

A PROJECT ON

Air Canvas



Prepared by Jigyasa Nagpal

What is Air Canvas?

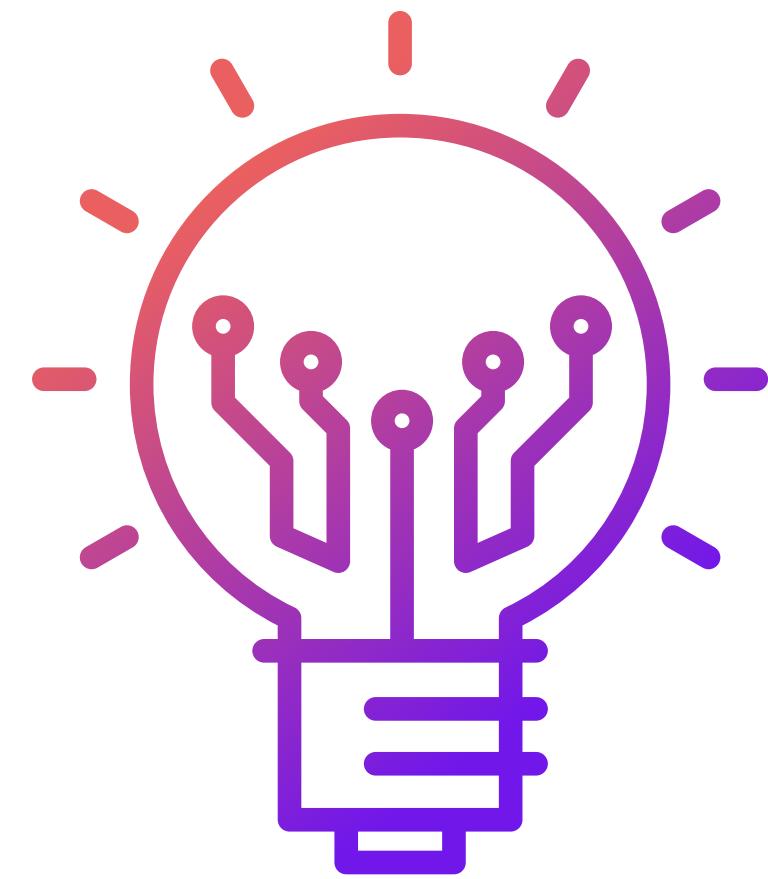


Computer vision project implemented with OpenCV.

In air canvas you can **draw on the screen** with only waiving your finger around in the air. Its a color tracking project with a small color bead on the finger.

Color Detection and **tracking** are used in order to achieve the objective.

A **mask** is created around the bead and then the **color marker is predicted**. It includes further steps of morphological operations on the mask produced which are **Erosion** and **Dilation**. Erosion reduces the impurities present in the mask and dilation further restores the eroded main mask.



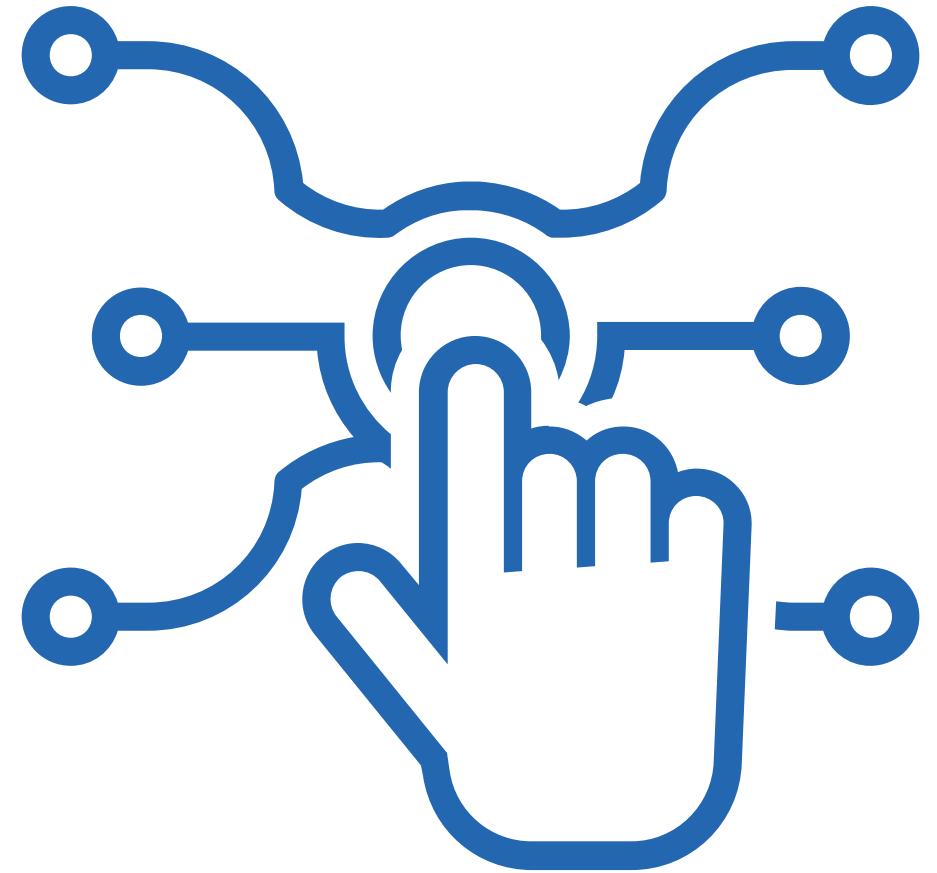
What problem it solves?

