



GTU COMPUTER ENGINEERING AGE PREDICTION FROM FACE USING DEEP LEARNING MODEL

**CSE 495
SECOND PRESENTATION**

Süleyman Gölbol

**Project Supervisor: Dr. Burcu YILMAZ
December 2022**



Contents

- Mobile App Schema
- Border Detection
- Azure Connection
- Model Details
- Success Criteria
- Timeline
- References



Mobile App Schema

AGE PREDICTOR



After first meeting I've started to create a mobile application for the project using Flutter.

Model Selector

How To Use ?

Türk Telekom

23% 10:44

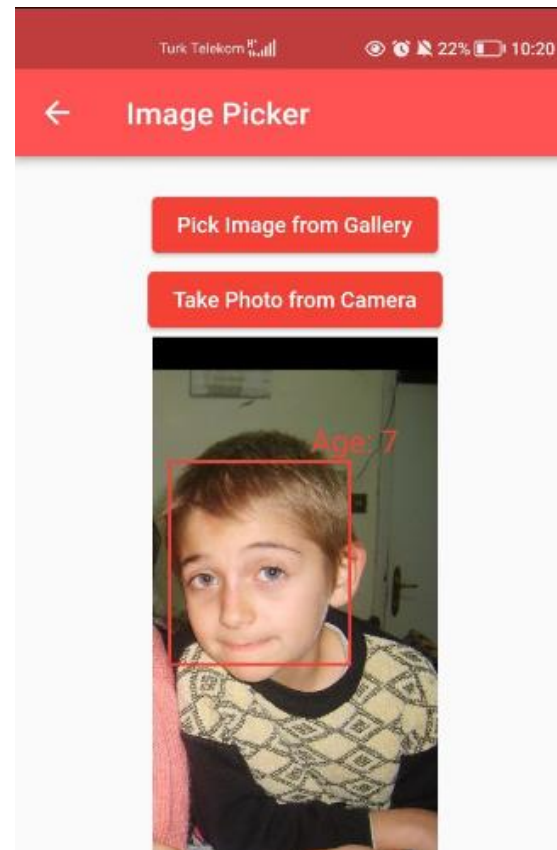
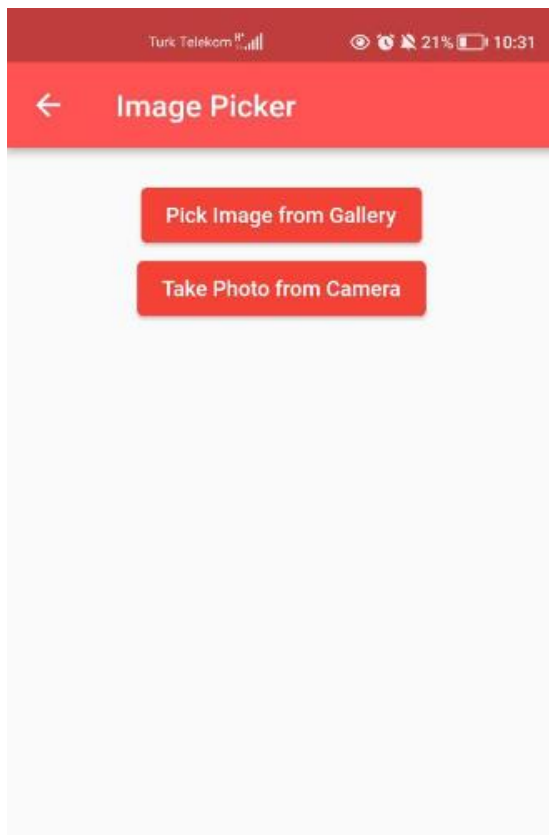
Select the model from list

