**CANVA 2.0**

**A PROJECT REPORT**

***Submitted by:***

**Hemang Jiwnani** (20BCE10528)

**Janhvi Rajyaguru** (20BCE10157)

**Siddhesh Khope** (20BCE10637)

**Kshitiz Prakash** (20BCE10068)

**Harsh Singh Kushwaha** (20BCE10826)

*in partial fulfillment for the award of the degree*

*Of*

**BACHELOR OF TECHNOLOGY**

*in*

**COMPUTER SCIENCE AND ENGINEERING**



**SCHOOL OF COMPUTING SCIENCE AND ENGINEERING**

**VIT BHOPAL UNIVERSITY**

**KOTHRI KALAN, SEHORE**

**MADHYA PRADESH - 466114**

DECEMBER 2021

**VIT BHOPAL UNIVERSITY, KOTHRI KALAN, SEHORE**

**MADHYA PRADESH – 466114**

**BONAFIDE CERTIFICATE**

Certified that this project report titled **“CANVA 2.0”** is the bonafide work of **“Hemang Jiwnani (20BCE10528), Janhvi Rajyaguru (20BCE10157), Siddhesh Khope (20BCE10637), Kshitiz Prakash (20BCE10068), and Harsh Singh Kushwaha (20BCE10826)** who carried out the project work under my supervision. Certified further that to the best of my knowledge the work reported at this time does not form part of any other project/research work based on which a degree or award was conferred on an earlier occasion on this or any other candidate.

**PROGRAM CHAIR PROJECT GUIDE**

Dr. Sandip Mal Dr. Ashish Kumar Sahu

Senior Assistant Professor Assistant Professor

School of Computer School of Computer Science and Engineering Science and Engineering

VIT BHOPAL UNIVERSITY VIT BHOPAL UNIVERSITY

The Project Exhibition I Examination is held on \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ACKNOWLEDGEMENT**

First and foremost I would like to thank the Lord Almighty for His presence and immense blessings throughout the project work.

I wish to express my heartfelt gratitude to **Dr. S. Poonkuntran**, Head of the Department, School of Computer Science and Engineering for much of his valuable support and encouragement in carrying out this work.

I would like to thank my internal guide **Dr. Ashish Kumar Sahu and** **Mrs. Dr. R. Thilagavathy**, for continually guiding and actively participating in my project, giving valuable suggestions to complete the project work.

I would like to thank all the technical and teaching staff of the School of Aeronautical Science, who extended directly or indirectly all support.

Last, but not least, I am deeply indebted to my parents who have been the greatest support while I worked day and night for the project to make it a success.

**LIST OF ABBREVIATIONS**

| **CV** | **Computer Vision** |
| --- | --- |
| **RAM** | **Random Access Memory** |
| **OS** | **Operating System** |
| **UI** | **User Interface** |
| **API** | **Application Programming Interface** |
| **ML** | **Machine Learning** |
| **RM** | **Requirements Management** |
| **CPU** | **Central Processing Unit** |
| **AI** | **Artificial Intelligence** |

**ABSTRACT**

The use of gesture as a natural interface serves as a motivating force for research studies in modeling, analyzing, and recognition of gestures. In particular, human-computer intelligent interaction needs vision-based gesture recognition, which involves many interdisciplinary studies. A survey on recent vision-based gesture recognition approaches is given in this paper. We shall review methods of static hand posture and temporal gesture recognition. Several application systems of gesture recognition are also described in this paper. We conclude with some thoughts about future research directions.

**Canva 2.0** is a graphic designing tool that is simple to use and wonderful for beginners and pros alike. Canva 2.0 is highly recommended for organizations and companies that want to try new and unique ways of designing library materials. Librarians will enjoy using this program and expressing themselves as they create library brands and accompanying materials. Drawing with the help of your fingertip is something that everyone wanted to do at some point in their life so this makes it highly appreciable for users.

**Canva 2.0** music is another important tool that can be used using Artificial Intelligence to detect facial expressions. Detecting facial features is a prerequisite to facial emotion recognition. One of the applications of this input can be for extracting the information to deduce the mood of an individual.

