IJNRD.ORG ISSN : 2456-4184
INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD) | IJNRD.ORG
An International Open Access, Peer-reviewed, Refereed Journal

AI VirtualPainter

Prof.D.P.Mhapasekar
Computer Engineering
SSPM'sCollegeofEngineering
University of Mumbai

Mr.MayureshPatade
Computer Engineering
SSPM'sCollegeofEngineering
University of Mumbai

Mr.RutulChindarkar
Computer Engineering
SSPM'sCollegeofEngineering
University of Mumbai

Mr.AdityaGaonkar
Computer Engineering
SSPM'sCollegeofEngineering
University of Mumbai

***Abstract*—The AI Virtual Painter project is an innovative applicationthatmergescomputervision,real-timehandtracking, and artistic expression for an immersive experience. Developed for artistic expression, interactivity, education, entertainment, and as a technology showcase, it utilizes OpenCV, MediaPipe, and machine learning to enhance hand tracking accuracy. Witha user-friendly interface, it represents a significant advancement in Human-Computer Interaction, transforming digital creativity throughthesynergyofcomputervisionandartificialintelligence.**
This project holds the capacity to revolutionize our interaction with the digital world by offering a versatile tool for seamlessand intuitive digital expression.

I. Introduction

The AI Virtual Painter project embodies a creative and immersiveexplorationattheintersectionofartandtechnology, rooted in the continually advancing fields of computer vision andhuman-computerinteraction.Thecentralconceptinvolves hand tracking, where a device monitors and traces a user's hand movements in real-time through sophisticated computer vision algorithms. Employing MediaPipe and OpenCV, the project generates immersive environments for an interactive art experience, facilitating the creation of simple digital art-work without the need for physical drawing tools. Delving deeper into AI Virtual Painter reveals potential applications in education, gaming, and accessibility, reshaping the methodsof teaching and learning while empowering users to unleash their creativity. The project's features enable users to craft digital art through finger tracking facilitated by computer vision technology.

II. Characteristics

FollowingarethecharacteristicsoftheAIVirtualPainter:

- **Real-Time Hand Tracking**: The system can track the user's hand or fingers in real-time,allowing them to interact with the digitally.
- **Gesture Recognition**: The project identifies and inter-prets a variety of hand gestures such as selecting colors and deletion of drawing.
- **Color Selection**: Users can select and choose different colors from a palette to create their artwork .
- **Eraser**: The project includes tools like an eraser for correcting mistakes and the ability to delete the entire artwork.
- **User-Friendly Interface**: An intuitive and user-friendly interface simplifies the interaction with the system, mak-ing it accessible to a wide range of users.
- **Artistic Filters and Effects**: Users might have access to artisticfiltersandeffectstoenhancetheirdigitalartwork.
- **Accessibility Features**: Accessibility features to accom-modateuserswithvaryinglevelsofmobilityordexterity.
- **PerformanceOptimization**:Thesystemisoptimizedfor real-time performance to ensure smooth and responsive interaction between the user.

III. RelatedWork

Several recent studies have explored diverse approaches in the field of hand gesture recognition for interactive applica-tions.

[1] Arwoko et al. proposed a system based on Keypoint Vector for hand gesture recognition, as presented in the 2022 International Electronics Symposium (IES).

[2] Bakheet and Al-Hamadi, in their work published in the EurasipJournalonImageandVideoProcessing,introduced a robust hand gesture recognition system employing multiple shape-oriented visual cues.

[3] Harish S. presented a computer vision-based hand ges-ture recognition system in a 2023 ResearchGate publication.

[4] Reddy, Kavya, and Sudheer developed a Virtual Paint and Volume Control system using hand gestures, as described in their work.

[5] Deval et al. explored hand gesture detection and recognition in the International Journal of Advanced Research in Science, Communication, and Technology.

[6] Akshat Rastogi, Aryan Aggarwal, Ayushi Vashistha, Anmol Rastogi have worked on virtual AI painter using OpenCVandMediaPipewaspresentedintheEuropeanChem- ical Bulletin.

[7] Shinde proposed a hand gesture recognition system using a camera in an IJERT publication.

[8] Patil et al. discussed virtual painting with OpenCV using Python.

[9] NiharikaM.,NehaJ.,MamathaRao,andVidyashreeK. P.developedavirtualpaintapplicationutilizinghandgestures, as detailed in their paper.

