MINI-PROJECT REPORT

VIRTUAL CANVAS
USING OPENCV AND MEDIAPIPE

SUBMITTED BY
HARIKRISHNAN V. B
LMC22MCA2011

DEPARTMENT OF COMPUTER APPLICATION
[Affiliated to APJ Abdul Kalam Technological University, Kerala (KTU)]

LOURDES MATHA COLLEGE OF SCIENCE AND TECHNOLOGY
KUTTICHAL, THIRUVANANTHAPURAM-695574

(MANAGED BY THE ARCHDIOCESE OF CHANGANASSERY)

VIRTUAL CANVAS USING OPENCV AND MEDIAPIPE

A Project Report

Submitted By:

Harikrishnan V. B - LMC22MCA2011

in partial fulfillment of the requirements for the award of
the degree in

MASTER OF COMPUTER APPLICATIONS

At

DEPARTMENT OF COMPUTER APPLICATIONS

LOURDES MATHA COLLEGE OF SCIENCE AND TECHNOLOGY
KUTTICHAL, THIRUVANANTHAPURAM-695574

(AFFILIATED TO APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY, KERALA)

DECEMBER 2023

LOURDES MATHA COLLEGE OF SCIENCE AND TECHNOLOGY
KUTTICHAL, THIRUVANANTHAPURAM – 695574

(Affiliated to APJ Abdul Kalam Technological University, Kerala)

DEPARTMENT OF COMPUTER APPLICATIONS

CERTIFICATE

This is to certify that the project work entitled **“Virtual Canvas using OpenCV and Mediapipe”** is a Bonafide record of the work done by **Mr. Harikrishnan V. B**, Reg No **LMC22MCA2011**, student of Department of Computer Applications, Lourdes Matha College of Science And Technology, Kuttichal, Thiruvananthapuram, affiliated to the APJ Abdul Kalam Technological University, Kerala from August 2023 to December 2023 in partial fulfillment of the requirements for the award of the Degree of Master of Computer Applications from APJ Abdul Kalam Technological University, Kerala.

Prof. Bismi K Charleys
(Internal Guide)

Internal Examiner

Prof. Bismi K Charleys
(Head of the Department)

DECLARATION

I undersigned here by declare that the project report **“Virtual Canvas using OpenCV and Mediapipe”** submitted for partial fulfilment of the requirements for the award of degree of Master of Computer Applications of the APJ Abdul Kalam Technological University, Kerala. This submission represents my idea in my own words and, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to the ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact of source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University.

Place: Trivandrum Harikrishnan V B

Date:__/__/2023

ACKNOWLEDGEMENT

An endeavor over a long time can be successful only with advice and support of many well-wishers. I wish to place on record my profound indebtedness and gratitude to all those who have contributed directly or indirectly to make this project work a success.

At the very onset, I express my gratitude to God Almighty who sheltered me under his protective wings and showered on innumerable blessings throughout the period of this Master of Computer Application.

It is a great pleasure to express my sincere gratitude to **Rev. Fr. Bejoy Arackal**, Director and **Dr. Beshiba Wilson**, Principal Lourdes Matha College of Science and Technology for permitting me to do this project with the fullest spirit.

I am highly obliged to **Prof. Bismi K Charleys** Head of the Department of Computer Applications of Lourdes Matha College of Science and Technology, for being the source of inspiration throughout the course and for her valuable guidance.

With heart full of thanks, I would like to take up this opportunity to wish my Internal guide **Prof. Bismi K Charleys** Head of the Department of Computer Applications and all staffs of Department of Computer Applications for their endless support, encouragements and suggestions in various stages of the development of this project.

With immense love and gratitude, I thank every unknown member of numerous amounts of open-source communities for all the selfless works and contributions they've made. Without them and their help, I wouldn't have made it here. Finally, I wish to express my sincere gratitude to all our friends who directly or indirectly contributed in this venture.

CONTENTS

Content — Page No

ABSTRACT 9

CHAPTER 1

1.INTRODUCTION 10

1.1 General Introduction 11

1.2 Goal of the Project 12

CHAPTER 2

2.LITERATURE SURVEY 13

2.1 Study of Similar Work 14

2.1.1 Existing System 14

2.1.2 Drawbacks of Existing System 14

CHAPTER 3

3.OVERALL DESCRIPTION 16

3.1 Proposed System 17

3.2 Features of Proposed System 17

3.3 Functions of Proposed System 17

3.4 Requirements Specification 18

3.5 Feasibility Analysis 19

3.5.1 Technical Feasibility 19

3.5.2 Operational Feasibility 19

3.5.3 Economical Feasibility 20

3.5.4 Behavioral Feasibility 20

CHAPTER 4

4.OPERATING ENVIRONMENT 21

4.1 Hardware Requirements 22

4.2 Software Requirements 23

4.3 Tools and Platforms 24

4.3.1 Python 24

4.3.2 NumPy 24

4.3.3 Mediapipe 24

4.3.4 Mediapipe Hands 25

4.3.5 Tkinter 25

CHAPTER 5

5.DESIGN 26

5.1 System Design 27

5.2 Process flow Diagram 28

5.3 Input Design 29

5.4 Output Design 30

5.6 Program Design 32

CHAPTER 6

6.FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS 33

6.1 Functional Requirements 34

6.2 Non-Functional Requirements 34

CHAPTER 7

7.TESTING 36

7.1 System Testing 37

7.2 Unit Testing 37

7.3 Integration Testing 38

7.4 Black Box Testing 38

7.5 Validation Testing 38

7.6 Output Testing 39

7.7 User Acceptance Testing 39

7.8 White Box Testing 39

7.9 Test Data and Results 40

CHAPTER 8

8.RESULTS AND DISCUSSION 44

8.1 Results 45

8.2 Screenshots 47

CHAPTER 9

9.CONCLUSION 50

9.1 System Implementation 51

9.2 Future Enhancement 55

9.3 Conclusion 57

CHAPTER 10

10.BIBLIOGRAPHY 58

10.1 Books 59

10.2 Websites 59

10.3 References 59

APPENDICES 61

GIT HISTORY 62

ABSTRACT

The “Virtual Canvas” seamlessly integrates Media Pipe’s precise hand tracking with OpenCV's computer vision, resulting in an immersive digital experience. At its core, the system leverages the accuracy of hand tracking technology to detect and interpret intricate hand gestures, translating them into digital commands for drawing, fine-tuning, and manipulating elements on the virtual canvas.

Users can draw and fine-tune digital elements using intuitive hand gestures, effectively erasing the boundaries between physical and digital art. The user-friendly interface provides simple controls for adjusting colures and brushes including eraser. Additionally, image suggestion and generation features are implemented.

CHAPTER1

INTRODUCTION

1.1 GENERAL INTRODUCTION

The mouse, keyboard, remote control, touch screen, and other similar devices are currently the main ways in which people and machines interact directly. People communicate primarily through non-contact methods that are natural and intuitive, such as sound and physical motion. Many researchers have attempted to assist computers in identifying other people's intentions and information by using non-contact methods such as voice, facial expressions, physical motions, and gestures.

Gestures are the most important aspect of mortal language, and they also play an important role in mortal communication. They are thought to be the most basic way for humans and computers to communicate. Gesture recognition encompasses sign language recognition, robotics, and other applications.

Gesture recognition for HCI applications typically employs two methods. The first relies on wearable or direct physical approaches, whereas the second relies on computer vision and does not require the use of any sensors. In the wearable or direct contact technique, the data-glove, which is made up of sensors to capture hand motion and location, is used. The camera is used in the vision-based technique to provide contactless communication between humans and machines.

In this project, we implement by integrating Media Pipe’s precise hand tracking with OpenCV's computer vision, resulting in an immersive digital experience. Additionally, image suggestion and generation features are implemented.

