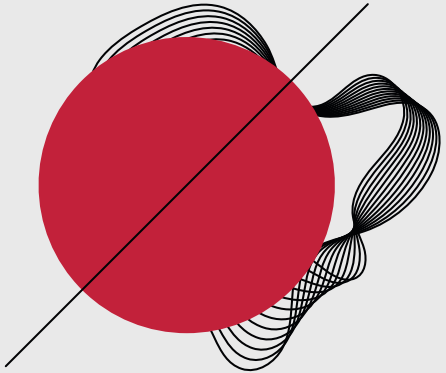
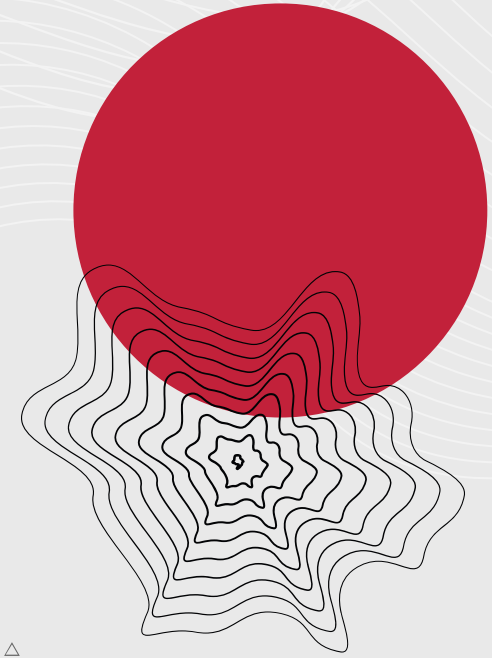
Graduation Project



Computer Vision Verification of Sample Self-Acquisition for Self Diagnostics Using Deep Learning

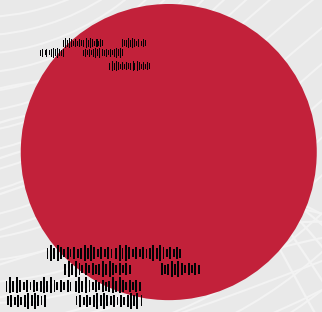


Under Supervision of
Prof. Dr.
Mahmoud Ibrahim Khalil

By

Ahmed Sameh Shahin	16P6063
Karim Walid Abdelazim	16P3090
Moataz Khaled Zakaria	16P8244
Mohammed Ehab Elsaeed	16P8160

AGENDA

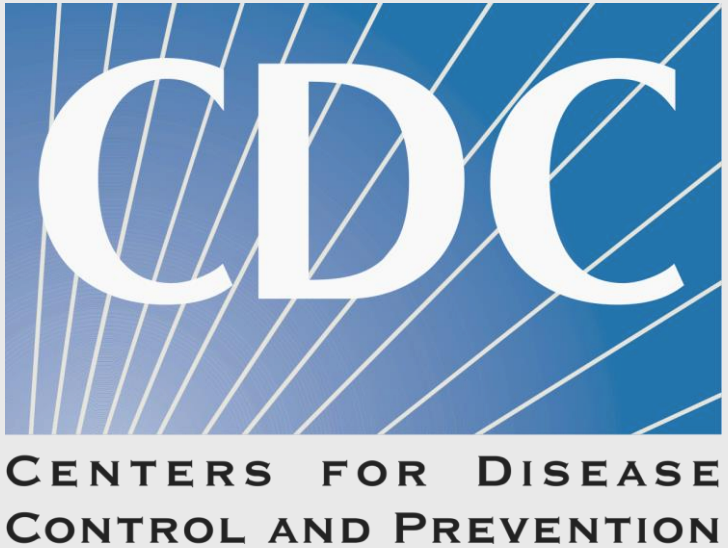


1. Introduction
2. Problem in hand
3. Approach Used
4. Problems Faced and Solutions
5. Final Result and Demos
6. Future Enhancements

01.

Introduction

Self-Testing Attempts



Xtrava's SPERA™

02.

Problem in Hand

Scope

Benefits:

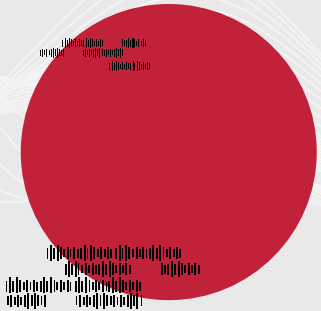
- Reduced costs
- Lower loads on hospitals and clinics
- Accuracy
- Ease-of-use
- Minimization of errors in sample collection and preparation .

Objectives:

- Detect users' movements.
- Detect test tools (swab)
- Monitor users' actions while performing the test.

Goals:

- Validate and verify that users follow the given steps during self-testing.



Problem Statement

An application to verify that a user followed the steps instructed through the application correctly.

More specifically, that the user inserted the provided swab in both the right and left nostrils without the need for a medical professional then used the provided solution with the acquired sample.

Functional Requirements

- Detect Aruco markers and their IDs.
- Detect and localize whether a swab is in the Left nostril or the right nostril or in neither.

Non-Functional Requirements

- Realtime, up to 1s delay in response.
- User-friendly interface.
- Reliable.
- Works on Android/iOS as a web/mobile App.