[10] Tounderstandthebroadercontextofhuman-computer interaction, the book ”Human-Computer Interaction” by Dixet al. offers a comprehensive resource.

IV. Methodologies

TheAIVirtualPainterprojectemploysdiversedatasets for machine learning model training, integrating OpenCV, MediaPipe,andcustomalgorithms.Theuser-friendlyinterface combines real-time hand tracking, artistic effects, and educational features. Rigorous testing ensures optimal performance, delivering an innovative and immersive digital art experience for artistic expression, education, and entertainment

- **Data Collection**: Collect diverse datasets of hand movements, artistic gestures, and color preferences to train machine learning models for precise hand tracking and gesture recognition.
- **Preprocessing**: Apply preprocessing techniques to enhancedatasets,includingnormalization,backgroundsubtraction,andnoisereduction,ensuringoptimalmodel training.
- **Technology Stack Selection**: Choose OpenCV and MediaPipe for computer vision and hand tracking functionalities. Integrate machine learning algorithms to enhance hand tracking accuracy.

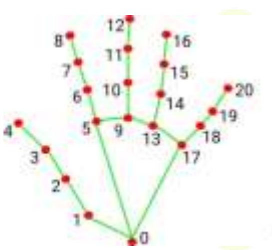

Fig.1.MediaPipe’sHandLandmarkModel [11]

- **Model Training**: Train machine learning models using the collected datasets to recognize hand movements, gestures, and color preferences.
- **System Integration**: Integrating machine learning models into a cohesive system and ensuring seamless communication for accurate hand tracking and gesture recognition.
- **User Interface Design**: Design an intuitive and user-friendly interface, incorporating artistic elements and interactive features. Consider accessibility for users with diverse abilities.
- **Enhancements**:Integratefeaturesthatenhancethevalue, suchasdynamicartisticeffects,interactiveelements,and engaging user interactions.
- **Performance Optimization**: Optimize the system for real-time performance using parallel processing and efficient algorithms, ensuring a smooth and responsive user experience.
- **Testing and Validation**: Conduct rigorous testing to validate hand tracking accuracy, gesture recognition, and overall systemfunctionality. Iterateon thedesign based ontestingresultsanduserfeedback.

V. ProposedSystem

AI Virtual Painter revolutionizes digital drawing by eliminating the need for traditional input devices. Leveraging computer vision, it utilizes the MediaPipe and OpenCV libraries for real-time hand tracking. Users can draw on computer screenandcanvaseffortlesslyusingtheirfingers,withthesystemtranslatingfingermovementsintostrokes.Thisinnovation ensures a seamless and intuitive drawing experience, marking a departure from conventional input methods. The integration of advanced technologies promises to redefine the way users interactwithdigitalcanvases,makingartisticexpressionmore accessible and engaging.

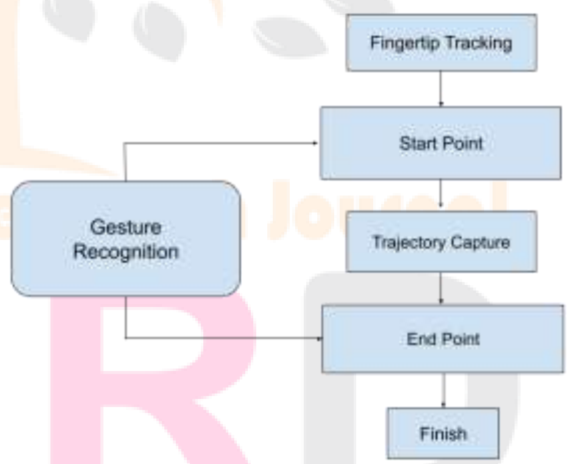

Fig.2.FlowchartofProposedSystem

VI. Algortihtms

The AI Virtual Painter project combines several computer vision and machine learning algorithms to achieve its immersive painting experience. Here’s a breakdown of the key algorithms involved:

- **BackgroundSubtraction**:Itisacommontechniqueused in computer vision for hand detection, particularly in scenarioswherethehandisthemovingobjectagainst a relatively static background. The idea is to differentiate between the foreground (hand) and the background by identifyingpixelsthathavechangedsignificantlybetween consecutive frames.