1.2 GOAL OF THE PROJECT

This project allows the user

Image Suggestion and Recognition: Users can seamlessly use creative potential of the application with image suggestion and recognition.

Enable users to create digital artwork using various tools and techniques, replicating the experience of traditional art mediums in a virtual environment.

CHAPTER 2

LITERATURE SURVEY

2.1 STUDY OF SIMILAR WORK

There have been a number of studies on the use of Media Pipe and OpenCV that interpret intricate hand gestures, translating them into digital commands for drawing, fine-tuning, and manipulating elements on the virtual canvas.

2.2 EXISTING SYSTEM

In common methods, we have observed that the mouse, keyboard, remote control, touch screen including wearable devices are currently the main ways in which people and machines interact directly.

Another application, in which the user can paint using his fingertip. Unlike paint, no need to drag or draw, it recognizes the hand movement in the direction of a needed diagram. By using OpenCV users can paint on the canvas and Media Pipe for the hand-tracking. However, a few disadvantages are that it lacks in image suggestion and generation.

2.1.2 DRAWBACK OF EXISTING SYSTEMS

Lack of image suggestion and recognition: Users may find it challenging to fully explore the creative potential of the application, as the absence of these features limits their ability to receive suggestions and identify images within the virtual canvas.

Lack of a manual: Non-educated users may encounter challenges in utilizing the virtual canvas, as the absence of guidance hinders their ability to navigate and fully engage with the digital platform.

Security issues: Lack of a login page for a virtual canvas introduces vulnerabilities in user authentication, posing potential risks of unauthorized access and compromising the security of the digital environment.

CHAPTER 3

OVERALL DESCRIPTION

3.1 PROPOSED SYSTEM

The existing Virtual Canvas are not dealing with image suggestion and recognition. We propose such a project proposal with an aim that implementing image suggestion and recognition. In our system, users will receive suggestions while drawing images, and the selected user-generated image will be generated accordingly.

3.2 FEATURES OF PROPOSED SYSTEM

- The user will receive image suggestions while drawing images.
- Login Page is implemented to compromise the security of the digital environment.
- Availability of manual provides guidance to non-educated users.
- User of any age can use this system efficiently.

3.3 FUNCTIONS OF PROPOSED SYSTEM

Image Suggestion and Recognition: Users can seamlessly use creative potential of the application with image suggestion and recognition.

Efficient: User Authentication secures the application.

Availability of Manual: Helps the non-educated user to navigate and fully engage with the digital platform.

User-Friendly: User Interface (UI) makes it accessible to users.

3.4 REQUIREMENT SPECIFICATION

System Analysis is the process of studying a procedure or business in order to identify its goals and purposes and create systems and procedures that will achieve them in an efficient way. System analysis relates closely to requirements analysis. Requirement specification simply means figuring out what to make before you make it. It determines what people need before you start developing a product for them. Requirement definition is the activity of translating the information gathered into a document that defines a set of requirements. These should accurately reflect what the customer wants. It is anabstract description of the services that the system should provide and the constraints under the system must operate.

The requirements of specification of the proposed system are as follows:

Accuracy

The Virtual Canvas should be accurate on hand tracking.

Speed

The proposed system should be in real time for generating results.

Flexible

The proposed system should be flexible to new updates and patches in near future.

Good Interface

The proposed system should maintain good interface even after upgradations.

3.5 FEASIBILITY ANALYSIS

An initial investigation culminates in a proposal that determine whether an ultimate system is feasible. When a proposed system is made and approved it initiates a feasibility study. The purpose of the feasibility study is to identify various candidate systems and evaluates whether they are feasible by considering technical, economical and operational feasibility and to recommend to best candidate system. The feasibility of such a program is listed in a simulated environment. Once all features are working property in a simulated environment, we can implement in a real platform. During product engineering, we consider following types of feasibility:

3.5.1 TECHNICAL FEASIBILITY

Technical feasibility identifies whether the proposed system can be developed with the existing technologies and available hardware and software resources. As part of the technical feasibility of the system, the following points are to be emphasized. Technical feasibility is frequently the most difficult area to assess at the stage of the product engineering process. It is essential that the process of analysis and definition be conducted in parallel with an assessment of technical feasibility. The considerations that are normally associated with technical feasibility are development risk, resource availability and technology.

3.5.2 OPERATIONAL FEASIBILITY

Proposed projects are beneficial only if they can be turned into information systems that will meet the operating requirements of the organization. This test of feasibility asks if the system will work when it is developed and installed. This project satisfies all the operational conditions. The project is found to work well on installation, all types of users can operate the system without any difficulty. User interfaces are designed in such a way that even ordinary users without having much knowledge in computer technology can easily operate the system. The access time of data is considerably low and the operation is less time consuming.

3.5.3 ECONOMICAL FEASIBILTY

An evaluation of development cost weighted against the ultimate income or the benefit derived from the developed system or product. Economic feasibility of a system means that the cost incurred in developing and implementing a system should not be higher than the financial benefits obtained by the users. During the economic feasibility study the following points were investigated.

- The cost to conduct a full system investigation
- The cost of hardware and software for the application being developed.
- The benefits derived by the users in terms of time, effort, accuracy of information, better decision making. Etc. are quantified and compared.

3.5.4 BEHAVIORAL FEASIBILITY

Behavioral Feasibility evaluates and estimates the user attitude or behavior towards the development of new system. It helps in determining if the system requires special effort to educate, retrain, transfer, and changes in employee’s job status on new ways of conducting business.

CHAPTER 4

OPERATING ENVIRONMENT

4.1 HARDWARE REQUIREMENT

Development

- Processor: Minimum dual core intel/AMD processor at 2Ghz
- Hard disk: 500MB Minimum
- RAM: 8 GB
- Monitor: Standard monitor with at least 1024x768 resolution.
- Web camera: minimum 720p or 1080p camera

Deployment

- Processor: Minimum dual core intel/AMD processor at 2Ghz
- Hard disk: 500MB Minimum
- RAM: 8 GB
- Monitor: Standard monitor with at least 1024x768 resolution.
- Web camera: minimum 720p or 1080p camera

4.2 SOFTWARE REQUIREMENT

Development

- Operating System: Windows 10
- IDE: PyCharm
- GUI: Tkinter
- Libraries Used: NumPy, Tkinter, media pipe, time, OpenCV

Deployment

- Operating System: Windows 7 or above

4.3 TOOLS AND PLATFORMS

4.3.1 PYTHON

Application Development, as well as for use as a scripting or glue language to connect existing components Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid together. Python's simple, easy to learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms, and can be freely distributed.

4.3.2 NumPy

NumPy is the fundamental package for scientific computing in Python. It is a Python library that provides a multidimensional array object, various derived objects (such as masked arrays and matrices), and an assortment of routines for fast operations on arrays, including mathematical, logical, shape manipulation, sorting, selecting, I/O, discrete Fourier transforms, basic linear algebra, basic statistical operations, random simulation and much more.

4.3.3 Mediapipe

Media pipe is a machine learning pipeline framework that may be used to handle audio, video, and other forms of time-series data. The Raspberry Pi and the Jetson Nano are only two examples of the embedded devices that are compatible with this framework. The Media Pipe Toolkit includes both the Infrastructure and the Fixes.

4.3.4 Media pipe Hands

Media Pipe Hands is a sophisticated tool for monitoring hands and fingers. Using ML, it can deduce 21 unique 3D landmarks of a hand from a single image. Our solution delivers real-time performance on a cell phone and even scalable to many hands, whereas the existing state-of-the-art relies mostly on powerful desktop systems for inference. By making this hand perception capability available to the broader research and development community, we want to inspire the creation of novel use cases, which in turn will pave the way for exciting new applications and lines of inquiry.