This data can then be used to get a list of songs that comply with the “mood" derived from the input provided earlier. This eliminates the time-consuming and tedious task of manually segregating or grouping songs into different lists and helps in generating an appropriate playlist based on an individual's emotional features. Facial Expression Based Music System parameters are provided. Thus our proposed system focuses on detecting human emotions for developing emotion-based music players, which are the approaches used by available music players to detect emotions, which approach our music player follows to detect human emotions, and how it is better to use our system for emotion detection. A brief idea about our systems working playlist generation and emotion classification is given This is achieved by observing the parts of the face like eves lOS movement etc. These are then classified and compared to trained sets of data. In this research, a human facial expression recognition system will be modeled using the eigenface approach. The proposed method will use the HAAR Cascade classifier to detect the face in an image. Fisher Faces calculation can be utilized for decreasing the high dimensionality of the eigenspace and after that anticipating the test picture upon the eigenface and computing the Euclidean separation between the test picture and meaning of the Eugene faces. The Grayscale image of the face is used by the system to classify basic emotions such as surprise, disgust, sadness, fear, neutral, anger, and happiness. Another important tool is Virtual Keyboard. It helps the designer write the text wherever he wants with a click of an option. Virtual Keyboard comes with 4 different virtual keyboards and supports all languages and keyboard layouts installed in the system.

**TABLE OF CONTENTS**

| **CHAPTER NO.** | **TITLE** | **PAGE NO.** |
| --- | --- | --- |
| **0** | * List of Abbreviations * Abstract | 4  5 |
| **1** | **PROJECT DESCRIPTION AND OUTLINE**   * 1.1 Introduction * 1.2 Motivation for the project * 1.3 Canva 2.0 and its techniques * 1.5 Problem Statement * 1.6 Objective of the project * 1.7 Organization of the project * 1.8 Summary | 11  11  11  11  12  12  12  13 |
| **2** | **RELATED WORK INVESTIGATION**   | * 2.1 Introduction * 2.2 Design & Art * 2.3 Existing Approaches/Methods 2.3.1 Dealing with Free-hand Drawing   2.3.2 Selection of music   * 2.4 Pros and Cons of the stated Approaches/Methods * 2.5 Issues/observations from investigation * 2.6 Organization of the project * 2.8 Summary | | --- | | 14  14  14  15  15  16  16  16  16  17 |
| **3** | **REQUIREMENT ARTIFACTS**   | * 3.1 Introduction * 3.2 Hardware and Software requirements * 3.3 Specific Project requirements * 3.4 Summary | | --- | | 18  18  18  19  19 |
| **4** | **DESIGN METHODOLOGY AND ITS NOVELTY**   | * 4.1 Methodology and goal * 4.2 Functional modules design and analysis * 4.3 Software Architectural designs * 4.4 Subsystem services * 4.5 User Interface designs * 4.6 Summary | | --- | | 20  20  20  21  22  23  23 |
| **5** | **TECHNICAL IMPLEMENTATION & ANALYSIS**   * 5.1 Outline * 5.2 Technical coding and code solutions * 5.3 Working Layout of Forms * 5.4 Test and validation * 5.5 Summary | 24  24  24  32  33  34 |
| **6** | **PROJECT OUTCOME AND APPLICABILITY**   * 6.1Outline * 6.2 Key implementations outlines of the project * 6.3 Significant project outcomes * 6.4 Project applicability on Real-world applications * 6.5 Inference | 36  36  36  36  37  37 |
| **7** | **CONCLUSION AND RECOMMENDATION**   * 7.1 Outline * 7.2 Limitation/Constraints of the Project * 7.3 Future Enhancements * 7.4 Inference | 38  38  38  39  39 |
|  | * **References** | 40 |

Chapter - 1: Project Description And Outline

**1.1 Introduction**

***"CANVA 2.0 - The New World of designers"***. Canva 2.0 will be the ultimate combination of Computer Vision and Artificial Intelligence. It will bring the future to the present. Techy things which we see in movies will now be the reality. It will let you draw on your screen with your hand and save it to your desired format. All developers need some music while working. Now no need to create a playlist, Canva 2.0 can recognize your mood and play music accordingly. ***Canva 2.0 coming soon!!***

**1.2 Motivation for the work**

Some Developers have tried creating screenwriting software, but it has never been used for designing and is an alternative to Canva.

People have tried creating music suggestions using AI/ML, but it was never integrated with Computer Vision.

**1.3 Canva 2.0 & it’s Techniques**

Our project is a simple python based project which has many amazing features which mainly include drawing with various colors smoothly, playing music, detecting one’s face, and saving the drawing in the JPEG format.

**1.4 Problem Statement**

“To revamp some features of the quaint version of Canva Pro into a more economical and digitally captivating version.”

**1.5 Objective of the work**

1. To intrigue the interest of young artists from pen-paper art to digital art.
2. To achieve the experience of drawing at your fingertips
3. To provide a high-end technology experience at a very low cost.
4. Aims to bring a huge update in existing technology which will help in changing the notions of people towards Canva.

**1.6 Organization of the project**

We made this project with the concrete motto in our minds, that is helping all the designers. We have divided the project into four parts and worked over it specifically.

1. ***Drawing with Ease***

We made the drawing easily as our project was able to detect the fingers and which helped us make the drawing with ease.