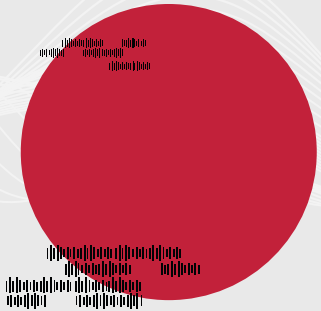
Assumptions and Dependencies

Assumptions/Constraints:

- User is not wearing any kind of eye-wear
- User is standing in front of a blank white(ish) background.
- User is wearing a black shirt
- User using only the provided tools, marked with the Aruco markers.

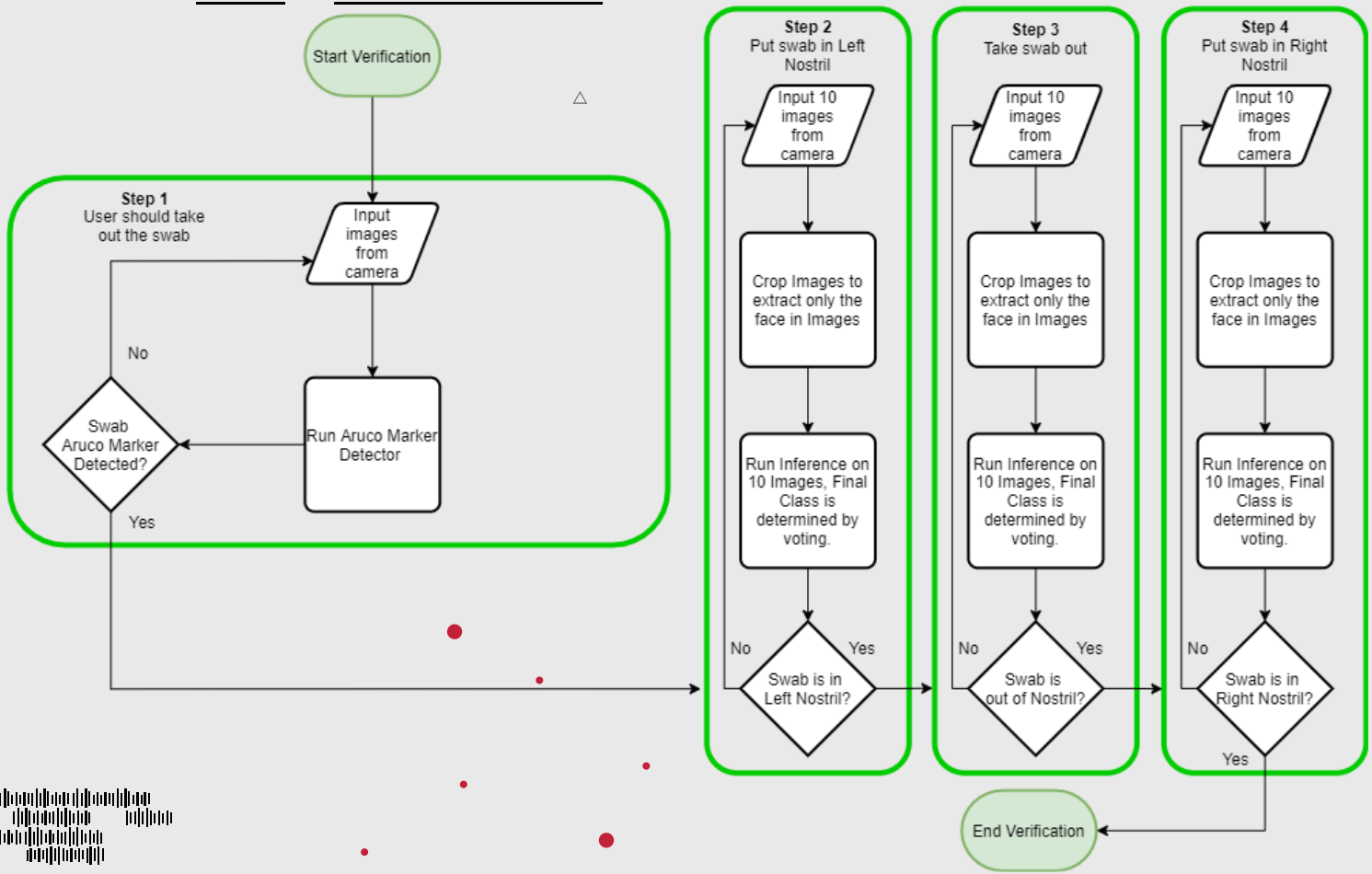
Dependencies:

- Low resolution or faulty camera
- Poor camera angles
- Poor lighting
- High exposure
- Markers are out of the camera's line of sight.