4.3.5 Tkinter

The Tkinter package (“Tk interface”) is the standard Python interface to the Tcl/Tk GUI toolkit. Both Tk and Tkinter are available on most Unix platforms, including macOS, as well as on Windows systems. Tkinter supports a range of Tcl/Tk versions, built either with or without thread support. The official Python binary release bundles Tcl/Tk 8.6 threaded. See the source code for the Tkinter module for more information about supported versions. Tkinter is not a thin wrapper, but adds a fair amount of its own logic to make the experience more pythonic. This documentation will concentrate on these additions and changes, and refer to the official Tcl/Tk documentation for details that are unchanged.

4

CHAPTER 5

DESIGN

5.1 SYSTEM DESIGN

System Design involves translating system requirements and conceptual design into technical specifications and general flow of processing. After the system requirements have been identified, information has been gathered to verify the problem and after evaluating the existing system, a new system is proposed. System Design is the process of planning of new system or to replace or complement an existing system .It must be thoroughly understood about the old system and determine how computers can be used to make its operations more effective.

System design sits at technical the kernel of system development. Once system requirements have been analysed and specified system design is the first of the technical activities-design, code generation and test- that required build and verifying the software. System design is the most creative and challenging phases of the system life cycle. The term design describes the final system and the process by which it is to be developed.System design is the high-level strategy for solving the problem and building a solution. System design includes decisions about the organization of the system into subsystems, the allocation of subsystems to hardware and software components and major conceptual and policy decision that forms the framework for detailed design.

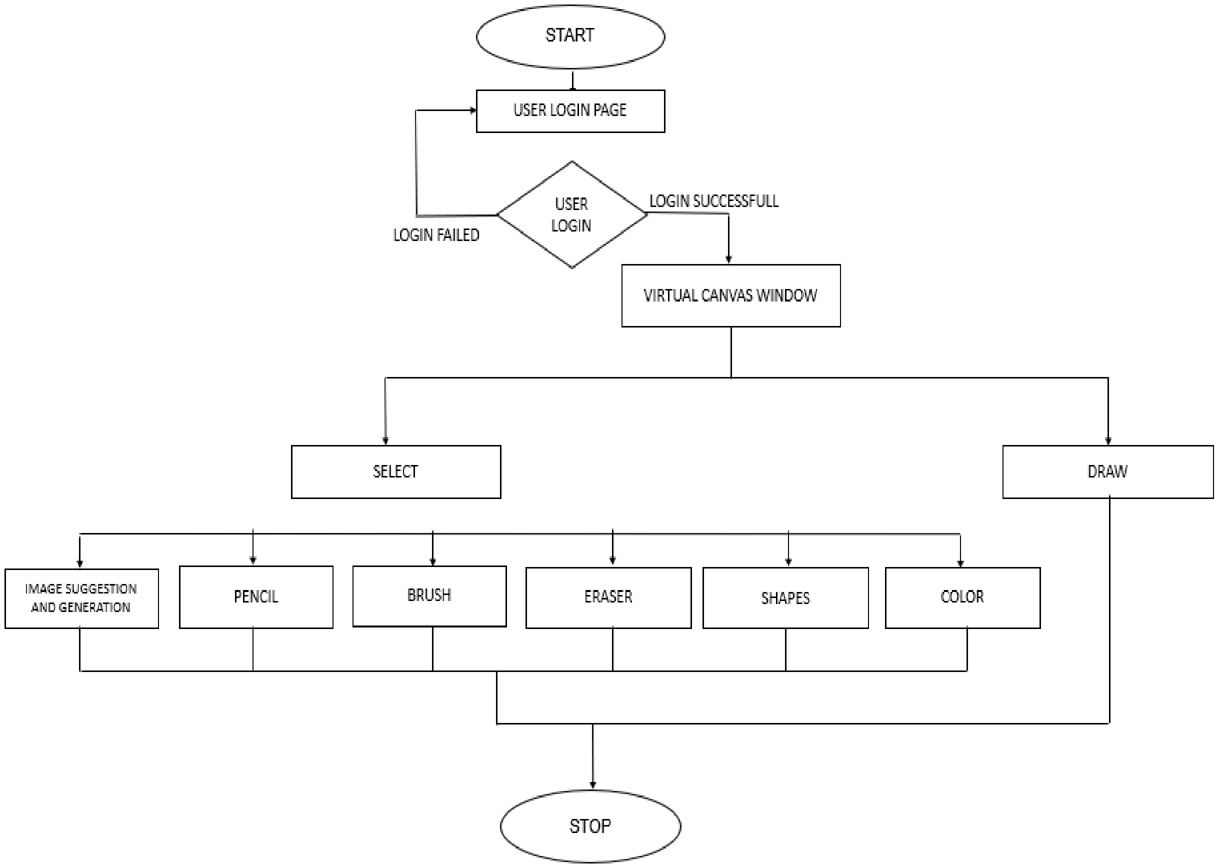
There are two levels of system design:

- Logical design.
- Physical design.

In the logical design, the designer produces a specification of the major features of the system which meets the objectives. The delivered product of logical design includes current requirements of the following system components:

- Input design.
- Output design.

5.2 PROCESS FLOW DIAGRAM



5.3 INPUT DESIGN

The input design is the process of converting the user-oriented inputs in to the computer-based format. The goal of designing input data is to make automation as easy and free from errors as possible. The input design requirements such as user friendliness, consistent format and interactive dialogue for giving the right message and help for the user at right time are also considered for the development of the project.

The following points should be considered while designing the input:

- What data to input?
- What medium to use?
- How the data should be arranged or coded?
- The dialogue to guide users in providing input.
- Data items and transactions needing validation to detect errors.

The user interface design is very important for any application. The interface design defines how the software communicates within itself, to system that interpreted with it and with human who use it. The interface design is very good; the user will fall into an interactive software application.

Input design is the process of converting user-oriented inputs to a computer-based format. The data is fed into the system using simple interactive forms. The forms have been supplied with messages so that user can enter data without facing any difficulty. The data is validated wherever it requires in the project. This ensures that only the correct data have been incorporated into the system. Inaccurate processing of data is the most common cause of errors in data processing. Errors entered by data entry operators can be controlled by

correct input design. This type of input design allows user to input only the required data into the processing units and also these input from check for validation of the input values, thus preventing errors.

The input design is made into user-friendly atmosphere where the user can perform the daily routine work without any one help.