model 1

IMAGE PAGE



Mobile App Schema

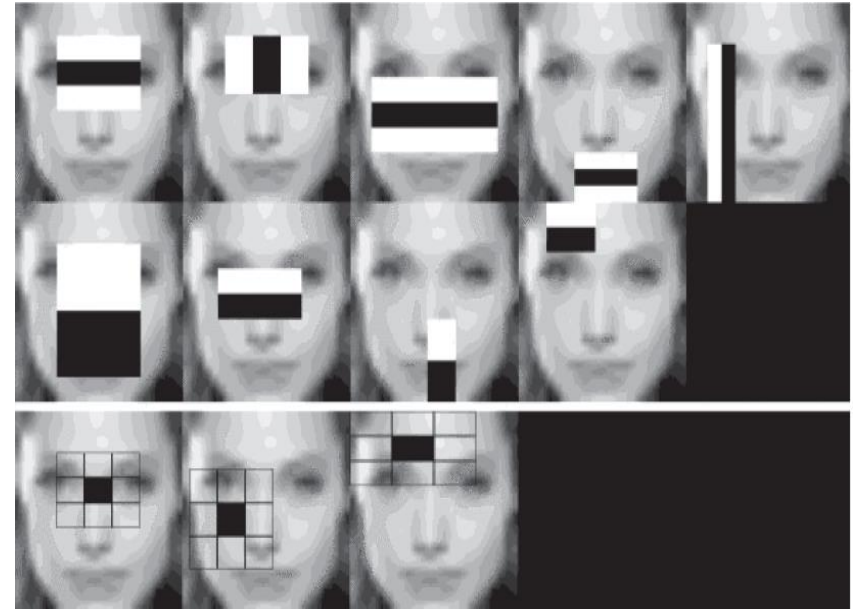


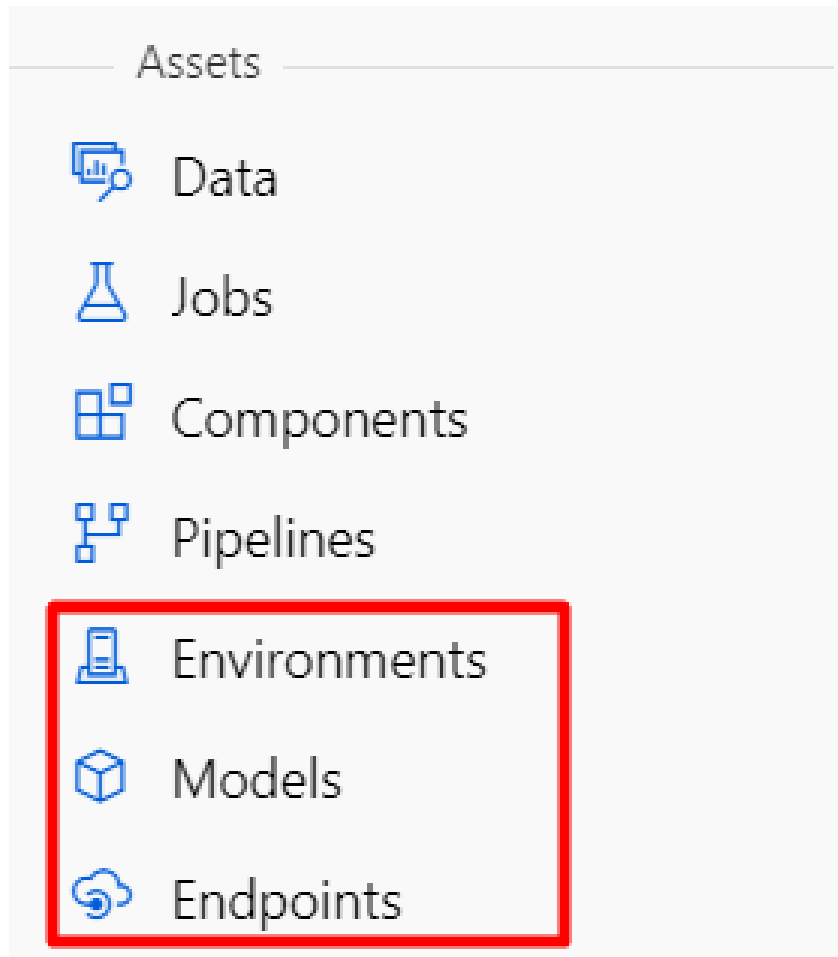
- After clicking on Image Page button, it opens Image Picker page, and it allows us to take a photo or select image from gallery. (Just preview)



Face Border Detection

- To get the face border coordinates I used Haar cascade frontal face xml. It helps to detect face borders to draw rectangle on it.
- Haar cascade uses the cascading window, and it tries to compute features in every window and classify whether it could be a face.





To get the age results I need to connect the mobile app to model.