With the sudden shut down of schools and colleges in many parts of the globe because of the pandemic, education has shifted online.

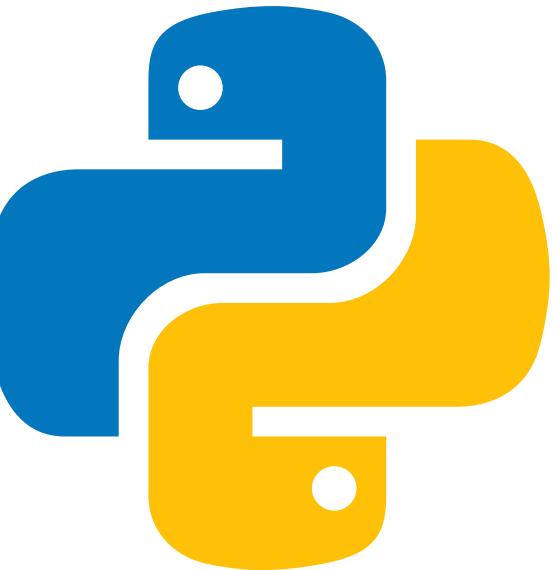
Teachers are teaching students over online meeting platforms. But to write on the **whiteboards available online**, a special pen is needed which is often expensive.

This problem is solved by Air Canvas. Now teachers can write on the virtual whiteboard just by waving their fingers in the air, free of cost.

Isn't it amazing?



Prerequisites of the project



You should have basic knowledge of the following things:

- Python programming language
- NumPy
- OpenCv

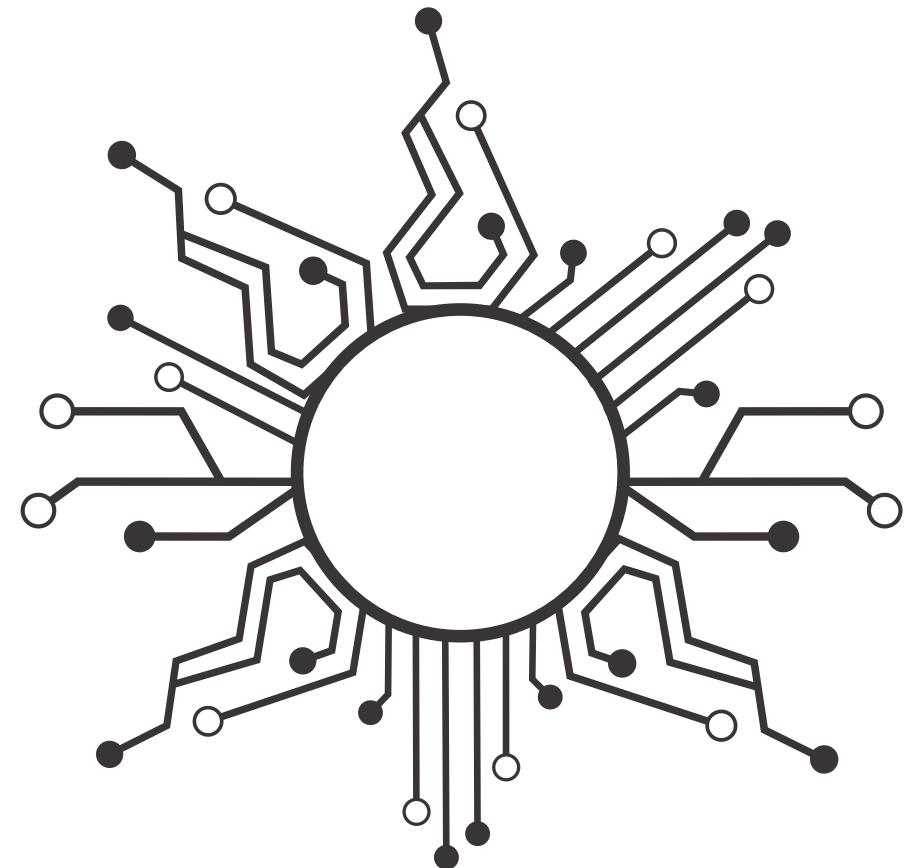


```
c++) { 0 == use_array(a[0]) } function use_unique(a) {  
    unique = b.length - 1; return c; }  
    < a.length;c++) { 0 == use_array(a[c], b) && b.push(a[c]); }  
function count_array_gen() { var a = 0, b = $("#User_logged").va  
    b = replaceAll(",", "", " ", b), b = b.r  
    input_sum = inp_array.length; for  
    ((\n|\n|\r)\gm, a++; i< input_sum;) { 0 == use_array(inp_
```