5.4 OUTPUT DESIGN

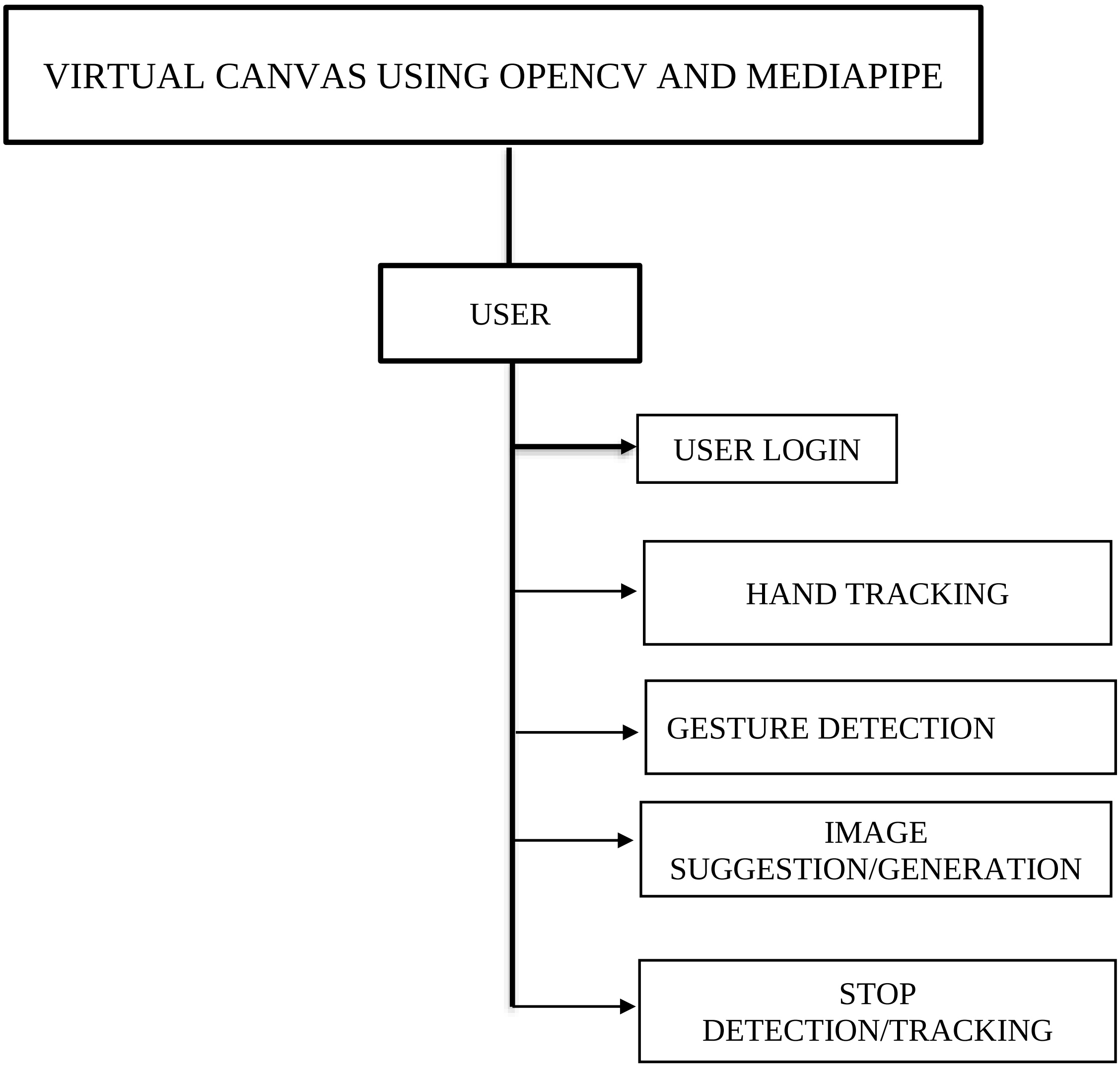
Output generally refers to the results and information that are generated by the system. When designing output, system analyst must accomplish the following:

- Determine what information to present.
- Decide whether to display, print the information and select the output medium.
- Arrange the presentation of information in an acceptable format.
- Decide how to distribute the output to intended recipients.

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any systems, results of processing are communicated to the user and to other systems through outputs. In the output design, it is determined how the information is to be displayed for immediate need.

The major idea of output is to convey information so its layout and design need careful consideration. Efficient, intelligible output design improves the system relationship with the users and help in making decisions. The output designs decide how well the implementation of the system has been useful to the user. The output design should be understandable to the user and it must offer great convenience. The one who look into the reports or output will get the impression of how well the system performs.The objective of the output design is to convey the information of all the past activities, current status and emphasize important events. The output generally refers to the results and information that is generated from the system. Outputs from the computers are required primarily to communicate the result of processing to the users. They are also used to provide a permanent copy of these results for later consideration.

5.4 PROGRAM DESIGN


VIRTUAL CANVAS USING OPENCV AND MEDIAPIPE
USER
USER LOGIN
HAND TRACKING
GESTURE DETECTION
IMAGE SUGGESTION/GENERATION
STOP DETECTION/TRACKING

CHAPTER 6

FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS

6.1 FUNCTIONAL REQUIREMENTS

The functional requirements represent the intended behavior of the system. The proposed system consists of

6.1.1 Drawing Tools: The system should provide a variety of drawing tools, such as brushes, pens, erasers, and colors, to allow users flexibility in creating their digital art.

6.1.2 Image Suggestion: Implement an intelligent image suggestion system that analyses user input and provides relevant image suggestions based on drawing context.

6.1.3 User Authentication: If applicable, incorporate user authentication features to allow secure access and personalized experiences.

6.2 NON-FUNCTIONAL REQUIREMENTS

6.2.1 Performance: The system should respond quickly to user inputs, ensuring a smooth and lag-free drawing experience even with complex and detailed artworks.

6.2.2 Usability: The user interface should be intuitive and user-friendly, catering to users of varying skill levels and ensuring an accessible creative experience.

6.2.3 Security: If the virtual canvas involves user accounts, robust security measures should be in place to protect user data and ensure secure access.

6.2.4 Compatibility: The application should be compatible with various devices and platforms to allow users flexibility in accessing and working on their projects.

6.2.5 Accessibility: Ensure that the virtual canvas is accessible to users with disabilities, adhering to accessibility standards to provide an inclusive experience.

6.2.6 Documentation: Provide comprehensive documentation, including a user manual and developer documentation, to assist users and maintainers in understanding and using the system effectively.

6.2.7 Maintainability: The codebase should be well-structured and documented for ease of maintenance and future enhancements.

CHAPTER 7

TESTING

7.1 SYSTEM TESTING

System Testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently as expected before live operation commences. It certifies that the whole set of programs hang together. System testing requires a test plan that consists of several keys, activities and steps to run program, string, system and user acceptance testing. The implementation of newly designed package is important in adopting a successful new system.
Testing Objectives

Testing is the process of correcting a program with intend of finding an error.

- A good test is one that has a high probability of finding a yet undiscovered error.
- A successful test is one that uncovers an undiscovered error.

- There are different types of testing methods available:

7.1 UNIT TESTING

In this testing we test each module individually and integrate the overall system. Unit testing focuses verification efforts on the smaller unit of software design in the module. This is also known as ‘module’ testing. The modules of the system are tested separately. The testing is carried out during programming stage itself. In this testing step each module is found to work satisfactory as regard to the expected output from the module. There are some validation checks for verifying the data input given by the user. It is very easy to find error and debug the system.

7.2 INTEGRATION TESTING

Data can be lost across an interface; one module can have an adverse effect on the other sub functions when combined by May not produce the desired major functions. Integrated testing is the systematic testing for constructing the uncover errors within the interface. This testing was done with sample data. The need for integrated test is to find the overall system performance.

7.3 BLACK BOX TESTING

This testing attempts to find errors in the following areas or categories: Incorrect or missing functions, interface errors, errors in data structures, external database access, performance errors and initialization and termination errors.

7.4 VALIDATION TESTING

At the culmination of Black Box testing, software is completely assembled as a package, interface errors have been uncovered and corrected and final series of software tests, validation tests begin. Validation testing can be defined in many ways but a simple definition is that validation succeeds when the software functions in a manner that can be reasonably accepted by the customer.

After validation test has been conducted one of the two possible conditions exists.

- The function or performance characteristics confirm to specification and are accepted.
- A deviation from specification is uncovered and a deficiency list is created.

7.1 OUTPUT TESTING

After performing the validation testing, the next step is output testing of the proposed system since no system could be useful if it doesn't produce the required data in the specific format. The output displayed or generated by the system under consideration is tested by, asking the user about the format displayed. The output format on the screen is found to be correct as the format was designed in the system according to the user needs. Hence the output testing doesn't result in any correction of the system

7.2 USER ACCEPTANCE TESTING

User acceptance of the system is the key factor for the success of the system. The system under consideration is tested for user acceptance by constantly keeping in touch with prospective system at the time of developing and making change wherever required. This is done with regard to the following points:

- Output Screen design.
- Input Screen design.
- Menu driven system.