1. ***Color Detection***

The color selection and its opacity is a ridiculous task. The thickness of the pen while drawing and everything was set up in a way that helps designers to work with ease.

1. ***Music Player***

It works on face detection and this requires many implications. On detection, it places on according to the mood of the designer.

1. ***Virtual Keyboard***

It helps the designer write the text wherever he wants with a click of an option.

**1.7 Summary**

Our project clearly signifies its role in the current sector of the Technical World. We want to help designers complete their tasks with ease by helping them work with their fingertips.

Chapter - 2: Related Work Investigation

**2.1 Introduction**

The work was initiated with a rough sketch of our output screen, which is how exactly the project would look once completed. Then we began with the color selection and made a palette wherein all the colors were displayed. The code was made in a format where we could change the color once its opacity was different. We then made sure if the designer was able to select the colors and draw. After inspecting that, we found that the drawing wasn’t perfect. So after working on the code, we made the drawing smooth.

**2.2 Design & Art**

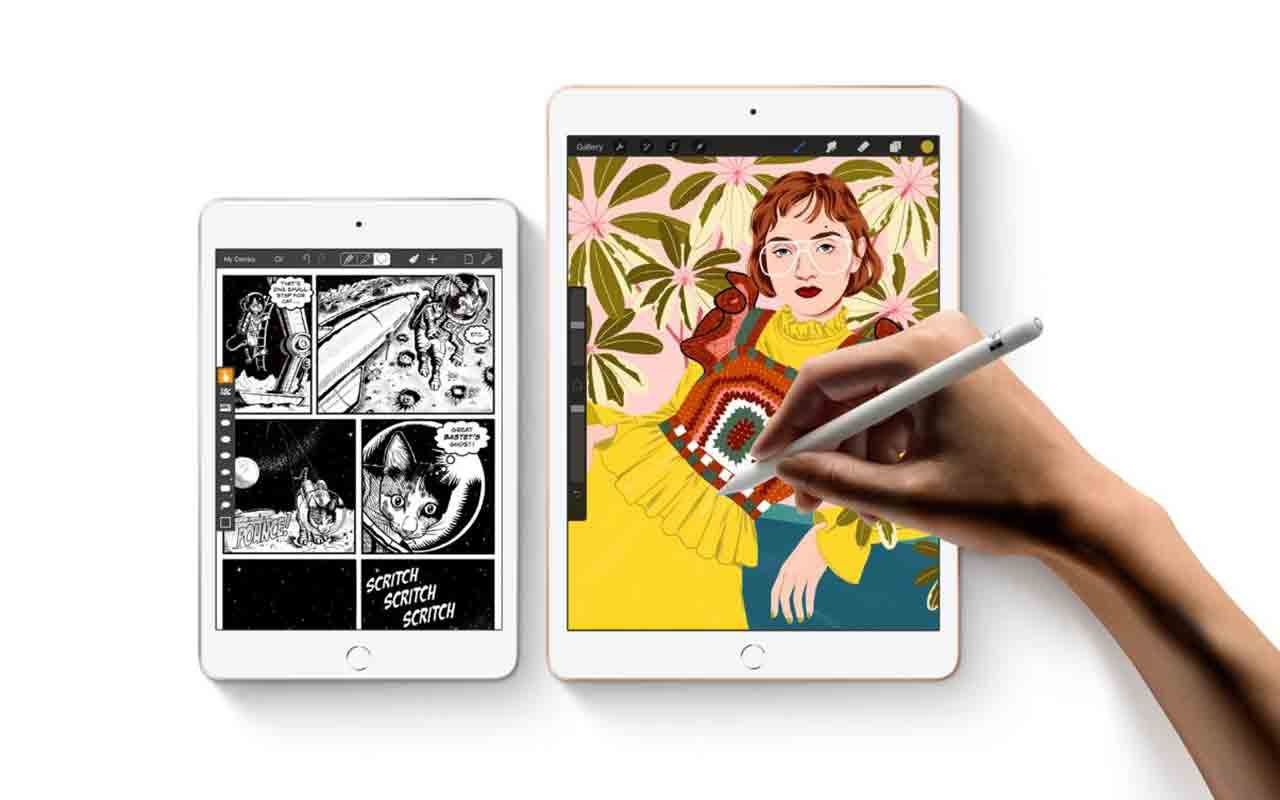
Today, the term design is generally used for what was formerly called the applied arts. The boundaries between art and design are blurred, largely due to a range of applications both for the term 'art' and the term 'design'. Applied arts can include industrial design, graphic design, fashion design, and decorative arts which traditionally include craft objects. In graphic arts (2D image making that ranges from photography to illustration), the distinction is often made between fine art and commercial art, based on the context within which the work is produced and how it is traded.

