Introduction to Image Recognition

@OsloMet Makerspace

Overview

# Computer Vision

# Viola-Jones Haar-Feature

# Installing OpenCV

# Face detection

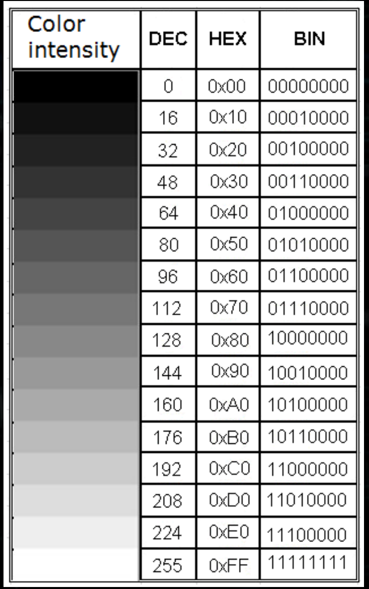
# Face tracking on camera

# Template matching

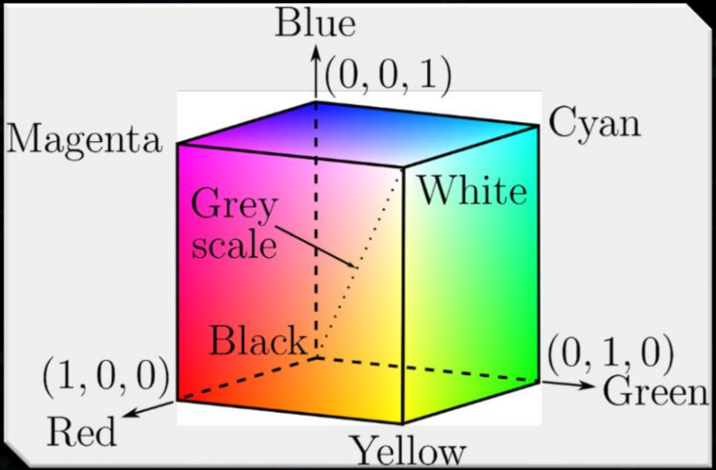
# Github link:

https://github.com/makeriet/image-recognition

Computer Vision

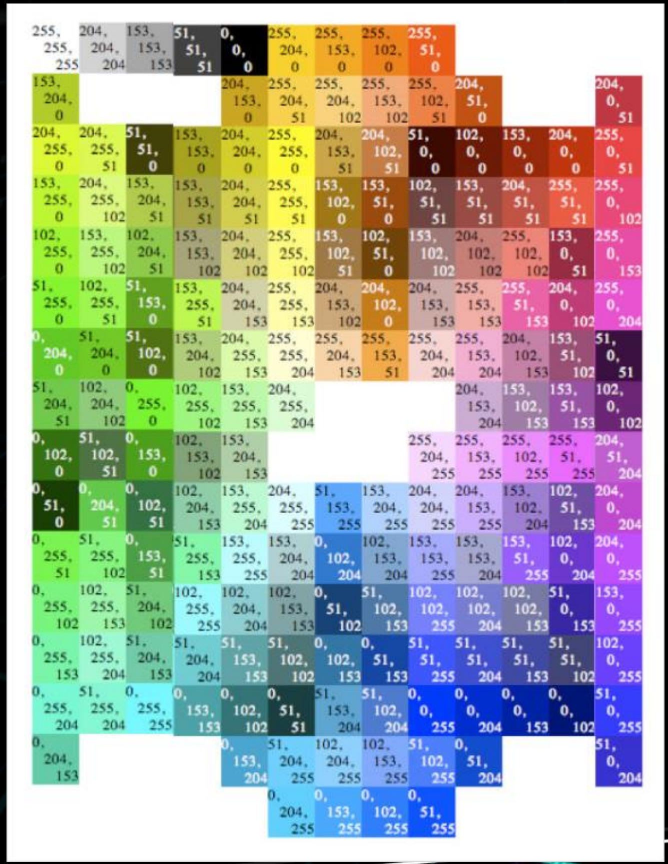


Color intensity	DEC	HEX	BIN
⬛	0	0x00	00000000
▨	16	0x10	00010000
▨	32	0x20	00100000
▨	48	0x30	00110000
▨	64	0x40	01000000
▨	80	0x50	01010000
▨	96	0x60	01100000
▨	112	0x70	01110000
▨	128	0x80	10000000
▨	144	0x90	10010000
▨	160	0xA0	10100000
▨	176	0xB0	10110000
▨	192	0xC0	11000000
▨	208	0xD0	11010000
▨	224	0xE0	11100000
⬜	255	0xFF	11111111