To deploy the model, I am using Microsoft Azure Machine Learning Studio and Github Actions. (Using Purebasic and Python)

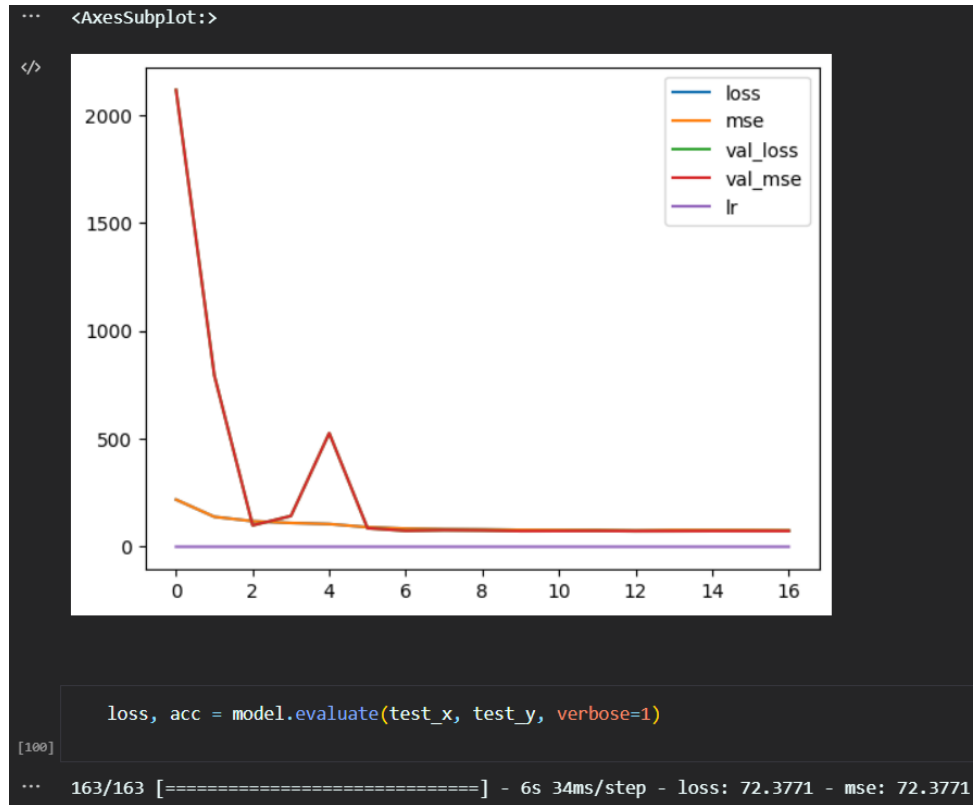
After saving the model, I upload the model to the models section. Then I create a docker image for environment and connect with script.



- For preprocessing, I converted images to pixel values. ✓
- Reshaping every image to become (48,48,1) size. ✓
- Cleansing datasets if it contains unnecessary data. ✓
- Splitting datasets into training and test data. ✓
- Finding suitable algorithms that detects the face features. ✓☑
- Creating/updating model using different layers and activation functions. ✓☑
- Training the faces with multiage pictures of people. ✓
- Sending model over the internet to camera device for prediction. ✗
- Detecting face and using photo as input in the model. ☑
- Detecting the age of person using a mobile application for testing. ✗



First Model Results



- The standard MSE (Mean Squared Error) metric is the item that we check for a regression problem.
- I will decrease the errors for a better model until the next presentation.



Success Criteria

1. Accuracy value over 80% using model for a dataset over 15000 values.
2. Detection should be made at most 5 seconds.
3. Accuracy value over 75% using different model using transfer learning.



1st Meeting (Preliminary Presentation)

- Gathering datasets, creating model.
- Starting to create mobile application.

October 26, 2022,
Wednesday

2nd Meeting

- Training, model fixes on project.
- Connecting application to deployed model for testing.

December 07, 2022,
Wednesday

Report Submission

January 15, 2023, Sunday

Trailer Submission

January 15, 2023, Sunday

3rd Meeting (Final Presentation)

January 18, 2023,
Wednesday

Demo

January 19, 2023, Thursday



1. Sumit Mund, Microsoft Azure Machine Learning, 2015
2. Sidra Mehtab, Jaydip Sen, Face Detection Using OpenCV and Haar Cascades Classifiers, March 2020
3. Jeff Heaton, “Applications of Deep Neural Networks with Keras”, September 2020