Equations and Formulas: While the process involves imagemanipulationandthresholding,therearen'tspecific equations or formulas universally applied. The key steps involve simple arithmetic operations and logical thresh-olding:

FrameDifference:

$$\text{FrameDifference}=\text{CurrentFrame}-\text{BackgroundModel} \quad (1)$$

- **KalmanFilter**: The Kalman filter combines predictions fromamathematicalmodelofthesystemwithnoisy measurements to provide a more accurate and reliable estimate of the system's state. The basic equations of the Kalman filter are as follows:

PredictionStep:

$$\hat{x}_{k|k-1}=F_k\hat{x}_{k-1|k-1}+B_ku_k \quad (2)$$

CovariancePrediction:

$$P_{k|k-1}=F_kP_{k-1|k-1}F^T_k+Q_k \quad (3)$$

UpdateStep:

$$K_k=P_{k|k-1}H^T_k(H_kP_{k|k-1}H^T_k+R_k)^{-1} \quad (4)$$

StateUpdate:

$$\hat{x}_{k|k}=\hat{x}_{k|k-1}+K_k(z_k-H_k\hat{x}_{k|k-1}) \quad (5)$$

ErrorCovarianceUpdate:

$$P_{k|k}=(I-K_kH_k)P_{k|k-1} \quad (6)$$

where:

$\hat{x}_{k|k-1}$isthepredictedstateestimateattime$k$given measurements up to time $k-1$.

$F_k$isthestatetransitionmatrix.

$B_k$is the control-input matrix.

$u_k$is the control input.

$P_{k|k-1}$isthepredictederrorcovariance.

$Q_k$istheprocessnoisecovariance.

$H_k$isthemeasurementmatrix.

$z_k$isthemeasurementvectorattime$k$. $R_k$is the noise covariance.

$K_k$istheKalman gain.

- **Contour Analysis**: Computer Vision technique used to identify and analyze the contours or outlines of objects within an image. The contours represent the boundariesof objects, and contour analysis involves extracting and processing these contours for various purposes, such as objectrecognition,shapeanalysis,andfeatureextraction. A parametric representation of a 2D contour could be given by a set of equations:

$$x(t)=f_x(t) \text{ and } y(t)=f_y(t) \quad (7)$$

Here, $x(t)$ and $y(t)$ are the parametric equations representing the contour, and t is a parameter that varies along the contour.

- **Mean Shift Algorithm**: It is a non-parametric clustering algorithm commonly used for image segmentation and tracking. It operates by iteratively shifting a set of data points toward the mean of the points in their neigh-borhood until convergence. The algorithm is especially useful for handling irregularly shaped clusters.

Foreachseedpointx_i:

$$m(x_i)=\frac{\sum_{x_j\in N(x_i)}K(x_j-x_i)\cdot x_j}{\sum_{x_j\in N(x_i)}K(x_j-x_i)} \quad (8)$$

where$N(x_i)$istheneighborhoodofx_i. Repeatthesestepsuntilconvergence.

- **Stroke Prediction and Smoothing**: Stroke prediction and smoothing are techniques commonly used in com-puter graphics and digital drawing applications to en-hance the fluidity and aesthetics of user-drawn strokes. These techniques aim to predict the trajectory of the stroke and reduce the impact of jotter or irregularities introduced by the user's hand movements.

Stroke Prediction: Let $P_i$represent the predicted point at time$i$,andP_{i-1}representthepreviouslydrawnpoint. A simple linear prediction equation might be:

$$P_i=P_{i-1}+\alpha\cdot(P_{i-1}-P_{i-2}) \quad (9)$$

where$\alpha$is a prediction coefficient.*StrokeSmoothing*: Assumingaseriesofpoints$P_i$representtherawstroke data:

$$\text{SmoothedPoint}_i=(1-\beta)\cdot P_i+\beta\cdot\text{SmoothedPoint}_{i-1} \quad (10)$$

whereβisasmoothingfactor.

- **Convolutional Neural Networks**: It enhance user inter-action by recognizing hand gestures, allowing intuitive control. Recurrent Neural Networks (RNNs) predict and smooth drawing strokes, improving fluidity and accu-racy. CNNs apply artistic filters for visual enhancement, while hand tracking accuracy benefits from CNN-based refinement.Utilizing a CNN for artistic style transfer, allowing users to apply various filters and effects to their digital artwork. RNNs analyze user interactions for a personalized interface experience. This integrated use of CNNsandRNNsrevolutionizesdigitalcreativity,offering aseamlessandintuitivetoolforartisticexpressioninreal-time.