In engineering, design is a component of the process. "The application of scientific and mathematical principles to practical ends such as the design, manufacture, and operation of efficient and economical structures, machines, processes, and systems." ~ American Heritage Dictionary.

**2.3 Existing Approaches/Methods**

# *2.3.1 Dealing with Free-hand Drawing*

There are various tools available like a digital pencil/stylus, a mouse which can be used to draw or create designs.



# 

# *2.3.2 Selection of Music*

While working on projects designers require some kind of music so they can create a playlist in any particular app and listen as per their convenience.

# 2.4 Pros and Cons of the Approaches/Methods

***Pros-***

Method 1**-** Using a digital pencil or stylus makes annotations on something it's far away that you are searching at a breeze. Pencil does a whole lot more than letting you comic strip drawings or take down notes.

Method 2 **-** Listening to using while working helps to increase productivity. And a preset playlist is something that is an add-on for your concentration.

***Cons-***

Method 1**-** Many designers or young artists who want to shift their pen and paper art to digital artwork discover it too luxurious because of the great problems of low price range tabs and expense troubles with the iPad.

Method 2 **-** To create a playlist and play the music you are required to open another tab/app that uses more RAM resulting in faster draining of battery.

# 2.5 Issue/observation from the investigation

The observed issues are that many small-scale developers and designers cannot afford high-priced iPads or Digital Pencils. Even if they go for some cheaper ones that are available in the market they face issues that they lack in quality.

As far as music is concerned, a preset playlist is always good as it is created by yourself but in order to do that one needs to open an entirely different app than the one they are working in and that causes more battery usage and consumes more RAM.

# 2.6 Summary

Many designers or young artists who want to shift their pen and paper art to digital artwork discover it too luxurious because of the great problems of low price range tabs and expense troubles with the iPad.

The observed issues are that many small-scale developers and designers cannot afford high-priced iPads or digital pencils.

Users require music while working so Canva 2.0 helps them to play the songs according to their mood by creating the playlist based on their facial expressions.

Chapter - 3: Requirement Artifacts

**3.1 Introduction**

A requirement describes a condition or capability to which a system must conform. A requirement is either derived directly from user needs or stated in a contract, standard, specification, or another formally imposed document.

In the Requirements Management (RM) application, you use artifacts and artifact types to define requirements and support and enhance the definition of requirements.

For example, you can use features and use cases to describe the requirements, and then enhance the definition by creating diagrams, wireframes, or storyboards.

Artifact is a general term for an object in a repository. You can manage artifacts in projects and folders, or you can use tags to filter groups of artifacts for a specific purpose.

# 3.2 Hardware and Software requirements

## *Software:*

1. IDE: VS CODE, GOOGLE COLLABORATION
2. O.S: WINDOWS 7/8/10, MAC-OS (Optional)

## *Hardware:*

1. RAM:- Min 2GB
2. Graphic Card:- 2GB
3. Storage:- 120GB HDD/SSD
4. CPU:- Intel i3/i5, Ryzen 3,4(2000)

# 3.3 Specific Project requirements

3.3.1 - 8GB DDR4 RAM

3.3.2 - I3 11th Generation

3.3.3 - Webcam 2.0

3.3.4 - Audio Speakers

3.3.5 - 4GB Graphic

# 3.4 Summary

Some of the hardware and software required are VS CODE, GOOGLE COLLABORATION, RAM, and Graphic Card. But the minimum requirements must be satisfied in order to use the application.

Chapter - 4: Design Methodology And its Novelty

# 4.1 Methodology and goal

The research methodology is the description, explanation, and justification of various methods of research for effectively conducting this research. It can be understood as a science of conducting research scientifically. It studies various steps that are generally adopted by researchers while dealing with the research problems along with the logic behind them.

In research, we learned that many designers or young artists who want to shift their ***pen and paper art to digital art*** find it too expensive because of quality issues of low budget tabs and ***price issues*** with the iPad. Many of them belonging to small villages don't get a proper chance to outcast their skills. So for being a ray of hope for all of them we came up with the idea of ***Canva 2.0.***

# 4.2 Functional modules design and analysis

The module is based on hand detection and Face recognition for drawing and listening to music respectively

1. **Media pipe** - MediaPipe is a cross-platform framework for building multimodal applied machine learning pipelines. Media-Pipe is a framework for building multimodal (eg. video, audio, any time series data), cross-platform (i.e Android, iOS, web, edge devices) applied ML pipelines.
2. **OpenCV** - (Open Source Computer Vision Library) is an open-source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in commercial products.

Thismodule will be mainly responsible for hand detection. We will be using a media pipe for recognition of hand and marking coordinates on hand. We will be assigned to different tasks such as color choosing, saving art, opening the keyboard, closing the keyboard, typing, shifting music. cv2 for creating the frontend some functions like rectangle, circle for creating better UI.