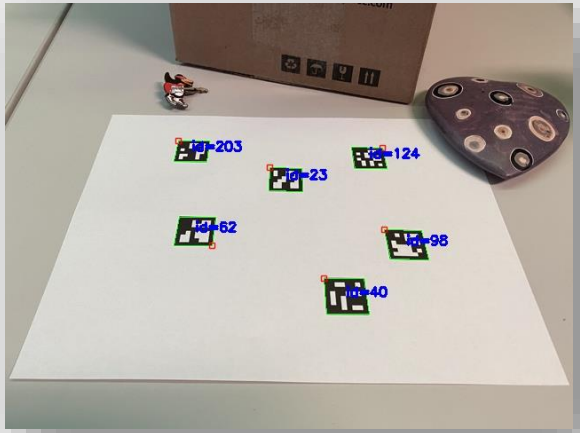


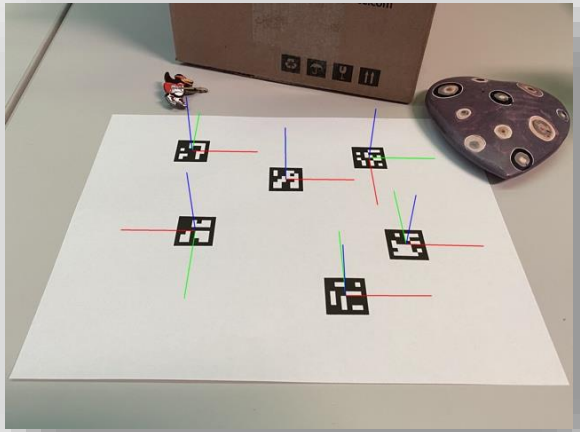
03.

Approach Used


Start Verification
Step 1
User should take out the swab
Input images from camera
Run Aruco Marker Detector
Swab Aruco Marker Detected?
No
Yes
Step 2
Put swab in Left Nostril
Input 10 images from camera
Crop Images to extract only the face in Images
Run Inference on 10 Images, Final Class is determined by voting.
Swab is in Left Nostril?
No
Yes
Step 3
Take swab out
Input 10 images from camera
Crop Images to extract only the face in Images
Run Inference on 10 Images, Final Class is determined by voting.
Swab is out of Nostril?
No
Yes
Step 4
Put swab in Right Nostril
Input 10 images from camera
Crop Images to extract only the face in Images
Run Inference on 10 Images, Final Class is determined by voting.
Swab is in Right Nostril?
No
Yes
End Verification

Visual Marker Detection



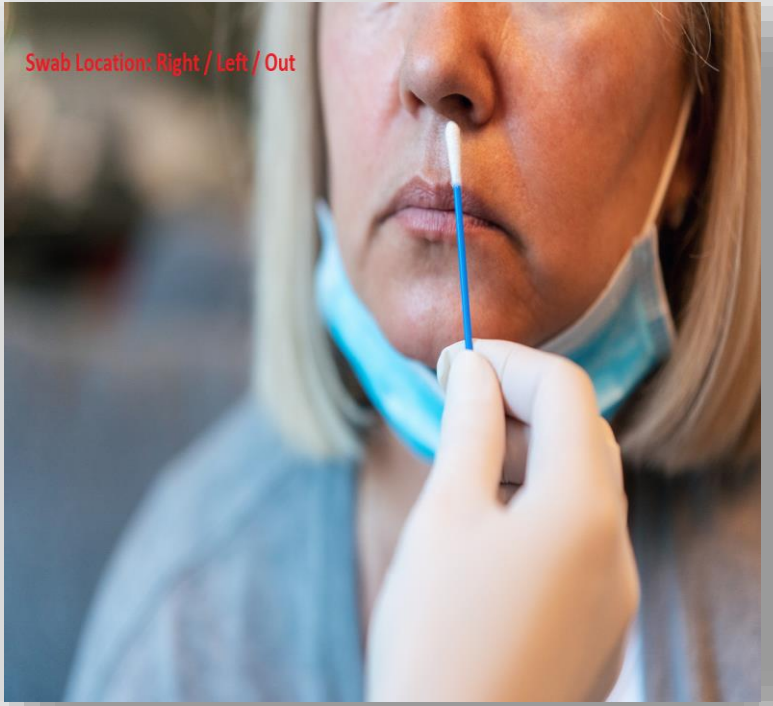


VISUAL MARKER DETECTION

OPEN CV

The OpenCV Library has built-in support for ArUco markers detection and generation.

Swab
LOCATION CLASSIFIER



SWAB LOCATION CLASSIFIER

Softmax Layer with 3 classes
Left – Right – False

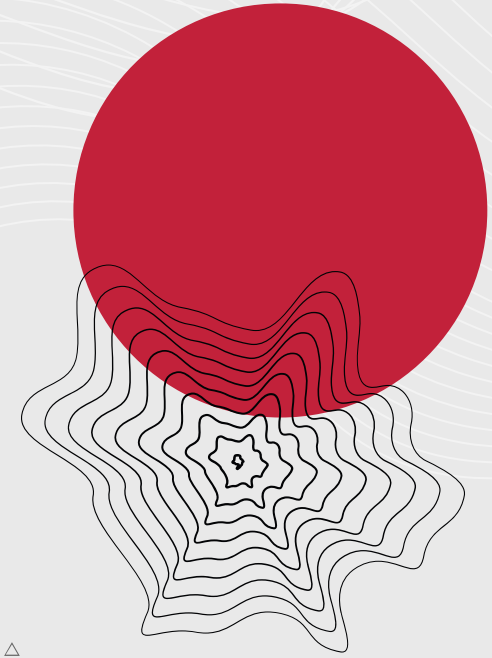
Transfer Learning of a **RESNET 50 DNN** Pretrained on the Imagenet Dataset with its top replaced by a Flattening Layer Then a 3-Unit Softmax Layer

Why RESNET 50?