VII. DesignDetails

TheAIVirtualPainterdesignincludesmultiplecomponents thatworktogethertocreateaseamlessuserexperience.Here's the detailed breakdown:

- **SystemArchitecture**:

- Camera Interface: This module connects to camera hardware to capture video input. Access and edit camera feeds using libraries such as OpenCV.
- MediaPipe Integration: This component integrates his MediaPipe library for hand tracking. Processes video images to identify and track the user's hands.
- Algorithmic approach: Depending on the approach chosen, this module can identify individual fingers using techniques such as contour analysis, convex hulls, andmachine learning models.
- Mapping algorithm: This component maps the detected finger position from camera space to screen space.Accuratelytrackhandmovementsandconvert them to screen coordinates.

- **UserInterface(UI)**:
  - UI elements: Graphical User Interface (GUI) elements allow users to select drawing tools (such as brush) and set parameters such as size and color.
  - Color selection interface: Users can select colors fromapalettethatcanbeimplementedasagraphical UI component.
  - Visualindicators:Theinterfaceprovidesvisualfeedback indicating the selected color, brush size, etc.
- **Handlinginteractions:**
  - Gesture recognition: If gestures are integrated, you will need a module that recognizes gestures (pinchto zoom, two-finger swipe, etc.).

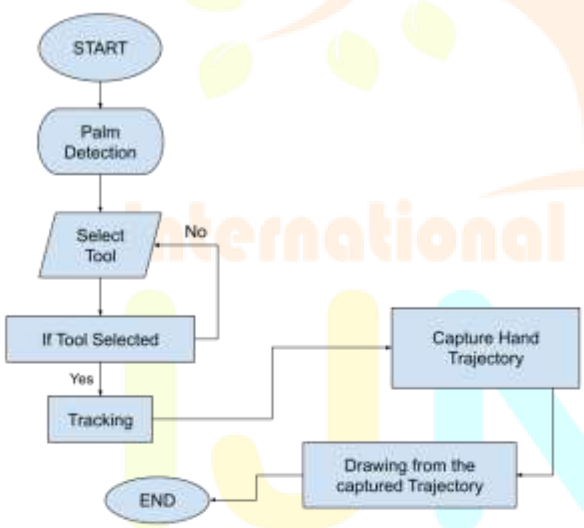

Fig.3.FlowchartofAIVirtualPainter

- **Integrationwithexternallibraries:**
  - MediaPipe integration: API Interface: This module manages integration with the MediaPipe library for hand tracking.
  - OpenCV integration: Camera feed processing: This component handles interaction with the OpenCV library for camera input processing.

VIII. EXPERIMENTALSETUP

A.DetailsAboutInputtoSystems

The AI Virtual Painter involves using computer vision techniques for hand tracking and gesture recognition to allow humans to draw on the screen with their fingers. The primary input to the application is the live video stream captured from a webcam or connected camera.

- **Camera input**: The project relies on a camera as its primary input source. This camera can be a webcam or another type of digital camera that can record video in real-time.
- **MediaPipe**: MediaPipe is a framework developed by Google that provides a variety of prebuilt machine learningmodelsforvariouscomputervisiontaskssuchashand tracking.
- **Hand Tracking**: The system uses MediaPipe's hand trackingmodeltodetectandtracktheuser'shandsin real time. The aim is to identify key points on the hand, such as the fingertips, palm, and joints.

Fig.4.HandTrackingusingMediaPipe

- **OpenCV**: OpenCV (an open-source computer vision library) may be used to access and manipulate video images. OpenCV provides a wide range of computer visiontoolsandfeaturesessentialfortaskssuchasimage processing and feature recognition.
- **Gesture Recognition**: Implementing gesture recognition tointerpretThemovementsoftheuser'shandsandposes. Forexample,certainhandgesturesmaybeassociated with actions such as drawing, erasing, changing colors, and selecting tools.
- **User interaction**: Recognized hand movements and gestures are used to interact virtually on the screen. The user'smovementiscapturedwiththeirfinger,thesystem interprets the movement and draws corresponding lines and shapes on the screen.