# 4.3 Software Architecture designs

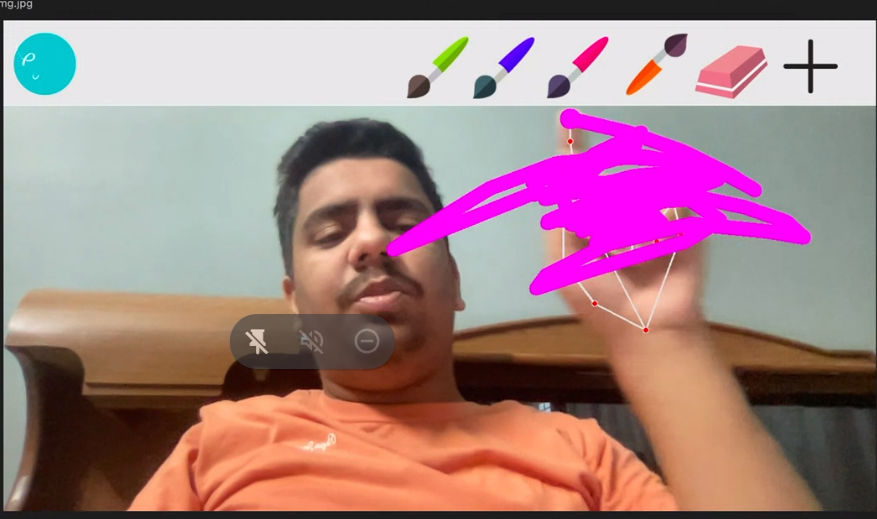
# 

# 4.4 Subsystem services

1. **Detection-** This part will mainly work for the detection of the hands and face of the user for future prediction of things. Once it detects the hands, the user can start drawing. After recognizing the face it will send it to the music player for further process.
2. **PDF Saver**-Once the designer has finished drawing his art. Just by clicking on this button, he will be able to put that art in a pdf. If time permitted us we will try to add a different format in which he will be saving his art ( For Example- .png, .jpef, .jpg)
3. **Colors & Keys-** Here we will connect all the colors possible to their default position with the help OS library. For starters only options of a few colors will be available and once eraser will be available for erasing.
4. **Music Player-**This will get a command from the source code after it detects the face and suggests a song. It will directly use the Spotify API for playing music. Other than that we will try to add music options. This feature will be only added if Time will permit us.

# 4.5 User Interface designs-

## UI PHOTO/SS



# 4.6 Summary - Currently there is no such tool in the market that is properly made for designing and teaching purposes. Most importantly, the integration of mood-based music players gives it an edge over other software. Computer vision makes it easier for individuals to work.

Chapter - 5: Technical Implementation & Analysis

# 5.1 Outline

The code includes many pythons and some machine learning libraries. Open-CV software is made better when users can easily contribute code and documentation to fix bugs and add features.

Once you start exploring with the help of computer vision Python plays the most important role and that's what our project is about. With the help of different classes and methods, we tried to explain our best, while working on computer vision we created new standards of using computer vision and making a matter of Designer software that the technical implementation of a different combination of different applications which are namely Media pipe, Open CV and all those you can add.

The output of this we are having an absolute software which can currently run on any Windows platform and give you are great out

# 5.2 Technical coding and code solutions

For reading and writing of the video just use and take the help of an open CV library which is directly interacting with the operating system software and system software and turning on your camera and recording it. Once the Video capturing started we are using the concept of overlaying the images. We are just using simple mathematical coordination and are overlaying some images and with the help of overlaying the images, we are just creating a user interface just like its software. Once the Software is created we are using the image Canvas with the help of Numpy to work over it. After creating the canvas we are also creating an ***Inverse Image*** so that we can cover it over there and while merging the ***Image Canvas*** and ***Image Inverse*** with the main screen with the help of bitwise OR and bitwise AND. Finally, the screen is overlaid over the main screen. So currently we are not drawing on the screen we are drawing over the screen.

Now technical coding and Solution- The first solution was this itself you cannot directly draw on your screen so we just created an overlaid screen. With the help of the overlaid screen \we started drawing the things and making them look more like software. Once the software is created we are using different functions, some like save, I am right, a function of an open CV which would be saving the image that is in the cv.rectangle for rectangles cv.circle to make some circles.

At last, we will be using a music player. It will be playing a major role in our code, which can be for turning on your camera and using the same coordinates that paint was using. As the paint starts using the coordinator it will pass on to the ***mood.py*** which is the mood detection library. Once it starts getting your images it will classify your image into sad, neutral, disgust, surprise, anger, fear, and happy. It detects it with the help of the simple conditions it will display the music accordingly which we have created. We are also giving users a chance to create their own playlist where they can add the songs into the folder named as that \_Song folder.