- **Pretrained Image Feature Extractor**
- **Solves the Vanishing Gradient Problem of Deep Neural Networks Using skip connections**
- **An Imagenet pretrained version is available in Keras**

04.

Problems Faced and Solutions

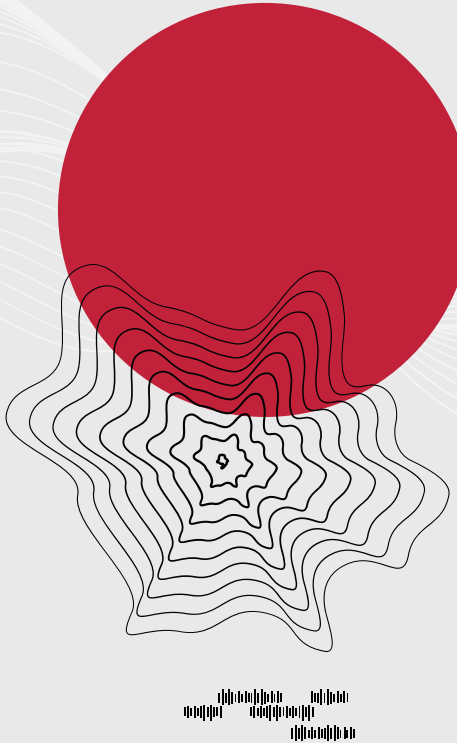


Problem 1: High Variance & Low Bias (Overfitting)

Set	Average Accuracy
Training	99%
Validation	60%

Solution Trial

Tried changing the softmax layer to be compatible with our desired outputs (3 classes)

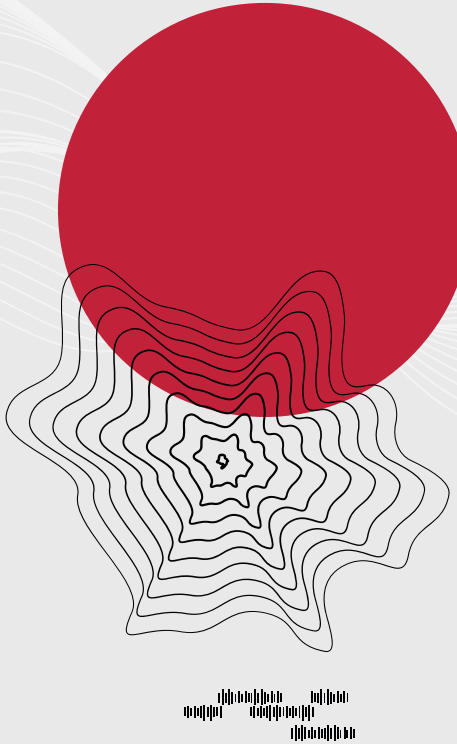


Result

Model still overfitted.

Solution Trial

Tried adding a dropout of value 0.7-0.9 before the softmax layer.

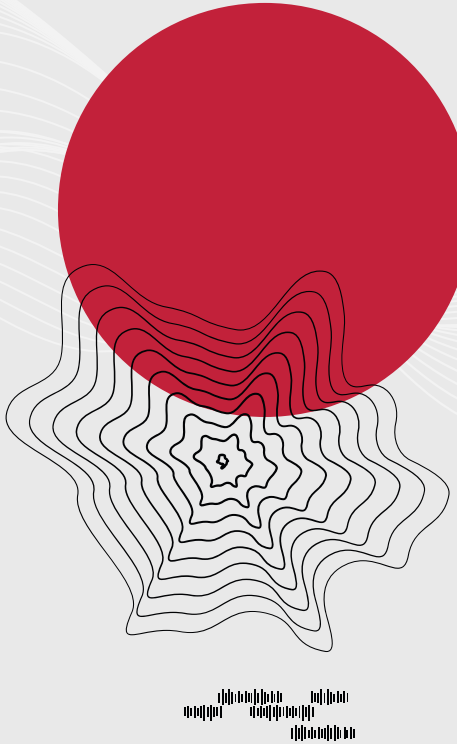


Result

Validation accuracy increased significantly (73%), but still unsatisfactory.

Solution Trial

Removed dropout and added a fully connected layer with 500 units.

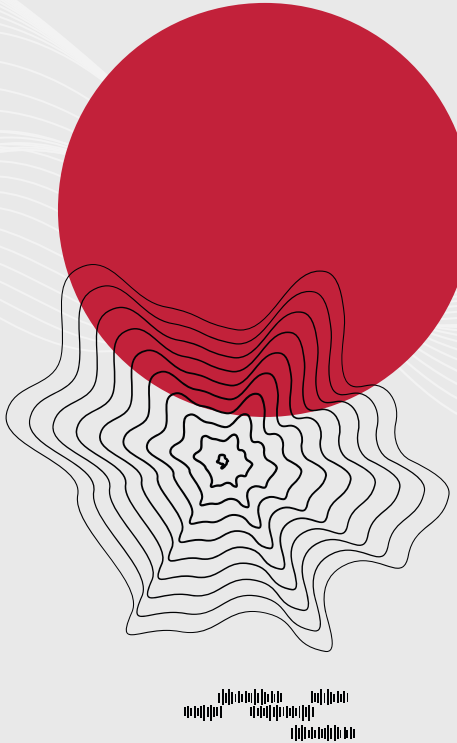


Result

Validation accuracy decreased (72%), and model started to overfit again.

