microcontrollers, and an Ultrasonic Distance Sensor to capture motions. The distance between the hand and the distance sensor is determined to record the gestures.
Another innovative technology is the LED-based hand gesture recognition system developed by Pavitra Ramasamy and Prabhu G. It enables users to create the alphabet or type anything they wish by simply waving their finger over an LED light source. The system tracks the color of the LED to extract the movement of the finger and sketch the alphabet. The background is black, and the tracked object's color is converted to white. The user can draw an image of the alphabet in black and white by stitching together several black and white frames.

3D hand gesture detection is another area of research that has led to the development of various techniques. Quentin De Smedt, Hazem Wannous, and Jean-Philippe Vandeborre used a skeleton-based model to gain an effective descriptor from the Intel RealSense depth camera's hand skeleton linked joints. The skeleton-based approach is better than the depth-based approach. Prajakta Vidhate, Revati Khadse, and Saina Rasal developed a virtual paint application that uses ball-tracking technology to track the hand movement and write on the screen. They used a glove with a ping-pong ball attached to it as a contour.

Finally, Ruimin Lyu, Yuefeng Ze, Wei Chen, and Fei Chen developed an airbrush model that employs the Leap Motion Controller to track hands and produce an immersive freehand painting experience. These innovative technologies have great potential for a wide range of applications and can significantly improve human-computer interaction.

IV. ALGORITHM USED FOR HAND TRACKING

The MediaPipe framework handles hand gesture identification and tracking, and the OpenCV library handles computer vision. The application uses machine learning principles to track and identify hand movements and hand tips.

I. Mediapipe

In order to detect initial hand locations in real-time on mobile devices, we developed a single-shot detector model that is similar to the face detection model used in MediaPipe Face Mesh. However, detecting hands is a complex task that presents unique challenges. Our model must be able to work across a variety of hand sizes with a large scale span (~20x) relative to the image frame and detect occluded and self-

ISSN: 2583-3286(Online)

occluded hands. Since hands lack high contrast patterns, detecting them reliably based on visual features alone is difficult. Therefore, providing additional context, such as arm, body, or person features, can aid in accurate hand localization.

To address these challenges, we employ different strategies. Firstly, we train a palm detector instead of a hand detector, as estimating bounding boxes of rigid objects like palms and fists is significantly simpler than detecting hands with articulated fingers. Palms are also smaller objects, making the non-maximum suppression algorithm effective even for two-hand self-occlusion cases, like handshakes. Additionally, modeling palms using square bounding boxes reduces the number of anchors by a factor of 3-5. Secondly, we use an encoder-decoder feature extractor to provide bigger scene context awareness even for small objects, similar to the RetinaNet approach. Lastly, we minimize the focal loss during training to support a large number of anchors resulting from the high scale variance. These strategies enhance the accuracy and reliability of our hand perception model, enabling it to detect hands in real-time on mobile devices.