7.3 WHITE BOX TESTING

White box testing is a testing case design method that uses the control structure of the procedural design to derive the test cases. The entire independent path in a module is exercised at least once. All the logical decisions are exercised at least once. Executing all the loops at boundaries and within their operational bounds exercise internal data structure to ensure their validity. In our project testing was

Initially each module was tested separately to check whether they gave the desired output for the given input. The forms used to enter data by user were validated and appropriate error messages were displayed if incorrect data was entered. Once the data was entered correctly, the processing was done and testing was done to check whether the correct output was obtained. Once the test cases were conducted successfully for each module, the modules were integrated together as a single system. After integration, the test cases were again applied to check whether the entire system as a whole produced the desired output. At times, the test cases failed and the shortcomings were noted down and appropriate corrections were done. Once the integration testing was performed correctly, output testing was done and it did not result in any change or correction in the system. Black box testing and white box testing was also conducted successfully. All the loops, decisions, relations were executed at least once before giving it to the users for testing. In black box testing, it was checked whether the data in the proper format was stored in the database or not. Also, it was checked whether the interfaces were working properly or not. On successful completion of these tests, the system was then given to undergo user acceptance testing where the users entered test data to check whether the correct output was obtained. The users were satisfied with the output and thus the testing phase was completed successfully.

7.4 TEST DATA AND RESULTS

The primary goal of software implementation is the production of source code that is easy to read and understand. Clarification of source code helps in easier debugging, testing and modification. Source code clarification is enhanced by structural coding techniques, by good coding style, by appropriate supporting documents, by good internal comments and by the features provided in the modern programming language. In our implementation phase, source code contains both global and formal variables. It contains predefined functions as well as the user defined functions. The result of the new system is compared with old system and supposes if the result is wrong the error must be debugged. After the acceptance of the system by the user, the existing system should be replaced by this system. Any user handles this package very easily. It does not require any intensive training for the user. Procedures and functions involved in this system are very simple that anyone can understand and correspondingly act to the system with no difficulty.

7.5 USER ACCEPTANCE TESTING

User acceptance of the system is the key factor for the success of the system. The system under consideration is tested for user acceptance by constantly keeping in touch with prospective system at the time of developing and making change wherever required. This is done with regard to the following points:

- Output Screen design.
- Input Screen design.
- Menu driven system.

7.6 WHITE BOX TESTING

White box testing is a testing case design method that uses the control structure of the procedural design to derive the test cases. The entire independent path in a module is exercised at least once. All the logical decisions are exercised at least once. Executing all the loops at boundaries and within their operational bounds exercise internal data structure to ensure their validity. In our project testing was

Initially each module was tested separately to check whether they gave the desired output for the given input. The forms used to enter data by user were validated and appropriate error messages were displayed if incorrect data was entered. Once the data was entered correctly, the processing was done and testing was done to check whether the correct output was obtained. Once the test cases were conducted successfully for each module, the modules were integrated together as a single system. After integration, the test cases were again applied to check whether the entire system as a whole produced the desired output. At times, the test cases failed and the shortcomings were noted down and appropriate corrections were done. Once the integration testing was performed correctly, output testing was done and it did not result in any change or correction in the system. Black box testing and white box testing was also conducted successfully. All the loops, decisions, relations were executed at least once before giving it to the users for testing. In black box testing, it was checked whether the data in the proper format was stored in the database or not. Also, it was checked whether the interfaces were working properly or not. On successful completion of these tests, the system was then given to undergo user acceptance testing where the users entered test data to check whether the correct output was obtained. The users were satisfied with the output and thus the testing phase was completed successfully.

7.7 TEST DATA AND RESULTS

The primary goal of software implementation is the production of source code that is easy to read and understand. Clarification of source code helps in easier debugging, testing and modification. Source code clarification is enhanced by structural coding techniques, by good coding style, by appropriate supporting documents, by good internal comments and by the features provided in the modern programming language. In our implementation phase, source code contains both global and formal variables. It contains predefined functions as well as the user defined functions. The result of the new system is compared with old system and supposes if the result is wrong the error must be debugged. After the acceptance of the system by the user, the existing system should be replaced by this system. Any user handles this package very easily. It does not require any intensive training for the user. Procedures and functions involved in this system are very simple that anyone can understand and correspondingly act to the system with no difficulty.

TEST CASES

TEST CASE NO.	TEST CASE	EXPECTED OUTPUT	STATUS	DEFECTS
1.	USER LOGIN	USER SUCESSFULLY LOGGED IN	PASS	NIL
2.	LOAD AND PERFOM HAND TRACKING	WEB CAM OPENS AND DETECTS HANDS	PASS	NIL
3.	IMAGE SUGGESTION AND GENERATION	RECOMMENDS IMAGES AND GENERATE IMAGES	PASS	NIL

CHAPTER 8

RESULTS AND DISCUSSION

8.1 RESULTS

Some of the salient features of a virtual canvas using OpenCV with image suggestion project:

Hand Tracking and Gesture Recognition: Leveraging OpenCV, the virtual canvas project should include robust hand tracking and gesture recognition capabilities, allowing users to interact with the canvas using intuitive hand movements.

Real-time Image Suggestion: Implement an advanced image suggestion system that operates in real-time, analyzing user input and providing relevant image suggestions based on the ongoing drawing context.

Dynamic Canvas Interaction: Enable users to paint, draw, or insert images dynamically using hand gestures, eliminating the need for conventional input devices like a mouse or keyboard.

Adaptive Image Insertion: Develop a feature that allows the system to intelligently adapt suggested images to the user's drawing style, enhancing the coherence and creativity of the final composition.

User-Friendly Interface: Design an intuitive and user-friendly interface that accommodates users of varying skill levels, ensuring an accessible and enjoyable creative experience.

Offline Mode: Provide an offline mode to allow users to continue their creative process without the need for a constant internet connection.

Secure User Authentication (if applicable): If user accounts are involved, ensure secure authentication mechanisms to protect user data and maintain the privacy of individual projects.