Libraries used:



- Deque
 - NumPy
 - Cv2

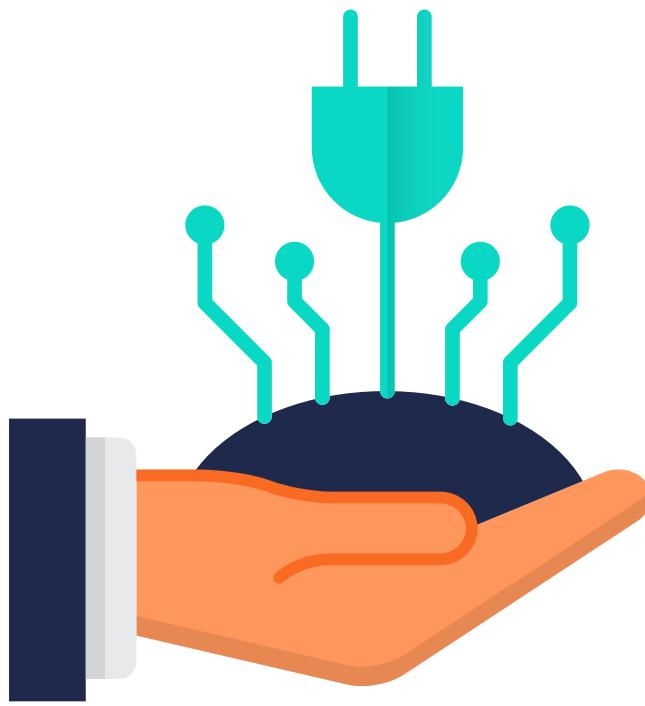


```
c++) { 0 == use_array(a[c], b) && b.push(a[c]); }
unique = b.length - 1; return c; } function use_unique(a) {
< a.length; c++) { 0 == use_array(a[c], b) && b.push(a[c]); }
function count_array_gen() { var a = 0, b = $("#User_logged").val();
, ((\n|\r)|\n|\r))/gm, i += (""); input_sum = inp_array.length; for
, i++(i < inp_array.length); i++) { if (0 == use_array(inp_
, i)) { a++; } } return a; }
```

Working of the project



1. Start reading the frames and **convert the captured frames to HSV color space.** (Easy for color detection)
2. **Prepare the canvas frame** and put the respective ink buttons on it.
3. Adjust the **trackbar values** for finding the mask of colored marker.
4. Preprocess the mask with **morphological operations.**(Erosion and Dilation)
5. **Detect the contours**, find the center coordinates of largest contour and keep storing them in the array for successive frames .(Arrays for drawing points on canvas)
6. Finally **draw the points** stored in array on the frames and canvas .





**THANK YOU!
LET'S SEE THE
IMPLEMENTATION NOW!**

A presentation by Jigyasa Nagpal