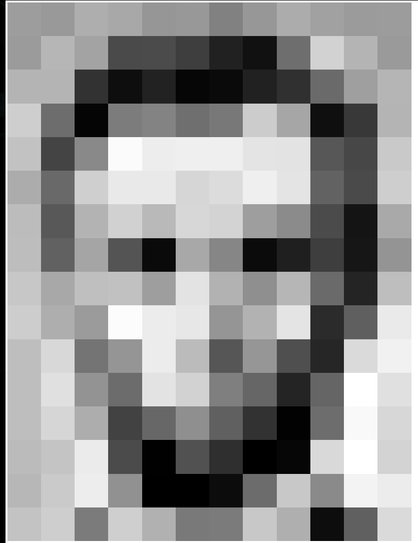
255 x 255 x 255 = 16581375

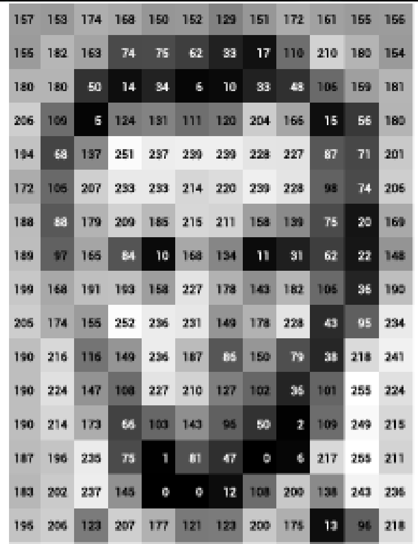
R G B

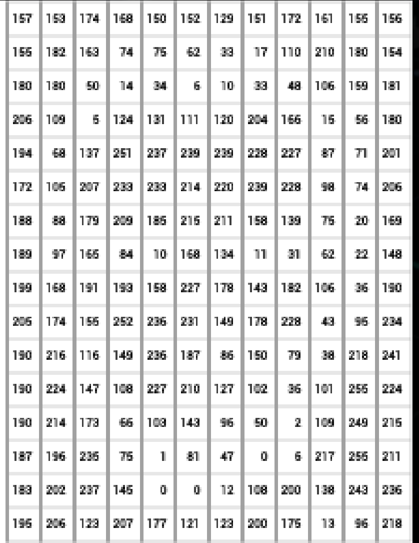


12px

16px



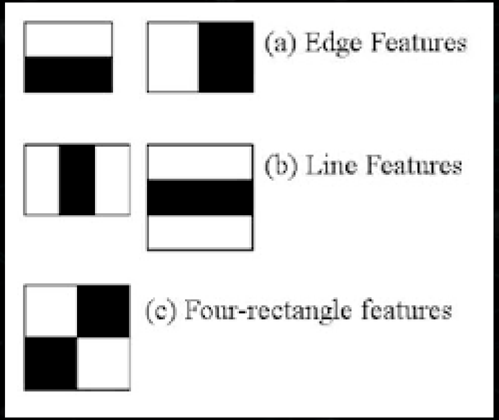




157	153	174	168	150	152	129	151	172	161	155	156
155	182	163	74	75	62	33	17	110	210	180	154
180	180	50	14	34	6	10	33	48	106	159	181
206	109	5	124	131	111	120	204	166	15	56	180
194	68	137	251	237	239	239	228	227	87	71	201
172	105	207	233	233	214	220	239	228	98	74	206
188	88	179	209	185	215	211	158	139	75	20	169
189	97	165	84	10	168	134	11	31	62	22	148
199	168	191	193	158	227	178	143	182	106	36	190
205	174	155	252	236	231	149	178	228	43	95	234
190	216	116	149	236	187	86	150	79	38	218	241
190	224	147	108	227	210	127	102	36	101	255	224
190	214	173	66	103	143	96	50	2	109	249	215
187	196	235	75	1	81	47	0	6	217	255	211
183	202	237	145	0	0	12	108	200	138	243	236
195	206	123	207	177	121	123	200	175	13	96	218

Viola-Jones Haar-Feature

- # algorithm used to identify objects in an image or video
- # proposed by Paul Viola and Michael Jones in 2001
- # powerful face/non-face classifiers can be constructed based on these features
- # they can be computed efficiently using the summed-area table or integral image technique.