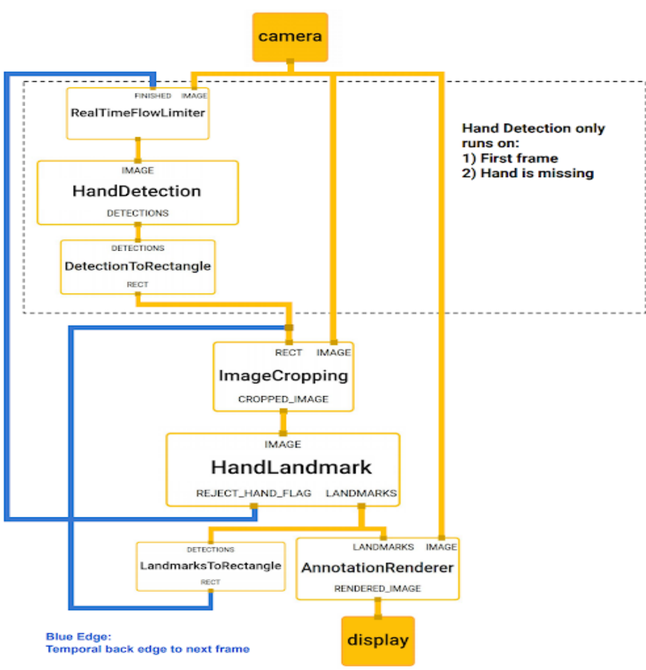

Figure 1. Hand Detection Model

II. Hand Landmark Model

To achieve precise keypoint localization of 21 3D hand-knuckle coordinates within the detected hand regions, our hand landmark model employs regression, or direct coordinate prediction, after detecting the palm over the entire image. This approach allows the model to obtain a reliable internal representation of hand posture, which is unaffected by self-occlusions or partially visible hands. To obtain ground truth data, we manually added 21 3D coordinates to approximately 30,000 real-world photos, as shown in the figure below. For corresponding coordinates, we extracted Z-values from the image depth map, if available. To better cover the range of potential hand poses and provide additional supervision on the nature of hand geometry, we also generated a high-quality synthetic hand model on a variety of backgrounds and mapped it to the associated 3D coordinates. By combining real-world and synthetic data, we were able to enhance the accuracy and reliability of our hand perception model.

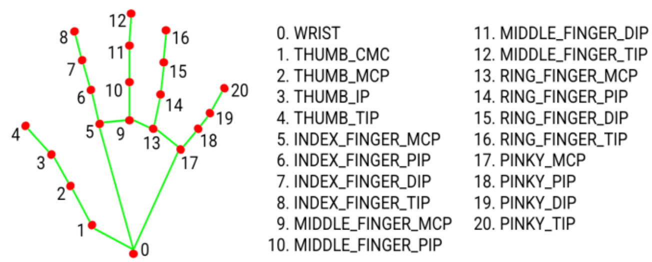

Figure 2. Hand Landmark Model

III. OpenCv

The OpenCV computer vision library includes image processing methods for object detection [14]. With the OpenCV library for the Python programming language, developers can create real-time computer vision applications. OpenCV library enables processing of images and videos, along with analytical techniques such as face and object detection. This versatile library provides a powerful tool for developing computer vision applications.

IV. Numpy

NumPy's most essential object is the N-dimensional array type known as ndarray. This object represents a collection of identically categorized elements, which can be accessed using a zero-based index. The items in an ndarray occupy the same amount of space in memory. Each item in an ndarray is represented by a data-type object called dtype. When retrieving an item from an ndarray object, it is represented by a Python object of one of the array scalar types. Thus, ndarray is a fundamental building block for many scientific computing and data analysis applications.

ISSN: 2583-3286(Online)

V. METHODOLOGY

Based on the web camera frames that were captured, a virtual paint programme was offered. The web camera sends the system the frames that it has received. Until the application is finished, the approach uses a web camera to collect each frame

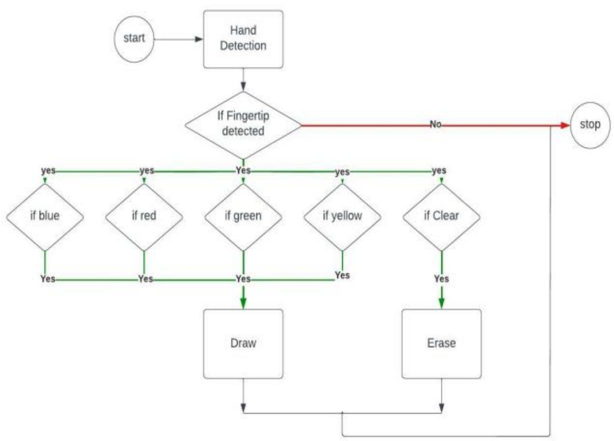

Figure 3. Flow chart of Mark-Air application

I. The Camera Used in the MARK AIR

The Mark Air system relies on the frames captured by the webcam of a laptop or PC. To initiate the recording process, a video capture object is created using the OpenCV computer vision package, as depicted in Figure 4. The camera begins recording video, and the frames are transmitted to the virtual AI system for further processing. The system utilizes the received frames to generate a digital canvas, where the user can interact and draw using hand gestures. By leveraging the OpenCV package and the webcam, the system can accurately capture and process the visual data in real-time, providing an immersive and responsive user experience.

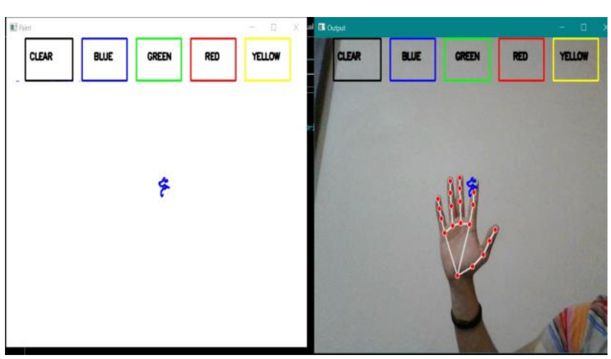

Figure 4. Capturing video using the webcam (computer vision).