**##Important Libraries**

**import mediapipe ##Handetection**

**import cv2**

**import time**

**import os**

**import numpy as np**

**import Module as htm**

**#####################**

**################################**

**brushThickness = 20**

**eraserThickness = 50**

**imgInv = 0**

**xp =0**

**yp=0**

**###############################**

**#folderPath = 'header'**

**folderPath = 'header'**

**myList = os.listdir(folderPath)**

**print(myList) ##Comment out**

**overlayList = []**

**def BrushThicknesschanger():**

**brushnew = detector.ThicknessChanger(img,brushThickness,lmList)**

**return brushnew**

**# myList = ['5.png','4.png','3.png','2.png','1.png']**

**for imPath in myList:**

**image = cv2.imread(f'{folderPath}/{imPath}') ## (f'header/5.png)**

**overlayList.append(image)**

**print(overlayList) ##Comment out**

**header = overlayList[0]**

**draColor=(255,0,255) ##Sense**

**cap = cv2.VideoCapture(0) ##video**

**cap.set(3,1280)**

**cap.set(4,720)**

**#import HandTrackingModule**

**detector = htm.handDetector(detectionCon = 0.7) ########**

**imagecanvas = np.zeros([720,1280,3],np.uint8)**

**while True:**

**#1. Import image**

**success, img = cap.read()**

**print(success,img)**

**img = cv2.flip(img,1)**

**#2. Find Hand landmarks**

**img = detector.findHands(img)**

**lmList = detector.findPosition(img,draw=False)**

**if len(lmList)!=0:**

**# print(lmList)**

**#Tip of index and midle finger**

**x1,y1 = lmList[8][1:]**

**x2,y2 = lmList[12][1:]**

**#3. Check which fingers are up**

**fingers = detector.FingersUp()**

**# print(fingers)**

**#4. if selection mode - Two fingers up**

**if fingers[1] and fingers[2]: ##True AND TRUE = TRUE**

**xp,yp = 0,0**

**# print("Selection mode")**

**##Checking for the click**

**if y1 < 125:**

**if 550<x1<650:**

**header = overlayList[0]**

**draColor = (0,128,0)**

**elif 670 <x1<770:**

**header= overlayList[1]**

**draColor = (0,0,255)**

**elif 790 <x1<890:**

**header= overlayList[2]**

**draColor = (255,255,0)**

**elif 890 <x1<990:**

**header= overlayList[3]**

**draColor = (128,0,128)**

**elif 990 <x1<1090:**

**header= overlayList[3]**

**draColor = (0,0,0)**

**elif 1100 <x1<1200:**

**#Sx1 = 150**

**brushnew = BrushThicknesschanger()**

**brushThickness = brushnew**

**#header = overlayList[4]**

**cv2.rectangle(img,(x1,y1-25),(x2,y2+25),draColor,cv2.FILLED)**

**#5. if Drawing mode - Index fingers up**

**if fingers[1] and fingers[2] == False: ## True and False == False [False == False]**

**# xp,yp = 0,0 #check for this line**

**cv2.circle(img,(x1,y1),15,draColor,cv2.FILLED)**

**# print("Drawing Mode")**

**if xp==0 and yp==0:**

**xp,yp=x1,y1**

**if draColor == (0,0,0):**

**cv2.line(img,(xp,yp),(x1,y1),draColor,eraserThickness)**

**cv2.line(imagecanvas,(xp,yp),(x1,y1),draColor,eraserThickness)**

**else:**

**cv2.line(img,(xp,yp),(x1,y1),draColor,brushThickness)**

**cv2.line(imagecanvas,(xp,yp),(x1,y1),draColor,brushThickness)**

**xp,yp = x1,y1**

**imgGray = cv2.cvtColor(imagecanvas,cv2.COLOR\_BGR2GRAY)**

**\_ , imgInv = cv2.threshold(imgGray,50,255,cv2.THRESH\_BINARY\_INV)**

**imgInv = cv2.cvtColor(imgInv,cv2.COLOR\_GRAY2BGR)**

**#print(img)**

**#print(imgInv)**

**img = cv2.bitwise\_and(img,imgInv)**

**img= cv2.bitwise\_or(img,imagecanvas)**

**# 1. Setting the Navigation bar**

**img[0:125,0:1280] = header**

**#img = cv2.addWeighted(img,0.5,imagecanvas,0.5,0)**

**#print(imgInv)**

**cv2.imshow("Image",img)**

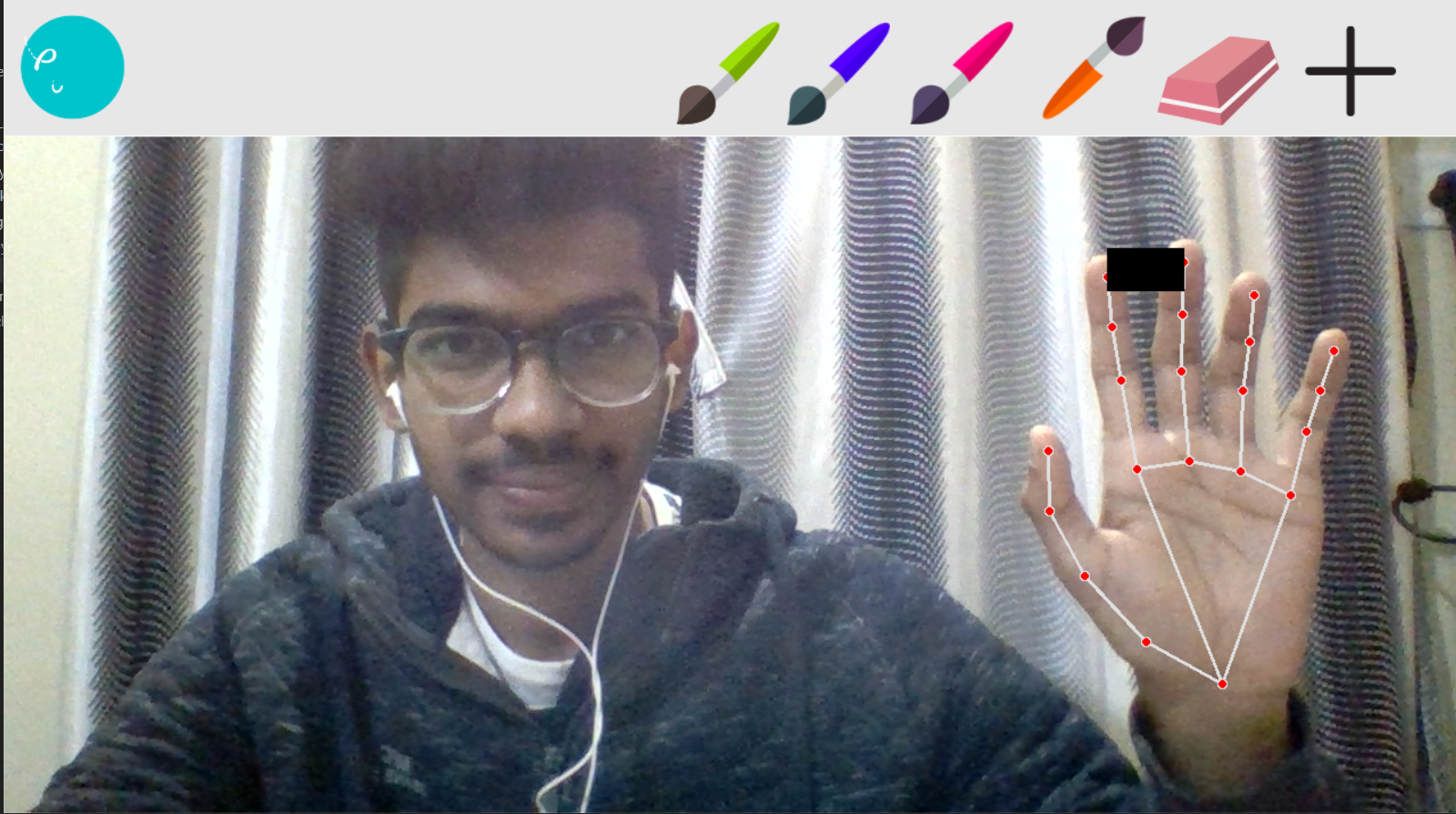
**cv2.imshow("canvas",imagecanvas)**

**cv2.imshow("Inv",imgInv)**

**cv2.waitKey(1)**

# 

# 5.3 Working Layout of Forms

****

# 5.4 Test and validation

When the software was just created we checked it on running on different windows including Windows Explorer, Windows 7, Windows 8, 10, and 11. And the software was successful in running all of those platforms. Other than that the contradiction with the software was facing was the new Apple-powered M1 chip was having architecture and hence not able to successfully install the TensorFlow library. So it wouldn’t be supporting our code till Apple finds a solution for the TensorFlow library. So that is a restriction till now and it's a test which we provide.

