**Persiapan**

1. Download dan install anaconda <https://www.anaconda.com> (Tutorial cara install : <https://ngodingdata.com/instalasi-anaconda-di-windows/> )
2. Buka pencarian, cari Anaconda Command Prompt
3. Buat sebuah environment baru dengan tuliskan di anaconda command prompt (Tutorial membuat environment : <https://medium.com/@hafizhan.aliady/membuat-virtual-environment-di-anaconda-dan-menambahkan-kernel-di-jupyter-notebook-9e3054fb91c2> ). Atau bisa ikuti Langkah dibawah ini:

* create -n <nama environment> pip python (Contoh: create -n facedistance pip python
* aktifkan environment dengan ketik : conda activate <nama environment> (Contoh : conda activate facedistance) #CATATAN : anaconda selalu diaktifkan saat ingin menjalankan program. Jika sudah aktif abaikan.
* install beberapa library dan framework berikut dan ketikan satu persatu pada command prompt:
* pip install cv zone
* pip install opencv-python
* pip install opencv-contrib
* pip install pygame
* pip install tk
* Download program <https://github.com/ASNProject/Face-Distance.git> ekstrak dan simpan dalam folder bebas.
* Buka folder program menggunakan anaconda command prompt dengan mengetikkan : cd <Nama Folder> (Contoh: cd Downloads/Face\_Distance)
* Jalankan program dengan mengetik : python main.py
* Untuk menghentikan program atau stop pada anaconda command prompt gunakan: ctrl+C

#CATATAN:

1. Point 1 dan 3 hanya dilakukan satu kali saat instalasi
2. Nama environment tidak boleh lupa

**Penjelasan Program**

* Import Library dan inisialisasi

from configparser import Interpolation

from tkinter import font

from turtle import distance

import cv2

import math

import pygame

distance = 0.0

font = cv2.FONT\_HERSHEY\_SIMPLEX

* Inisialisasi dan pemanggilan beberapa komponen seperti file model haarcascade dan port untuk kamera

cap = cv2.VideoCapture(0) #memanggil kamera dengan menggunakan opencv

face\_cascade = cv2.CascadeClassifier('haarcascade\_frontalface\_default.xml')

eye\_cascade = cv2.CascadeClassifier('haarcascade\_eye.xml') #memanggil model haarcascade

#CATATAN: harcascade menggunakan 2 buah yaitu untuk deteksi wajah dan mata. Keduanya opsional bisa hanya salah satu yang digunakan (Utama menggunakan wajah)

* Membaca kamera dan mempersiapkan frame

ret, huge\_frame = cap.read()

frame = cv2.resize(huge\_frame, (0,0), fx=0.5, fy=0.5, interpolation=cv2.INTER\_NEAREST)

* Merubah hasil gambar dari kamera menjadi keabuan/gray

gray = cv2.cvtColor(frame, cv2.COLOR\_BGR2GRAY)

faces = face\_cascade.detectMultiScale(gray, 1.3, 5)

* Pembacaan wajah dan mata

for (x, y, w, h) in faces:

# Mengenali wajah dan memberikan kotak pada wajah yang terdeteksi

cv2.rectangle(frame, (x,y), (x+w,y+h), (255,0,0), 2)

roi\_gray = gray[y:y+h, x:x+w]

roi\_color = frame[y:y+h, x:x+w]

# Opsional jika tidak ingin menggunakan mata silahkan komen atau hapus program dibawah

eyes = eye\_cascade.detectMultiScale(roi\_gray)

for (ex, ey, ew, eh) in eyes:

cv2.rectangle(roi\_color,(ex,ey),(ex+ew,ey+eh),(0,255,0),2)

* Perhitungan jarak (Saya mengambil referensi dari jurnal yang ada didalam Folder Referensi)

distancei = (2\*3.14 \* 180)/(w+h\*360)\*1000 + 3

print (distancei)

distance = int(distancei\*2.54)

* Logika jika jarak wajah dekat dengan layer

cv2.putText(frame, 'Jarak = ' + str(distance) + ' Cm', (5,100), font, 1,(255,255,255),2)

if distance <= 26:

pygame.mixer.init()

pygame.mixer.music.load('beep.wav')

pygame.mixer.music.play()

#CATATAN: Logika dalam progam ini jika jarak wajah kurang dari 26 Cm maka akan berbunyi ‘beep’ dan jika lebih dari itu tidak berbunyi apapun.

SCREENSHOOT