II. Acquiring and Analyzing Video Data

The Mark Air system utilizes the webcam and captures every frame of the video until the application is terminated. The code accompanying the system converts the color format of the video frames from BGR to RGB to enable the detection of hands in each frame of the video. This process is carried out on a frame-by-frame basis, allowing for accurate detection of hand movements throughout the entire duration of the video.

```
def findHands(self, img, draw = True):

imgRGB = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)

self.results = self.hands.process(imgRGB)
```

III. Identification of Finger Position and Associated Action Execution

With the help of MediaPipe, we can identify the tip identity of a particular finger and determine the coordinates of the fingers that are raised, as demonstrated in Figure 6. This information is then used to detect which finger is lifted and to execute the corresponding mouse function accordingly. By analyzing the position of the fingers in each frame, the system can accurately identify the finger movements and take appropriate actions in real-time.

Figure 5. Selecting an option

IV. For the Mouse to Perform ACTION

The computer system utilizes the pynput Python module to implement a program that enables it to recognize specific hand gestures. In this program, the computer is programmed to execute a right mouse button click if the tip ID of the middle finger is 2 and the tip ID of the index finger is 1, and both of these fingers are raised. Moreover, the program also checks if the distance between these two fingers is less than 40 pixels before triggering the right mouse button click action. By incorporating these parameters, the system can accurately identify the intended gesture and respond appropriately.

ISSN: 2583-3286(Online)

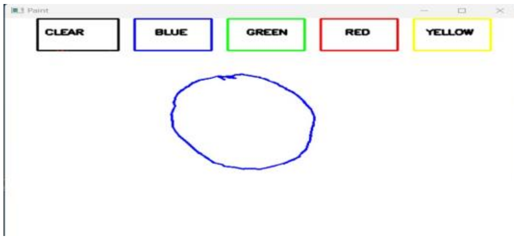

Figure 6(i). Performing an action

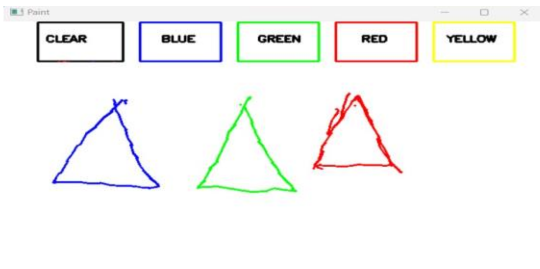

Figure 6(ii). Performing an action

VI. PROJECT SCOPE

Some of the points for the future scope of this project are as follows:

I. Implementing more advanced hand gesture recognition algorithms to improve accuracy and recognition speed.
II. Integrating the system with other technologies, such as voice recognition or eye tracking, to create a more comprehensive human-computer interface.
III. Expanding the range of recognized hand gestures to include more complex gestures, such as sign language or hand signals used in sports or dance.
IV. Developing a machine learning component that can learn and adapt to individual users' hand gestures and preferences.
V. Building a mobile app version of the system that can be used on smartphones and tablets, making it more accessible and convenient for users.
VI. Exploring potential applications of the technology in fields such as healthcare, where it could be used to assist people with disabilities or injuries in controlling devices or communicating with others.
VII. Incorporating the system into smart homes or Internet of Things (IoT) devices, allowing users to control various household appliances or devices with hand gestures.
VIII. Collaborating with experts in fields such as robotics, computer vision, or human-computer interaction to advance the technology and develop new applications.
IX. Conducting user testing and gathering feedback to improve the usability and functionality of the system for different users and environments.
X. Exploring potential security concerns and developing methods to protect users' privacy and prevent unauthorized access to the system or its data.

VII. RESULT AND DISCUSSION

The gesture-controlled whiteboard system developed using OpenCV, MediaPipe, and NumPy demonstrated promising results in terms of accuracy, performance, and usability.

To evaluate the accuracy of hand detection and finger movement tracking, we conducted experiments with a diverse set of participants performing various writing and drawing tasks on the virtual whiteboard. The system achieved a high level of accuracy in detecting hand movements and accurately mapping them to the corresponding positions on the whiteboard. The hand detection algorithm implemented using OpenCV successfully identified hands in real-time with minimal false positives or false negatives. The hand landmark estimation provided by MediaPipe accurately tracked the positions of fingertips, enabling precise tracking of finger movements for writing or drawing on the whiteboard.