IX. IMPLEMENTATIONSTEP

Implementing the AI Virtual Painter project involves several steps, combining computer vision, hand tracking, and interactive graphics. Below is a simplified outline of the implementation process:

- **Set Up Development Environment**: Install necessary software and libraries, including Python, MediaPipe, OpenCV, and any other dependencies.
- **UnderstandMediaPipeandOpenCV**:Familiarizeyourself with the functionalities of MediaPipe and OpenCV, especially the modules related to hand tracking and computer vision.
- **Camera Setup**: Connect and configure the camera to capture video input. Ensure that the camera has a clear view of the user's hand movements.
- **Initialize Hand Tracking**: Use MediaPipe to initialize handtrackingbydetectingandlocalizingtheuser'shands in the camera feed.

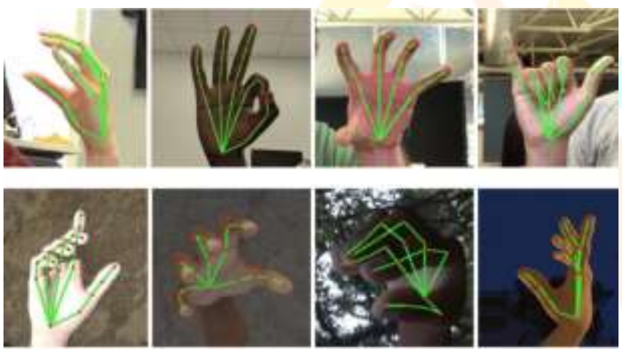

Fig.5.HandPositionTrackingusingCamera [12]

- **Extract Hand Landmarks**: Utilize the hand tracking results to extract the landmarks or key points of theuser's hand. These landmarks will be used to track hand movements.
- **Define Canvas**: Set up a virtual canvas or interactive space where users can draw. This could be a 2D or 3D space, depending on the project's goals.

Fig.6.Header

- **Optimization**:Optimizethecodeforefficiency,consideringfactorssuchasframerate,responsiveness,andoverall performance.
- **Real-time Rendering**: Implement real-time rendering of thedigitalartworkontheinteractivecanvas.Thisrequires updating the display continuously as the user interacts with the system. [13]

Fig.7.RealTimeRendering

- **Testing**: Test the system extensively to ensure that hand tracking is accurate, drawing is smooth, and the overall user experience meets expectations.
- **Draw on the Canvas**: Based on the tracked hand landmarks, implement a drawing mechanism that allows users to create digital artwork. This could involve using gestures or movements to control the drawing tools.

Fig.8.Canvas

- **Iterate and Improve**: Gather feedback from users and iterate on the project, making improvements based on user experience and any identified issues.

X. PERFORMANCEMETRICS

The performance evaluation parameters for the project include accuracy in hand and gesture recognition, low latency for real-time interactions, high frame rate for smooth interactions, tracking robustness under various conditions, precision and recall in gesture recognition, quick response time, and compatibility across diverse devices and operating systems.

For the given application the video is recorded in 30fps for time 10s. Here are some potential performance metrics with associated equations:

- **Hand Tracking Accuracy**: Percentage of accurately tracked hand movements.

$$\text{Acc.} = \frac{\text{No.ofaccuratelytrackedframes}}{\text{Totalnumberofframes}} \times 100\%$$
$$= \frac{255}{300} \times 100\%$$
$$= 85\% \quad (11)$$

- **GestureRecognitionAccuracy**:Accuracyinrecogniz- ing predefined gestures.

$$\text{Acc.} = \frac{\text{No.ofcorrectlyrecognizedgestures}}{\text{Totalnumberofgestures}} \times 100\%$$
$$= \frac{224}{300} \times 100\%$$
$$= 74.6\% \quad (12)$$

- **Frame Rate**: Number of frames processed per second.

$$\text{Frame Rate} = \frac{\text{No. of Frames}}{\text{TotalTime(inseconds)}}$$
$$= \frac{300}{100}$$
$$= 30\text{fps} \quad (13)$$

- **Drawing Accuracy**: Accuracy in translating hand movements to on-screen strokes.

$$\text{Accuracy} = \frac{\text{No.ofaccuratelydrawnstrokes}}{\text{Totalnumberofstrokesattempted}} \times 100\%$$
$$= \frac{282}{300} \times 100\%$$
$$= 94\% \quad (14)$$

- **Responsiveness**: Time delay between a user's action and system response.