Solution Trial

Changed the fully connected layer from 500 units to 200 units.



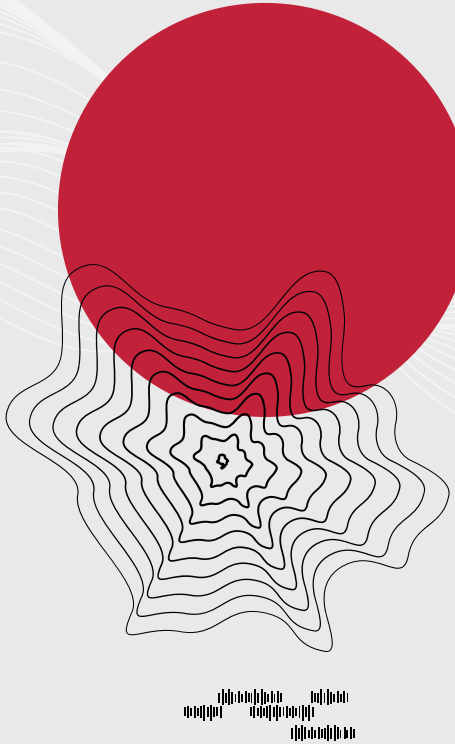
Result

No significant change.

Solution Trial

Feed the model more data

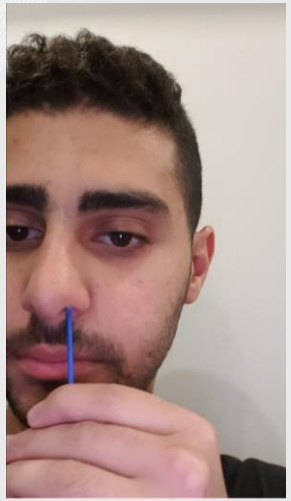
Add a 100-unit fully-connected (FC) layer before the SoftMax layer. The FC layer serves as a linear combinatoric function for the nonlinear activation maps produced in the last convolutional layer of the Resnet.

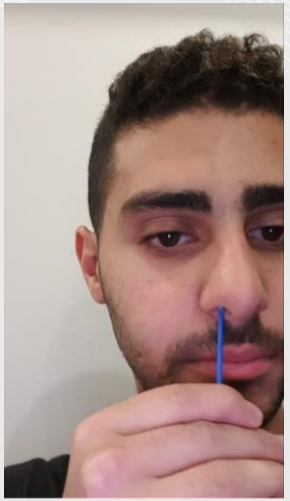


Result

Model performs much better on validation set but still not satisfactory, accuracy is now 85%.

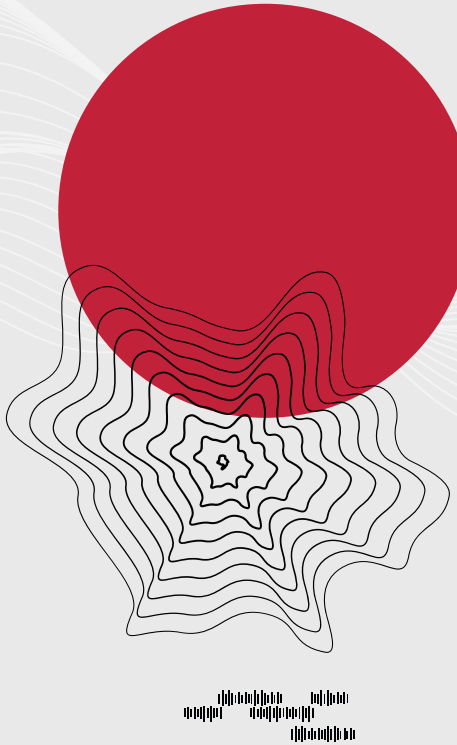
Problem 2:
Model misclassifies side when face is not centered (L/R) But no False Positives





Failed Solution Trial

Tried increasing the dataset by appending augmentated training data (random x-axis translations)



Result

The problem persisted.

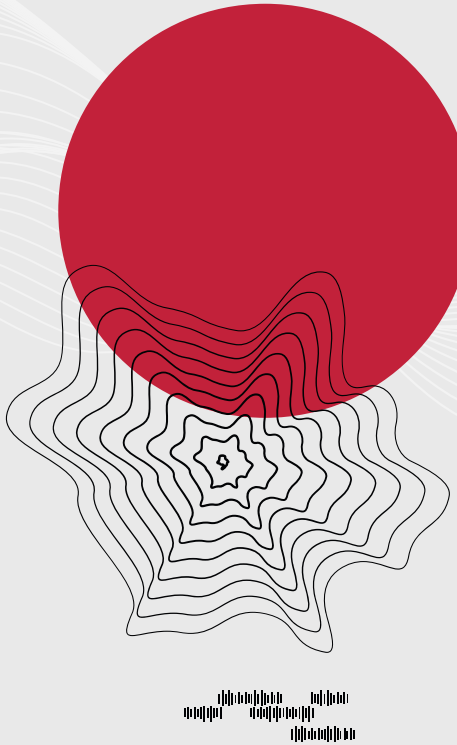
Successfull Solution Trial

Crop faces in dataset and before inference in runtime.