Other than that while changing the color and all those features we tried to keep it properly. We just tried implementing each of the colors and making some figures on our own to make it in such a way that it would be able to write a name on its own and make some signatures. All of the colors were working in the described distance.-

##Checking for the click

if y1 < 125:

if 550<x1<650:

header = overlayList[0]

draColor = (0,128,0)

elif 670 <x1<770:

header= overlayList[1]

draColor = (0,0,255)

elif 790 <x1<890:

header= overlayList[2]

draColor = (255,255,0)

elif 890 <x1<990:

header= overlayList[3]

draColor = (128,0,128)

elif 990 <x1<1090:

header= overlayList[3]

draColor = (0,0,0)

elif 1100 <x1<1200:

And it was working properly and also the eraser concept was also working properly. And at last, by closing the code was properly saving other than that the concept of the music play was also working completely fine, the user just needed to change the path of the music player.

# 5.5 Summary

The basic idea used was to create one Image inverse and one canvas screen and combining those two gives our output screen. We just tried implementing each of the colors and making some figures on our own. Some functions like save, I am right, a function of open CV is used. And for the music player mod.py library is used to recognize your mood and play songs as per one's choice. The code has been under-processed with many tests and validations. Proper care has been taken to solve any kind of issues with the bugs in the code. Each and every part of the code is considered in order to make it user-friendly.

Chapter - 6: Project Outcome And Applicability

# 

# 6.1 Outline

Drawing the attention from pen and paper art towards digital art by providing them the experience of drawing at their fingertips.

Later aims to bring the huge update in existing technology by providing a high-end technology experience at a very low cost which will change notions of users towards Canva.

# 6.2 Key implementations outlines of the System

We are focusing to imprint a new world of Art and Craft which not only focuses on the drawing but also gives an artist many other astounding options to think and make himself comfortable while drawing. The music is the best partner in crime while working with something creative and that motivated us to add to our project.

# 6.3 Significant project outcomes

The most significant and crucial outcome of this project is that it helps all the designers perform their work with ease. We tried our best to assure that we succeed in fulfilling our motto of reaching out to the maximum designers and helping them in the best way.

# 6.4 Project applicability on Real-world applications

1. ***COMMERCIAL USE:***

* It can be an alternative for the blackboard in the education industry as it is very cheap and also a tough competition to the smart class as well. Many government schools that don't have enough funding, can be a savior for them.
* Many companies can use it to make their company products and show them to different investors for funding.

1. ***PERSONAL USE:***

* It can help young designers to make visual arts without the need for a tab(iPad), digital pen, etc.
* It can help in day-to-day use, for example:- to write a to-do list digitally.
* It can engage small kids to scribble on the wall without actually spoiling the wall.

# 6.4 Inference

After working over the code, we conclude that our project is ready to implement but we still have many more features to add and execute. We believe we have done a satisfactory task and completed the project successfully.

CHAPTER 7 -CONCLUSION AND RECOMMENDATION

# 7.1 Outline

Canva 2.0 is a comprehensive design tool that is simple to use and wonderful for experts and beginners alike. One of the most useful features includes the use of your fingertips to draw anything on the screen and which is available at a very low price. These features make the program user-friendly and easy to use. It is an online platform that allows users to access many features like digital drawing, music player, virtual keyboard in a single app or website. The program is intuitive. However, time is needed to learn the program’s unique features. Canva 2.0 has had its problems, including operating systems of both Mac and Windows i.e for Mac it is not working. The basic program has many selections of color combinations and is a good value for those libraries that do not want to make a larger investment. Canva 2.0 is highly recommended for organizations that want to try new and unique ways of designing library materials. Librarians will enjoy using this program and expressing themselves as they create library brands and accompanying materials.

# 7.2 Limitation/Constraints of System

1. In our system, we tried adding more colors but unfortunately, the program is becoming complex, therefore, we have been limited with our number of colors.
2. Drawing cannot be made with both hands. As another hand appears in the window the previous hand gets kicked out.
3. Also, the code is supported in windows as of now. The iOS system is not running the entire implication of the code.

# 7.3 Future Enhancements

1 We are working continuously to add more colors to the palette so that a designer can add more of them to their designs.

2 Code is under constant deep learning in order to understand and implement the part of the simultaneous working of music as well as drawing.

3. We will try adding the virtual keyboard.The keyboardwill work as just after clicking this the signal will be sent to python source code. Just after receiving a signal, a separate code for Virtual Keyboard will start and a person can type using fingertips.

# 7.4 Inference

Canva is a graphic design tool that lets users design flyers on the fly. Canva 2.0 is itself a very big update in itself. It provides features such as drawing using fingertips and listening to music using face recognition. It also aims to bring in some future enhancements like the one mentioned earlier, the virtual keyboard.

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# 

# References:-

1. <https://link.springer.com/chapter/10.1007/3-540-46616-9_10?utm_source=canva&utm_medium=iframely>

1. <https://www.python.org/doc/?utm_source=canva&utm_medium=iframely>
2. <https://opencv.org>
3. <https://pypi.org>
4. <https://www.fastcompany.com/40514362/graphic-design-startup-canva-just-turned-into-a-unicorn>