$$\text{Responsiveness} = \frac{\text{No.ofinteractions}}{\text{Totalprocessingtime}}$$
$$= \frac{20}{30}$$
$$\approx 0.67s \quad (15)$$

- **User Satisfaction(Survey-based)**: User feedback on the overall satisfaction with the system.

$$\text{UserSatisfaction} = \frac{\text{Sumofuserratings}}{\text{No.ofuserresponses}} \times 100\%$$
$$= \frac{20}{22} \times 100\%$$
$$= 90\% \quad (16)$$

XI. Conclusion

The AI Virtual Painter project represents a groundbreaking fusionofcomputervision,gesturerecognition,andreal- time interaction, transforming digital creativity. Through the integration of technologies like MediaPipe and OpenCV, the system allows users to draw on a screen using their fingers, eliminatingtheneedfortraditionalinputdevices.Theproject excels in accuracy, tracking robustness, and gesture recognition precision. It achieves low latency and high frame rates, ensuring a seamless and responsive drawing experience.

The precision and recall metrics validate the robustness of gesture recognition, while response time optimization contributes to smooth user interactions. The system's compatibility across devices and operating systems enhances its accessibilityandusability.Inconclusion,theAIVirtualPainter notonlypioneersintuitivedigitalpaintingbutalsoexemplifies thepotentialofcomputervisionandgesture-basedinterfaces in revolutionizing interactive and immersive applications.

Thus,theAIVirtualPainterprojectoffersaglimpseinto the future of how technology can enhance creativity and interaction.Withcontinueddevelopmentandinnovation,ithas the potential to become a valuable tool for artistic expression, education, and entertainment.

Acknowledgment

We would like to express our sincere gratitude to our supervisor, Prof. Mhapasekar D.P. , for his invaluable guidance, patience, and support throughout the course of this research. His insightful feedback and constructive criticism have been instrumental in shaping this work. I would also like to thank the participants who generously gave their time and shared their experiences for this study. Finally, I would liketo thank my family and friends for their unwavering support and encouragement.

References

[1] Heru Arwoko, Eko Mulyanto Yuniarno, and Mauridhi Hery Purnomo.HandgesturerecognitionbasedonKeypointVector. *2022InternationalElectronics Symposium (IES)*, 8 2022.

[2] Samy Bakheet and Ayoub Al-Hamadi. Robust hand gesture recognitionusing multiple shape-oriented visual cues.*Eurasip Journal on Imageand Video Processing*, 2021(1), 7 2021.

[3] HarishS.ComputervisionbasedHandgesturerecognitionsystem. *ResearchGate*,22023.

[4] D Sudheer Reddy, V Shiva Narayana Reddy K. Kavya.Virtual PaintAnd Volume Control Using Hand Gesture, 7 2022.

[5] Ms.Nimisha Deval, Anisha Sawant, Monali Shinde, and Kshitija Fase.Hand gesture detection and recognition.*International Journal ofAdvanced Research in Science, Communication and Technology*, pages618–621, 6 2022.

[6] Akshat Rastogi and Aryan Aggarwal. VIRTUAL AI PAINTER USINGOPENCV AND MEDIAPIPE.*European Chemical Bulletin*, Volume -12, Special Issue-1(Part-A) (2023)(Special Issue-1(Part-A) (2023)), 3 2023.

[7] VirajShinde.Handgesturerecognitionsystemusingcamera.*IJERT*,1 2014.

[8] Yash Patil, Mihir Paun, Deep Paun, Karunesh Singh, and Vishal KisanBorate.Virtual Painting with Opencv Using Python.*InternationalJournalofScientificResearchinScienceandTechnology*,5(8):189–194,12 2020.

[9] NiharikaM,NehaJ,MamathaRao,andVidyashreeKP.VirtualPAINTAPPLICATIONUSINGHANDGESTURES. https://www.irjet.net/archives/V9/i4/IRJET-V9I4399.pdf, 4 2022. [Online;accessed2023-10-19].

[10] Alan Dix, Janet Finlay, Gregory D. Abowd, and Russell Beale. *Human-computer interaction*.Pearson Education, 1 2003.

[11] MediaPipe.MediaPipe'sHandLandmarkModel.

[12] On-Device,Real-TimeHandTrackingwithMediaPipe,82019.

[13] PedroRavaglia.GitHub - PedroRavaglia/AI-Virtual-Painter: Pythonproject that tracks hand position in real-time and draws on screen usingthe index finger.