In terms of performance, the system exhibited real-time responsiveness, ensuring a smooth and seamless user experience. The integration of OpenCV, MediaPipe, and NumPy facilitated efficient processing of image frames and hand landmark data, enabling real-time detection and tracking of finger movements. The system maintained a high frame rate, allowing users to write or draw on the whiteboard without any noticeable lag or delay.

User feedback and usability testing played a crucial role in assessing the system's usability. Participants expressed a positive experience with the gesture-controlled whiteboard, highlighting its intuitive nature and ease of use. They found the system responsive and accurate in capturing their finger movements, making it a suitable alternative to traditional physical whiteboards. Participants particularly appreciated the ability to erase or modify content easily by simply moving their finger in an erasing gesture. The overall user satisfaction and engagement with the system were high, indicating its potential for effective communication and collaborative environments.

The gesture-controlled whiteboard system developed using OpenCV, MediaPipe, and NumPy showcases the potential of computer vision and image processing techniques in creating interactive and intuitive user interfaces. By leveraging hand

ISSN: 2583-3286(Online)

detection and tracking algorithms, the system enables users to write and draw on a virtual whiteboard simply by moving their fingers. This technology has various practical applications, especially in educational and collaborative settings.

The accurate hand detection and tracking achieved by the system contribute to its overall effectiveness. By employing OpenCV, the system effectively identifies hands in real-time, ensuring a reliable input for finger movement tracking. The integration of MediaPipe enables precise landmark estimation, allowing for accurate mapping of finger positions on the virtual whiteboard. The use of NumPy enhances the system's efficiency by optimizing data processing operations.

The performance of the system is a key aspect for providing a seamless user experience. The real-time responsiveness and high frame rate achieved by the system ensure that users can interact with the virtual whiteboard without any noticeable delay. This real-time capability is essential for maintaining user engagement and enabling fluid writing or drawing motions.

Usability testing and user feedback are essential for evaluating the practicality and user-friendliness of the system. The positive feedback received from participants confirms the system's intuitive nature and ease of use. Participants found the gesture-controlled whiteboard to be a viable alternative to traditional physical whiteboards, appreciating the convenience of erasing and modifying content by using specific finger movements. The system's usability and user satisfaction indicate its potential for enhancing communication and collaboration in educational and interactive environments.

In conclusion, the gesture-controlled whiteboard system developed using OpenCV, MediaPipe, and NumPy demonstrates promising results in terms of accuracy, performance, and usability. The successful integration of these open-source libraries enables accurate hand detection, precise finger movement tracking, real-time responsiveness, and a seamless user experience. This technology has the potential to transform traditional whiteboard interactions and find applications in educational institutions, collaborative workplaces, and interactive presentations. Further improvements and optimizations could be explored to expand the system's capabilities and enhance its overall performance

VIII. CONCLUSION

The development of the Mark Air project enables hands-free note-taking has the potential to revolutionize the writing process. This innovative solution eliminates the need for individuals to physically hold their cell phones when quickly jotting down ideas on the go. Beyond convenience, this technology greatly enhances communication. Additionally, the program's user-friendly interface ensures accessibility for adults facing challenges with traditional keyboards.

As its functionality expands, this program can also control Internet of Things (IoT) devices, establishing itself as a vital tool in the technology field. With the introduction of air painting, it offers a unique and immersive experience adaptable to various settings. Implementing this program within smart clothing software enhances individuals' digital engagement, facilitating more efficient interaction with the digital world.

To guarantee accuracy and effectiveness, the integration of discovery algorithms such as YOLO v3 can significantly improve the program's fingerprint recognition capabilities, enhancing both speed and precision. Furthermore, the ongoing progress in Artificial Intelligence augments the future potential for advancements in hands-free writing techniques.

Ensuring the program's security is of utmost importance to prevent unauthorized use. By implementing advanced security measures, the program's integrity remains intact, providing users with a safe and trustworthy experience. Overall, this program possesses immense potential to revolutionize writing and communication practices in the digital age.

REFERENCES

[1] Y. Huang, X. Liu, X. Zhang, and L. Jin, "A Pointing Gesture Based Egocentric Interaction System: Dataset, Approach, and Application," 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Las Vegas, NV, pp. 370-377, 2016.

[2] Yang M.H, Ahuja N, Tabb M, Extraction of 2D Motion Trajectories and its Application to Hand Gesture Recognition. IEEE Trans on PAMI vol.29. 2002. pp 1062-1074

[3] S. Khan, M. E. Ali, S. Das and M. M. Rahman, "Real Time Hand Gesture Recognition by Skin Color Detection for American Sign Language," 2019 4th International Conference on Electrical Information and Communication Technology (EICT), 2019, pp. 1-6, doi: 10.1109/EICT48899.2019.9068809.