Installing OpenCV

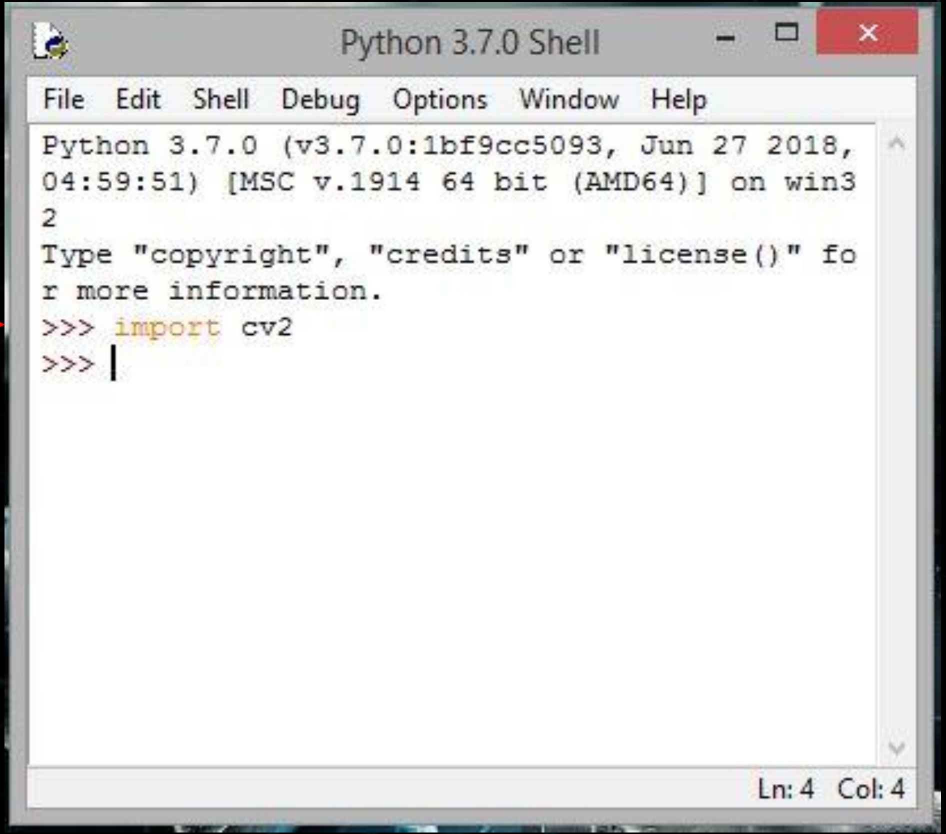
1. Open **“Command Prompt”** in Windows or **“Terminal”** in Mac OS.

2. Enter the following command:

pip install opencv-python (for Python version 2)

-OR

pip3 install opencv-python (for Python version 3)



```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018,
04:59:51) [MSC v.1914 64 bit (AMD64)] on win3
2
Type "copyright", "credits" or "license()" fo
r more information.
>>> import cv2
>>> |
```

OpenCV classifier

Location >>>> C:\Program Files\Python37\Lib\site-packages\cv2\data

haarcascade_eye_tree_eyeglasses.xml
haarcascade_mcs_leftear.xml haarcascade_eye.xml
haarcascade_mcs_lefteye.xml
haarcascade_frontalface_alt2.xml
haarcascade_mcs_mouth.xml
haarcascade_frontalface_alt_tree.xml
haarcascade_mcs_nose.xml
haarcascade_frontalface_alt.xml
haarcascade_mcs_rightear.xml
haarcascade_frontalface_default.xml
haarcascade_mcs_righteye.xml haarcascade_fullbody.xml
haarcascade_mcs_upperbody.xml

haarcascade_lefteye_2splits.xml
haarcascade_profileface.xml haarcascade_lowerbody.xml
haarcascade_righteye_2splits.xml
haarcascade_mcs_eyepair_big.xml haarcascade_smile.xml
haarcascade_mcs_eyepair_small.xml
haarcascade_upperbody.xml