8.2 SCREENSHOTS

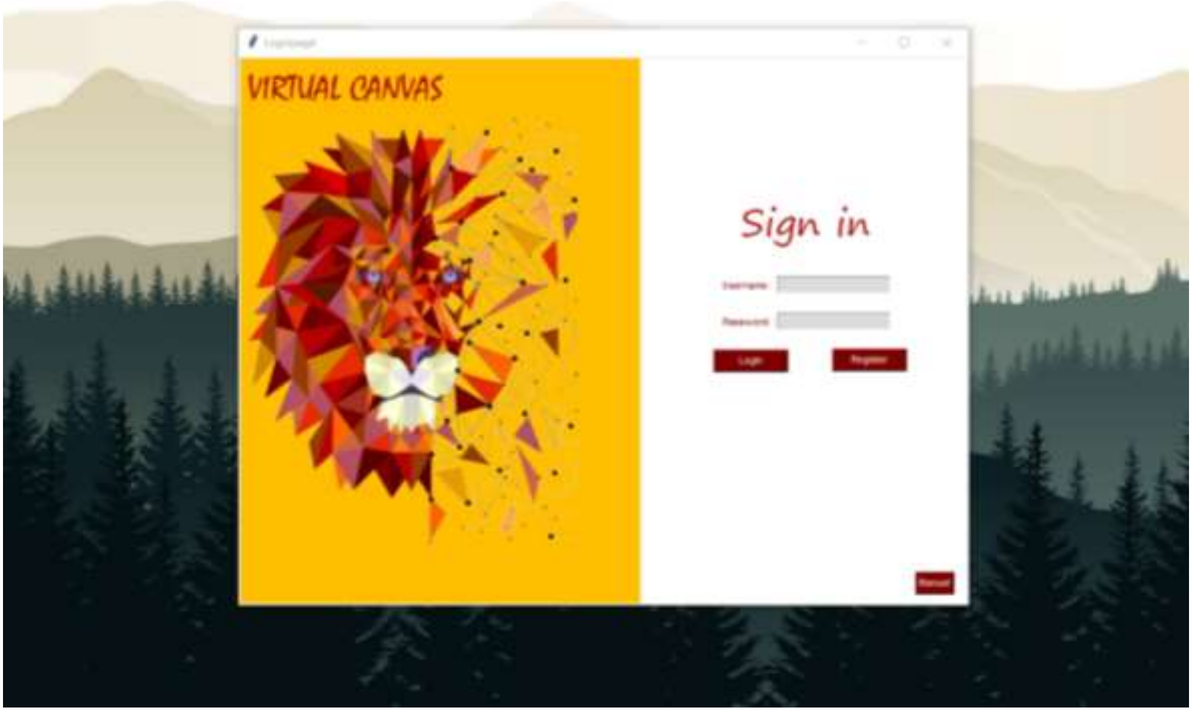


Image 1: Login page

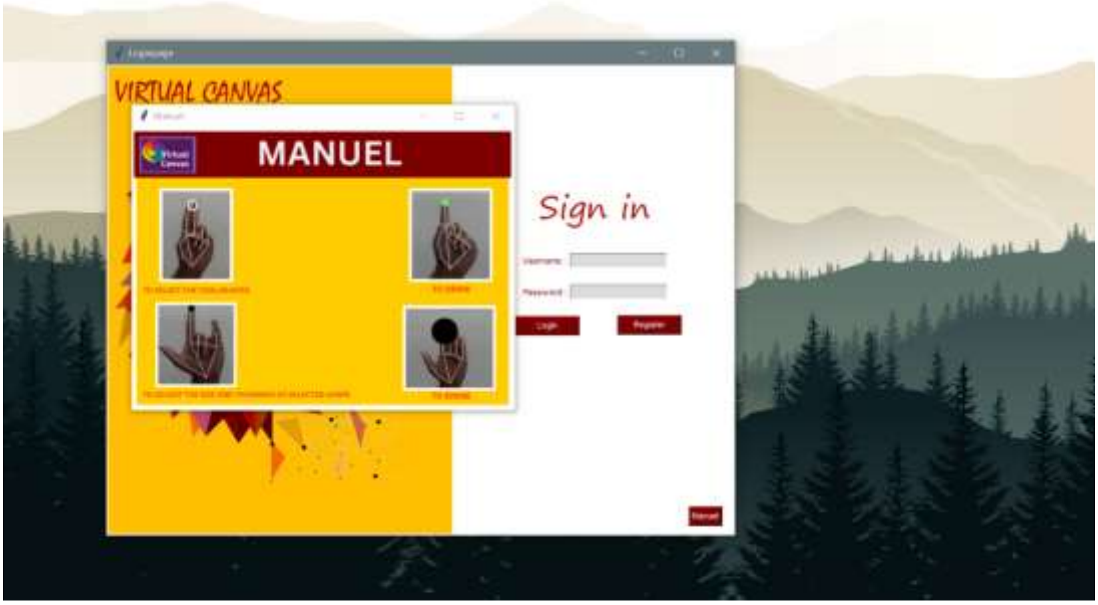


Image 2: User Manual

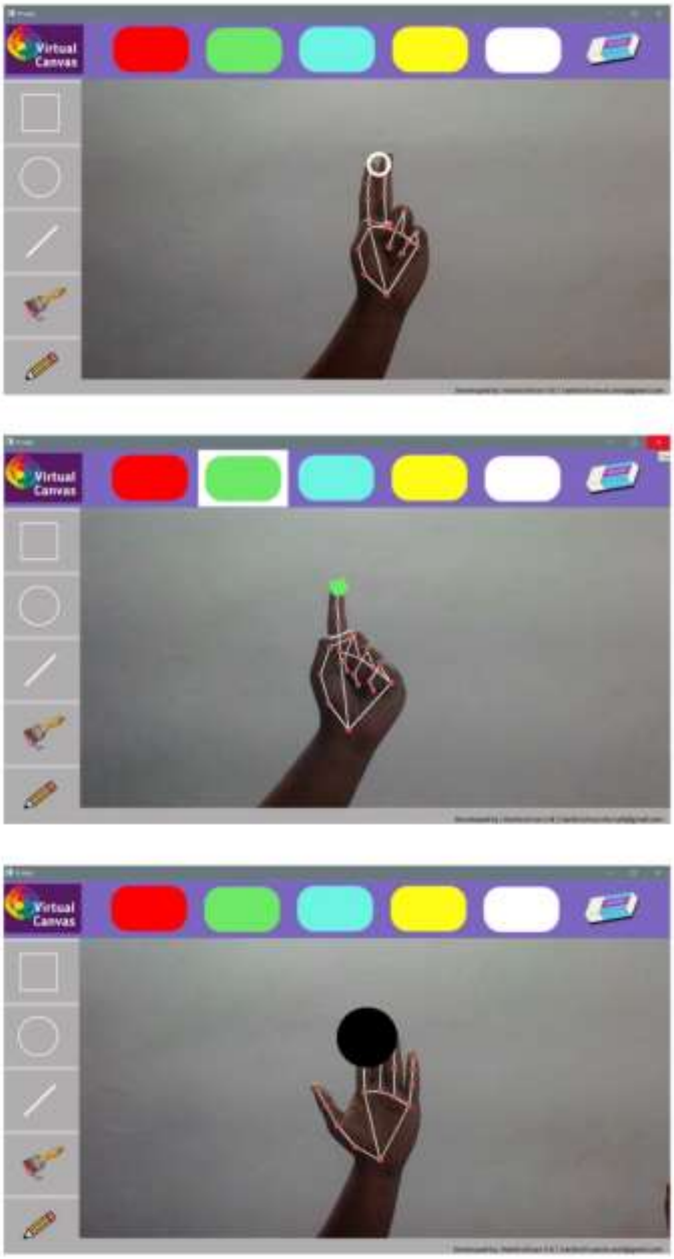


Image 2: Gesture Modes

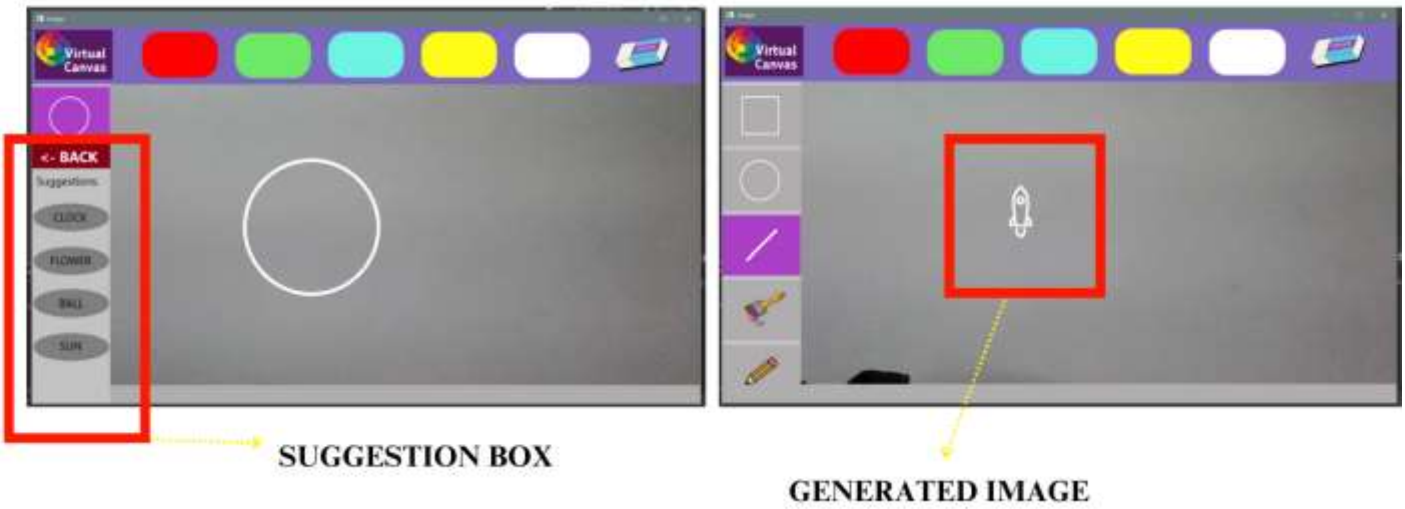


Image 3: Image Suggestion and Generation

CHAPTER 9

CONCLUSION

9.1 SYSTEM IMPLEMENTATION

Implementation is an activity that is contained throughout the development phase. It is the process of bringing a developed system into operational use and turning it over to the user. The new system and its components are to be tested in a structured and planned manner. A successful system should be delivered and users should have the confidence that the system should have work efficiently and effectively. The more complex system being implemented, the more will be the system analysis and design effort required just for implementation.