[4] P. Ramasamy, G. Prabhu and R. Srinivasan, "An economical air writing system converting finger movements to text using web camera," 2016 International Conference on Recent Trends in Information Technology (ICRTIT), 2016, pp. 1-6, doi: 10.1109/ICRTIT.2016.7569563.

[5] Tomas Simon, Hanbyul Joo, Iain A. Matthews, and Yaser Sheikh. Hand keypoint detection in single images using multiview bootstrapping. CoRR, abs/1704.07809, 2017

[6] Yuan-Hsiang Chang, Chen-Ming Chang, "Automatic

ISSN: 2583-3286(Online)

Hand-Pose Trajectory Tracking System Using Video Sequences", INTECH, pp. 132- 152, Croatia, 2010
[7] 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
[8] Q. De Smedt, H. Wannous and J. Vandeborre, "SkeletonBased Dynamic Hand Gesture Recognition," 2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2016, pp. 1206-1214, doi: 10.1109/CVPRW.2016.153.
[9] Prajakta Vidhate, Revati Khadse, Saina Rasal, “Virtual paint application by hand gesture recognition system”, 2019 International Journal of Technical Research and Applications, 2019, pp. 36-39
[10] R. Lyu et al., "A flexible finger-mounted airbrush model for immersive freehand painting," 2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS), 2017, pp. 395-400, doi:10.1109/ICIS.2017.7960025.
[11] Elena Sánchez-Nielsen, Luis Antón-Canalís, Mario Hernández-Tejera, “Hand Gesture recognition for Human Machine Interaction”, In Proc. 12th International Conference on Computer Graphics, Visualization and Computer Vision : WSCG, 2004
[12] B. Stenger, “Template based Hand Pose recognition using multiple cues”, In Proc. 7th Asian Conference on Computer Vision : ACCV 2006
[13] Lars Bretzner, Ivan Laptev and Tony Lindeberg, “Hand gesture recognition using multiscale colour features, hierarchical models and particle filtering”, in Proceedings of Int. Conf. on Automatic face and Gesture recognition, Washington D.C., May 2002
[14] M. Black, & Jepson, “ Eigen tracking: Robust matching and tracking of articulated objects using a view-based representation”, In European Conference on Computer Vision, 1996.
[15] C C Wang, K C Wang, “Hand Posture recognition using Adaboost with SIFT for human robot interaction”, Springer Berlin, ISSN 0170-8643,
[16] R. Lienhart and J. Maydt, “An extended set of Haar-like features for rapid object detection,” in Proc. IEEE Int. Conf. Image Process., 2002, vol. 1, pp. 900–903.
[17] Andre L. C. Barczak, Farhad Dadgostar, “Real-time hand tracking using a set of co-operative classifiers based on Haar-like features”, Res. Lett.
[18] Qing Chen , N.D. Georganas, E.M Petriu, “Real-time Vision based Hand Gesture Recognition Using Haar-like features”, IEEE Transactions on Instrumentation and Measurement -2007
[19] P. Viola and M. Jones, “Robust real-time object detection,” Cambridge Res. Lab., Cambridge, MA, pp. 1–24, Tech. Rep. CRL2001/01, 2001.
[20] Ju, S., Black, M., Minneman, S., Kimber, D, “Analysis of Gesture and Action in Technical Talks for Video Indexing”, IEEE Conf. on Computer Vision and Pattern Recognition, CVPR 97 ,1997
[21] Berry,G.: Small-wall, “A Multimodal Human Computer Intelligent Interaction Test Bed with Applications”, Dept. of ECE, University of Illinois at Urbana-Champaign, MS thesis
[22] Camillo Lugaresi, Jiuqiang Tang, Hadon Nash, Chris McClanahan, Esha Uboweja, Michael Hays, Fan Zhang, ChuoLing Chang, Ming Guang Yong, Juhyun Lee, Wan-Teh Chang, Wei Hua, Manfred Georg, and Matthias Grundmann. Mediapipe: A framework for building perception pipelines. volume abs/1906.08172, 2019.
[23] Zhag F, Bazarevsky, Vakunov A et.al, MediaPipe Hands: On – Device Real Time Hand Tracking, Google Research. USA. 2020. https://arxiv.org/pdf/2006.10214.pdf.