How To Crop Faces?

Option1: Use OpenCV's DNN Face Detector which uses SSD and uses ResNet-10 as its Backbone.

Option2: Use Haar Cascade



Result

Option1: Successful, but many crops with no faces (False Positives) are detected.

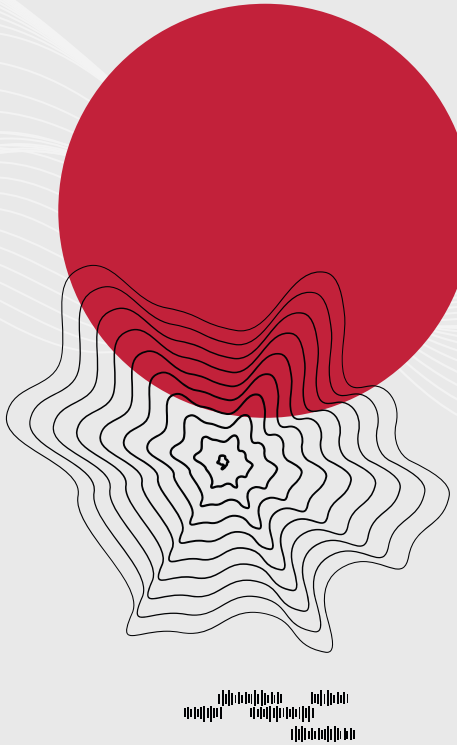
Option2: Few number of faces are detected, not robust to partial face occlusion.

DNN Problem: No-face crops (False Positives)

Solution Trial

Upon Manual Analysis of crops, we found that y-coordinates of false bounding boxes started way too low in the picture (past 1/3 of image height).

How To Crop Faces **Correctly**?
Use OpenCV's DNN Face Detector which uses SSD and uses ResNet-10 as its Backbone. Remove all crops where starting y is more than third of image height.



Result

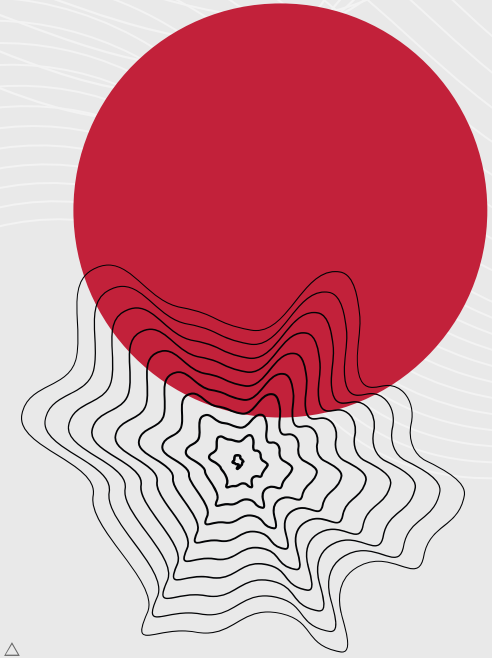
Very good results, nearly no false positives.
Around 95% Accuracy obtained on validation set.