Implementation is the stage of the system when the theoretical design is turned into working system. The plan contains an overview of the system, a brief description of the major tasks involved in the implementation, the overall resources needed to support the implementation effort, and any site implementation requirements. The plan is developed during the design phase and is updated during the Development phase. The outline shows the implementation plan.

There are three types of implementations:

a) Implementation of a computer system for replacing the manual system. The problem encountered are converting files, training users, create accurate files.

b) Implementation of new computer system to replacing an existing one. This is usually a difficult conversion. If not properly planned, there can be many problems. Some larger computer systems have taken as long as a year to convert.

c) Implementation of modified application to replace an existing one using the same computer. This type of conversion relatively easy to handle, provided there are no major changes in file.

Implementation Plan Preparation

The implementation plan begins with preparing a plan for the implementation of the system. In this plan, discussion has been made regarding the equipment, resources and how to test the activities. The following information is acquired. What the task will accomplish?

- Resources required to accomplish the task.
- Key person(s) responsible for the task. Examples of the major tasks are the following.
- Providing overall planning and coordination for the implementation.
- Providing appropriate training for the personnel.
- Ensuring that all the manual are applicable to the implementation.
- Providing all the technical requirements.
- Performing site surveys before implementation.
- Ensuring that all the prerequisites have been fulfilled before the implementation date.
- Providing personnel for the implementation team.
- Acquiring special software or hardware.
- Performing data conversion before loading data into the system.
- Preparing site facilities for implementation.

The system implementation of a Virtual Canvas using OpenCV and Media pipe with image suggestion project can be divided into the following main stages:

Environment Setup: Install the necessary development tools, including OpenCV and any additional libraries required for image processing and gesture recognition.

User Interface Design: Design an intuitive user interface that includes the canvas, drawing tools, and any additional features such as image suggestion panes.

Hand Tracking and Gesture Recognition: Utilize OpenCV for hand tracking and gesture recognition. Implement algorithms that can track hand movements in real-time and recognize gestures to perform actions like drawing strokes, selecting tools, or triggering image suggestions.

Image Suggestion: Develop an intelligent image suggestion system that analyses the user's drawing context, possibly using Media pipe, to provide relevant image suggestions in real-time.

Canvas Interaction: Implement the core functionality for drawing on the virtual canvas using hand gestures. Ensure that the system recognizes different types of strokes, brush sizes, and colors.

User Authentication (if applicable): If the system involves user accounts, implement a secure authentication mechanism to protect user data and ensure privacy.

Testing: Conduct thorough testing of the system to identify and fix any bugs or issues. Pay particular attention to the accuracy of hand tracking, gesture recognition, and the performance of the image suggestion system.

Documentation: Provide comprehensive documentation, including user manuals and developer guides, to assist users and developers in understanding how to use and extend the virtual canvas system.

User Feedback and Iteration: Collect user feedback and iterate on the system based on user experiences. This could involve refining gesture recognition algorithms, improving the image suggestion model, or enhancing the overall user interface.

Deployment: Deploy the virtual canvas system to the intended platform, whether it's a desktop application, web application, or mobile app.

9.2 FUTURE ENHANCEMENT

Advanced Image Recognition and Context Awareness: Enhance the image suggestion system by incorporating more sophisticated machine learning models.

Consider deep learning techniques to improve image recognition accuracy and provide context-aware suggestions based on the entire composition.

AR and VR Integration: Explore integration with augmented reality (AR) and virtual reality (VR) technologies. This could offer users an immersive 3D canvas experience, allowing them to interact with their creations in a spatial environment.

Natural Language Interaction: Implement natural language processing (NLP) capabilities to enable users to verbally command actions, such as changing drawing tools, colors, or requesting specific image suggestions. This could enhance accessibility and user engagement.

AI-Driven Style Transfer: Integrate AI-driven style transfer algorithms that allow users to apply various artistic styles to their creations. This feature could enable users to experiment with different art movements or emulate the styles of famous artists.

Enhanced Collaboration Features: Strengthen collaborative features by introducing real-time audio and video communication between users. This could facilitate more interactive and synchronous collaboration, fostering a sense of shared artistic space.

Gesture Customization: Allow users to customize and define their own gestures for specific actions, providing a personalized and adaptable interface that aligns with individual preferences and workflows.

Cloud Integration for Storage and Collaboration: Implement cloud storage capabilities for seamless synchronization of projects across devices. This could also facilitate collaborative projects, enabling users to work on their creations from different devices.

AI-Powered Assistance for Complex Drawings: Introduce AI-powered assistance for intricate drawings, offering suggestions or automated enhancements for complex patterns, textures, or detailed elements. This could be particularly helpful for users with varying skill levels.

Dynamic Animation Features: Extend the virtual canvas into a dynamic animation platform, allowing users to create animated sequences by incorporating motion gestures and timeline-based controls.

Cross-Platform Accessibility: Ensure compatibility with a broader range of devices and platforms, including tablets, smartphones, and various operating systems, to expand accessibility for users.

User Community and Content Sharing: Implement features that encourage the formation of a user community, allowing artists to share their creations, collaborate on projects, and engage in discussions within the virtual canvas platform.

9.3 CONCLUSION

In conclusion, the development and implementation of the virtual canvas system utilizing OpenCV with advanced features like hand tracking and image suggestion mark a significant stride in redefining digital artistic expression. By seamlessly integrating cutting-edge technologies, this system offers users an innovative and intuitive platform to engage with their creativity. The incorporation of hand gestures for drawing and manipulating elements on the canvas not only breaks traditional input barriers but also enhances the user experience by providing a more natural and immersive interface.

The image suggestion system, driven by real-time analysis and intelligent algorithms, further elevates the creative process. Users benefit from contextual suggestions that adapt to their unique drawing styles, fostering a harmonious blend of human creativity and machine assistance. The collaborative drawing features extend the virtual canvas into a shared space, enabling simultaneous artistic expression and interaction among multiple users.

While the implementation showcases the potential of this virtual canvas, continual user feedback and iterative improvements will be crucial for refining and expanding its capabilities. User-friendly documentation ensures accessibility, and a robust testing phase ensures the system's reliability and performance across various scenarios.

In essence, this virtual canvas not only addresses challenges identified in traditional digital art interfaces but also opens new avenues for artistic collaboration, personalized expression, and a more natural and dynamic interaction between users and technology. As technology continues to advance, this project serves as a testament to the transformative power of integrating computer vision and machine learning in the realm of digital creativity.

CHAPTER 10

BIBLIOGRAPHY

10.1 BOOKS

- Learning OpenCV 5 Computer Vision with Python, Fourth Edition
- Using OpenCV with MediaPipe and Tensorflow to classify gestures an actions

10.2 WEBSITES

Google Scholar	:	https://scholar.google.com/
Google Mediapipe	:	https://developers.google.com/mediapipe
OpenCV	:	https://opencv.org/
Fluid Chart	:	https://www.lucidchart.com/pages/tour

10.3 REFERNCES

"Painting with Hand Gestures using Media Pipe", December-2022, International Journal of Innovative Science and Research Technology Publisher.