Face detection

# Import OpenCv

# Read the image file

# Resize picture if required

# Convert image into grayscale

# Initialize haar cascade feature for frontalface

# Detect faces on image

# Draw the rectangle around the image

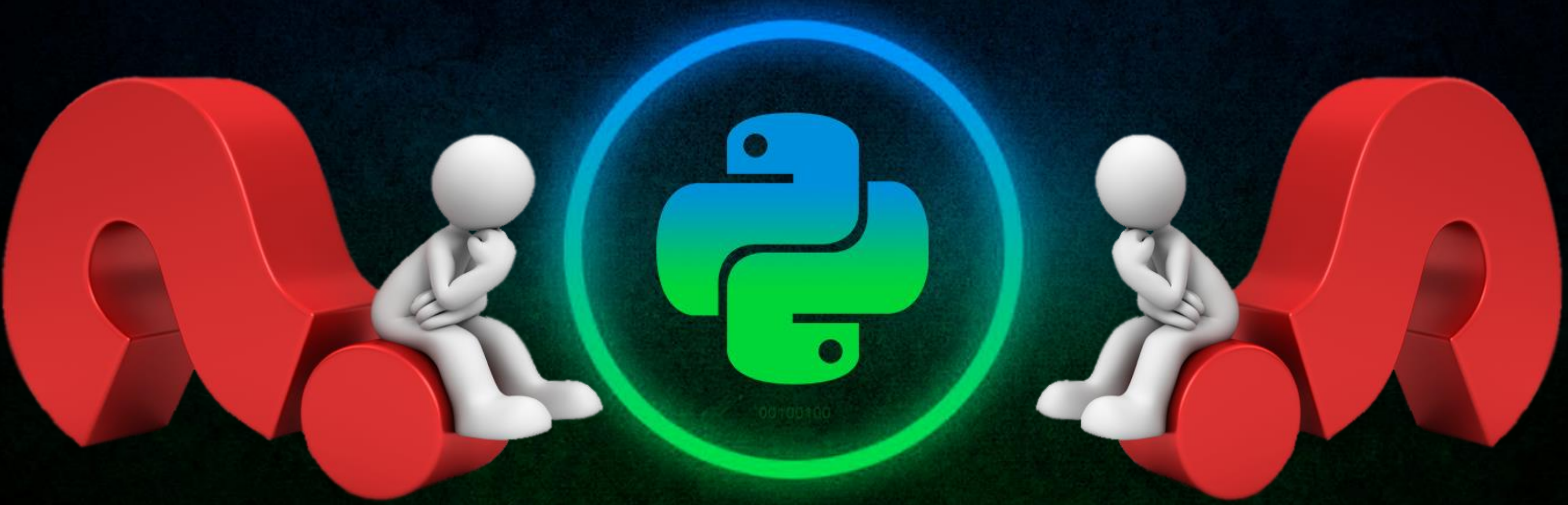
# Display the output image with faces detected

Face tracking

```
# Import OpenCv

# Initialize webcam

# Initialize haar cascade feature for frontalface

# Create a loop to continuously read the video

    # Resize or flip the video if required

    # Convert the video into grayscale and also blur it

    # Detect faces in the video

        # Draw the rectangle around the faces in the video

    # Display the video with the rectangle

    # Release the webcam when you close the program
```

Template matching

- Import OpenCv
- Initialize webcam
- Open a template image and get its width and height
- Create a loop to continuously read the video
    - Resize or flip the video if required
    - Convert the video into grayscale and also blur it
    - Match image template with the blurred video
    - Find the area where the template is matched for maximum time
    - Draw the rectangle around the area in the video, where the template is matching
    - Display the video with the rectangle
    - Release the webcam when you close the program



```
print ("Any Questions?")
print (”Thank You”)
```