Training Data

Initial

Final

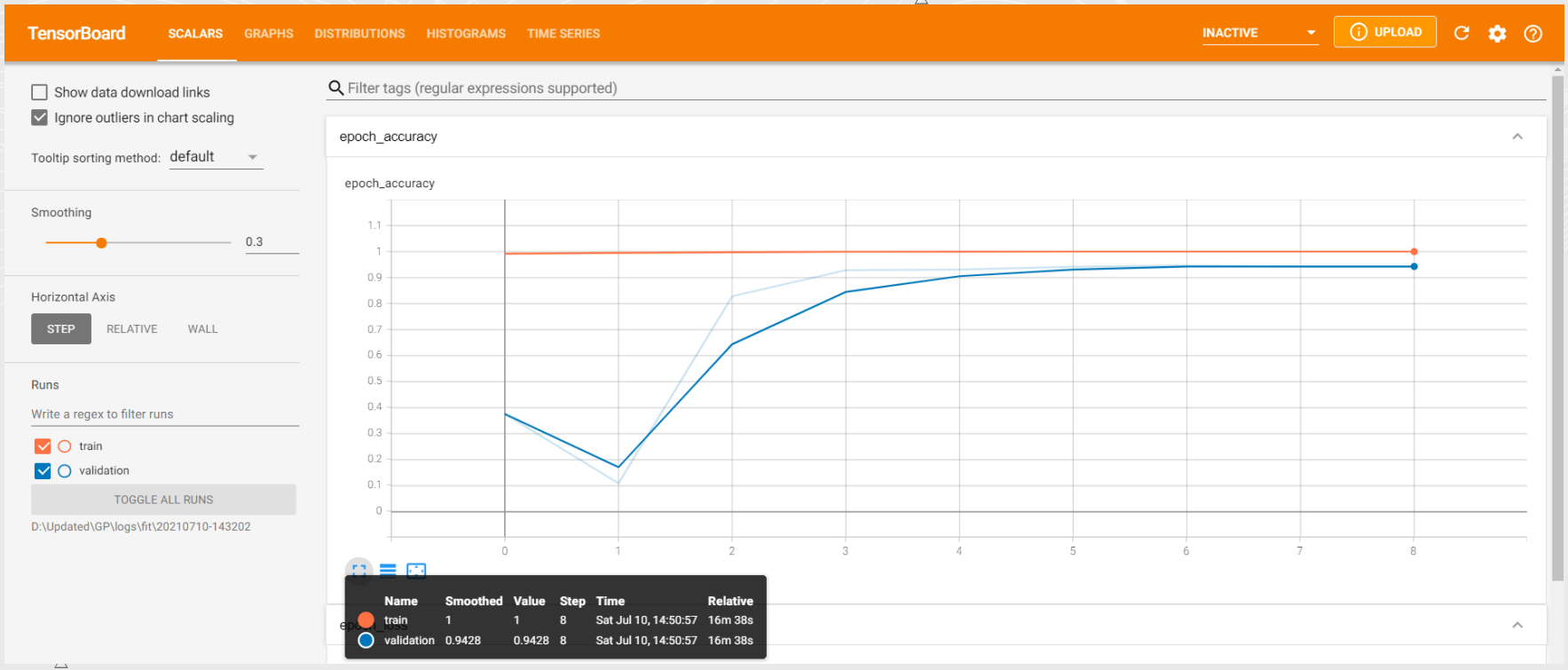
05.
Final Result and Demos



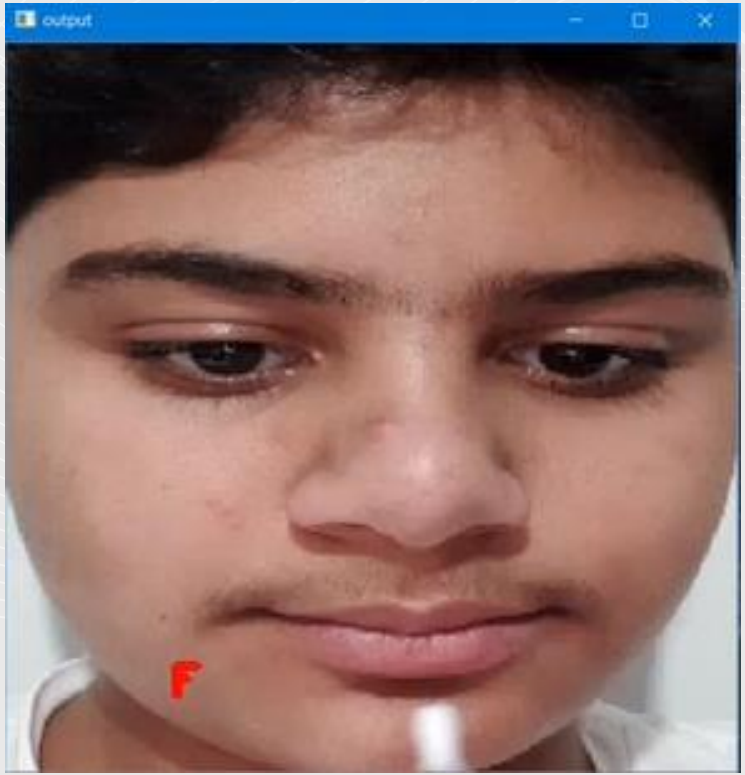
Final Result

Set	Average Accuracy
Training	100%
Test	94.5%

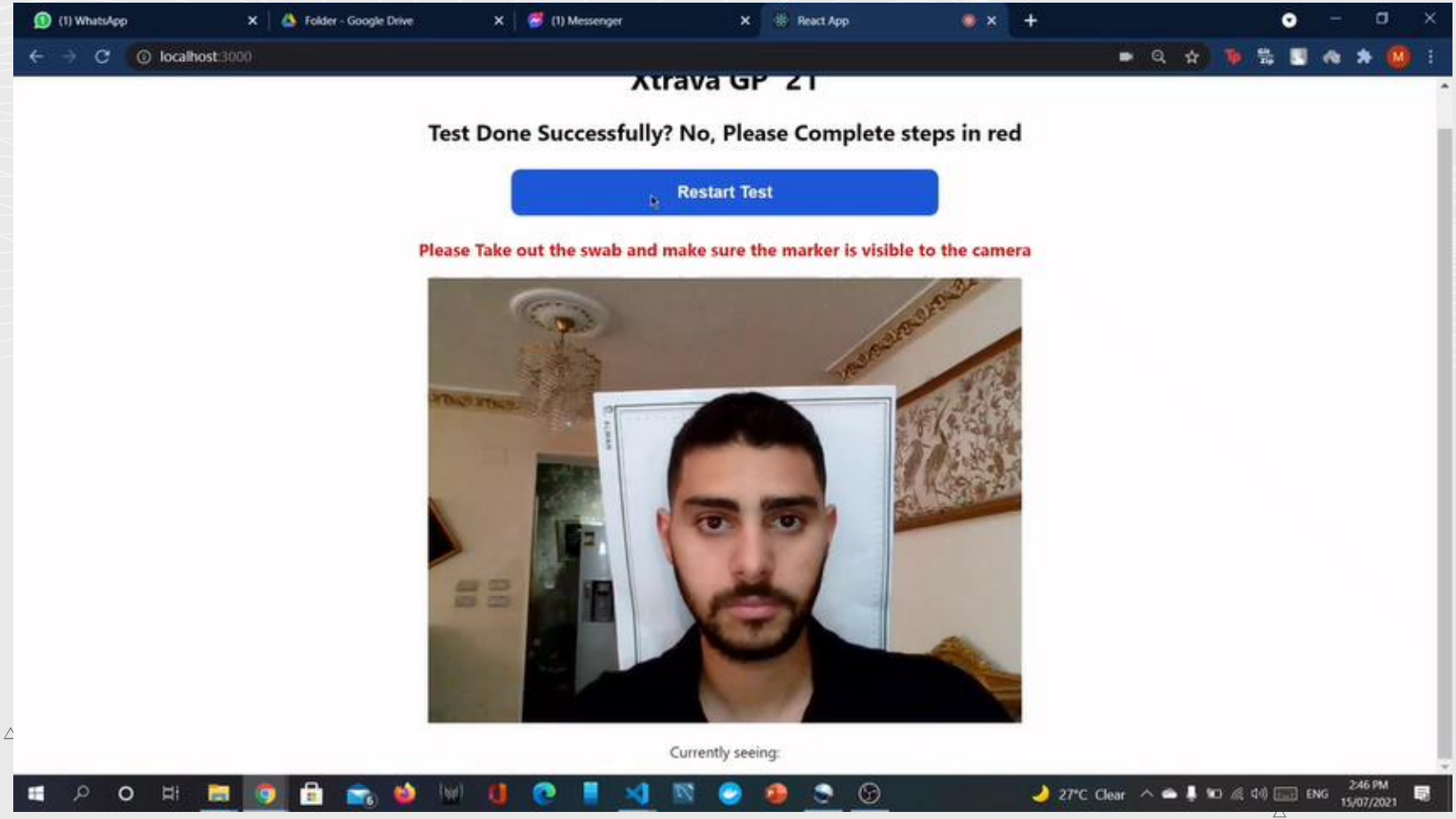
Tensorboard Log



Demo on Data From Different Distributions



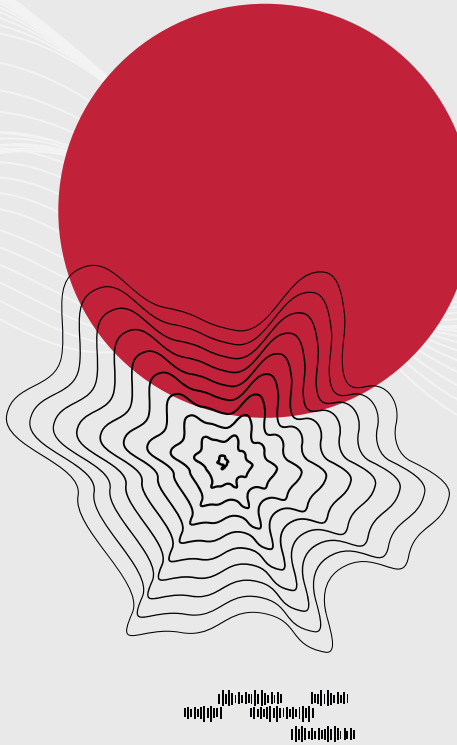
Demo 2 on Data From Different Distributions



06.