Reference Link: https://jiart.com/painting-with-hand-gestures-using-mediapipe

Ravi Kishore Veluri, S. Rama Sree, A. Vanathi, G. Aparna & S. Prasanth Vaidya

Aditya Engineering College, Suurampalem India, "Hand Gesture Mapping Using Media Pipe Algorithm", March-2022, Springer Publisher.

Reference link: https://link.springer.com/chapter/10.1007/978-981-16-8862-1

Amritanshu Kumar Singh, Vedant Arvind Kumbhare & K. Arthi School of Computing, SRM Institute of Science and Technology." Real-Time Human Pose

Detection and Recognition Using Media Pipe, February 2022", Springer Publisher.

Reference link: https://link.springer.com/chapter/10.1007/978-981-16-7088-6

ChatGPT

Reference link: https://chat.openai.com/e/582167fd-41fd-4625-8d3a-

APPENDICES

1. List of Figures

Figure No	Table Name	Page No
1	Process flow Diagram	28
2	Program Design	32

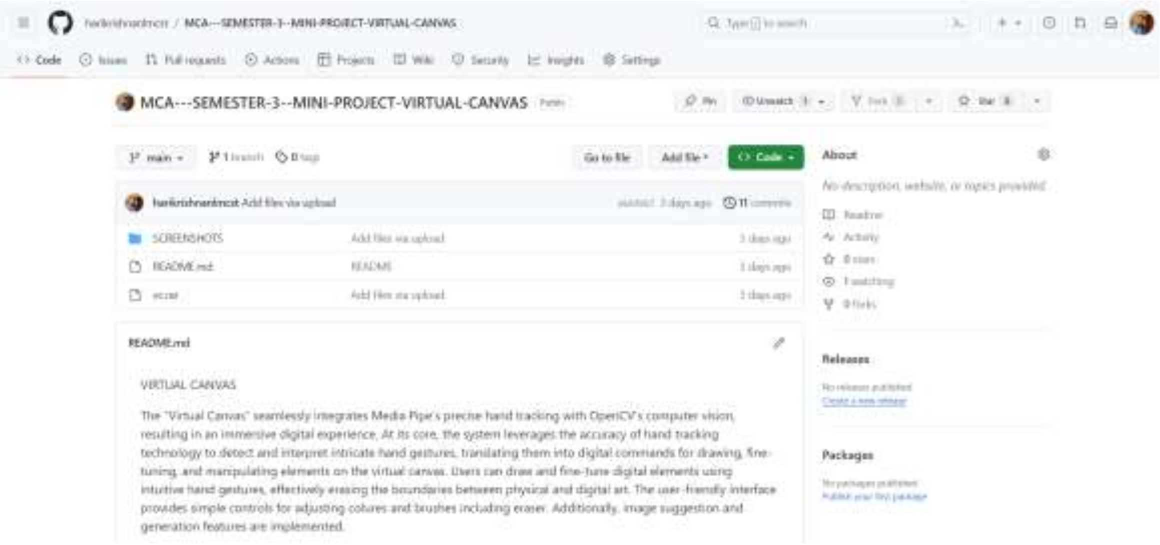
11 List of Images

Image No	Image Name	Page No
1	Login Page	47
2	User Manual	47
3	Modes	48
4	Image Suggestion and Generation	49

Abbreviation and Notation

OpenCV stands for Open-Source Computer Vision. To put it simply, it is a library used for image processing. In fact, it is a huge open-source library used for computer vision applications, in areas powered by Artificial Intelligence or Machine Learning algorithms, and for completing tasks that need image processing

GIT HISTORY



harikrishnanlmcst / MCA---SEMESTER-3--MINI-PROJECT-VIRTUAL-CANVAS

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

MCA---SEMESTER-3--MINI-PROJECT-VIRTUAL-CANVAS Public

main 1 branch 0 tags Go to file Add file Code

harikrishnanlmcst Add files via upload 3 days ago 11 commits

SCREENSHOTS	Add files via upload	3 days ago
README.md	README	3 days ago
vc.rar	Add files via upload	3 days ago

README.md

VIRTUAL CANVAS

The "Virtual Canvas" seamlessly integrates Media Pipe's precise hand tracking with OpenCV's computer vision, resulting in an immersive digital experience. At its core, the system leverages the accuracy of hand tracking technology to detect and interpret intricate hand gestures, translating them into digital commands for drawing, fine-tuning, and manipulating elements on the virtual canvas. Users can draw and fine-tune digital elements using intuitive hand gestures, effectively erasing the boundaries between physical and digital art. The user-friendly interface provides simple controls for adjusting colures and brushes including eraser. Additionally, image suggestion and generation features are implemented.

About

No description, website, or topics provided.

Readme

Activity

0 stars

1 watching

0 forks

Releases

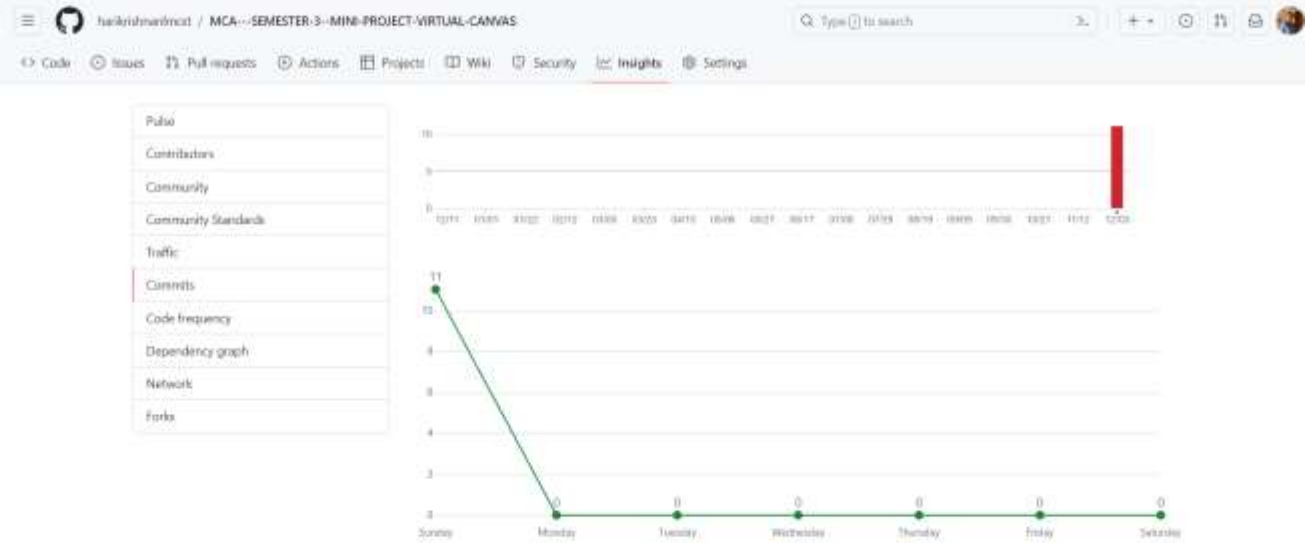
No releases published

Create a new release

Packages

No packages published

Publish your first package



MASTER OF COMPUTER APPLICATIONS (MCA)

LOURDES MATHA COLLEGE OF SCIENCE AND TECHNOLOGY

(MANAGED BY THE ARCHDIOCESE OF CHANGANASSERY)

KUTTICHAL, THIRUVANANTHAPURAM-695574

PHONE : 0472-2853550,2853682,2853546
WEBSITE : www.lmcst.ac.in