Future Enhancements

Non Maximum Suppression

To eliminate false positives in DNN face detector.

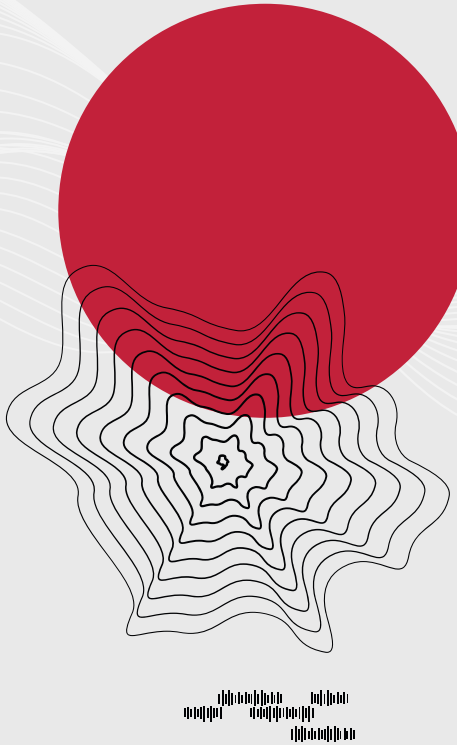


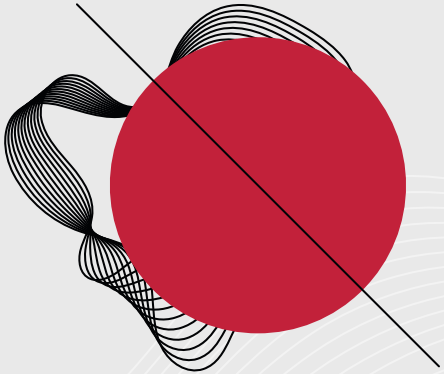
Web App

Hosting the application on Google Cloud Platform which has the needed resources.

Android and iOS application

Mobile applications available in the play/app store for the users.





Thank You!

Any Questions?