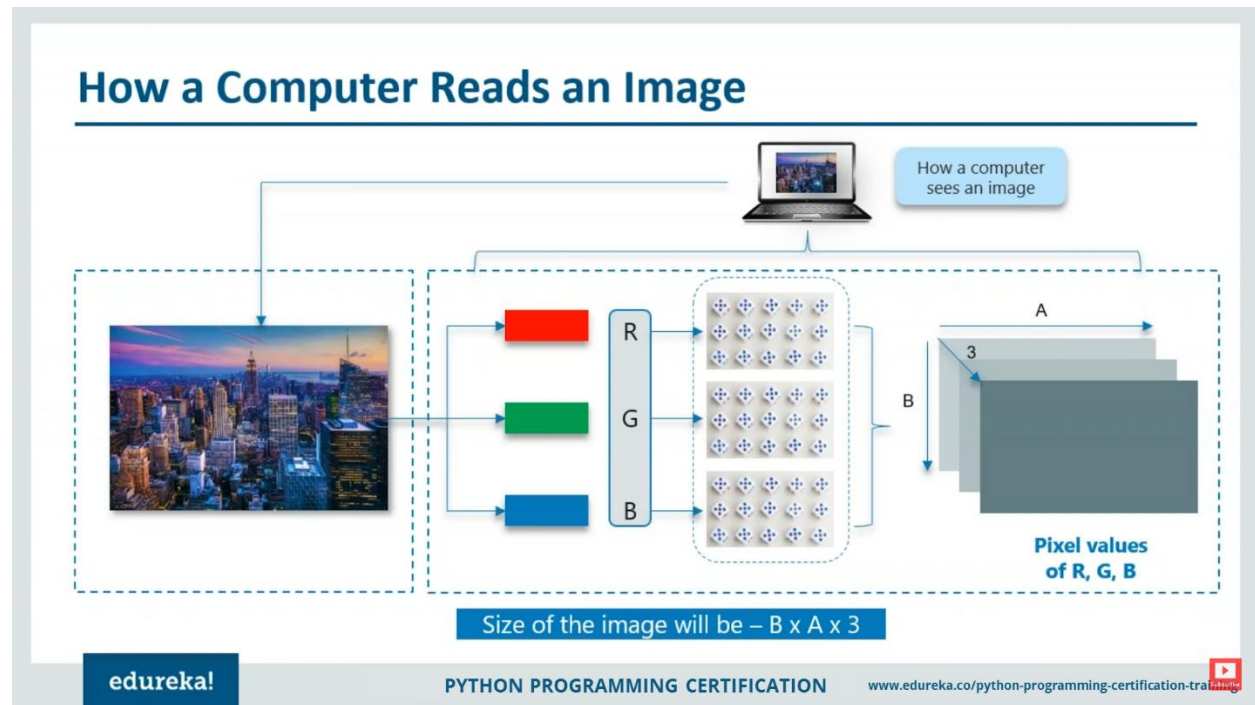


How a Computer Reads an Image?

Computer see a form of matrix numbers between 0 to 255. For Colored images there will be a 3 Channels (*R, G, B*; *Size of the Image BxAx3, 700(A)x1400(B)*) while Grayscale or Black and White Color there is only 1 channel.



What is OpenCV?

All the OpenCV array structures are converted to and from **NumPy** arrays. OpenCV-Phyton is a library of Phyton designed to solve computer vision problems.

Basic Operations with OpenCV

Video Tutorial: OpenCV Python Tutorial | Creating Face Detection System And Motion Detector Using OpenCV | Edureka

link: <https://www.youtube.com/watch?v=-ZrDjwXZGxl>

- Load Images Using OpenCV
- Image Shape / Resolution
- Displaying the Image
- Resizing the Image
- Face Detection
- Capturing Video
- Use Case – Motion Detector
- Bokeh Plot (Motion Graph)

Load Images using OpenCV

Load Images Using OpenCV

```
import cv2

# Colored Image
img = cv2.imread("Penguins.jpg",1)

# Black and White (Gray Scale)
img_1 = cv2.imread("Penguins.jpg",0)
```

Python stores the image as a NumPy array / matrix of numbers

Import the OpenCV Module

Read the image in RGB / Colored format

Read the image as a gray scale image or black and white image

Path to the image

edureka! PYTHON PROGRAMMING CERTIFICATION www.edureka.co/python-programming-certification-training

Image Shape / Resolution

Image Shape / Resolution

```
import cv2

# Black and White (Gray Scale)
img = cv2.imread("Penguins.jpg",0)

print(img.shape)
```

ic Operations

C:\Users\Saurabh\AppData\Local\Progr

(768, 1024)

Shape of the NumPy array

768 rows and 1024 columns

Go ahead and do that for colored image as well and notice the difference in the shape and the image NumPy array

edureka! PYTHON PROGRAMMING CERTIFICATION www.edureka.co/python-programming-certification-training

Displaying the Image

Displaying The Image

```
# Black and White (Gray Scale)
img = cv2.imread("Penguins.jpg",0)
cv2.imshow("Penguins", img)
cv2.waitKey(0)
# cv2.waitKey(2000)
cv2.destroyAllWindows()
```

The diagram illustrates the execution of the code for displaying an image. Arrows point from specific code elements to descriptive boxes:

- `img = cv2.imread("Penguins.jpg",0)` points to "Image object".
- `cv2.imshow("Penguins", img)` points to "Opens a window to display the image".
- `cv2.waitKey(0)` points to "Name of the window".
- `# cv2.waitKey(2000)` points to "Wait until a user presses a key".
- `cv2.destroyAllWindows()` points to "Wait for 2000 milliseconds" and "Closes the window based on waitforkey parameter".

edureka! PYTHON PROGRAMMING CERTIFICATION www.edureka.co/python-programming-certification-training

Resizing the Image

Resizing The Image

```
import cv2

# Black and White (Gray Scale)
img = cv2.imread("Penguins.jpg",0)
resized_image = cv2.resize(img, (650,500))
cv2.imshow("Penguins", resized_image)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

The diagram illustrates the execution of the code for resizing an image. Arrows point from specific code elements to descriptive boxes:

- `resized_image = cv2.resize(img, (650,500))` points to "Resized Image".
- `cv2.imshow("Penguins", resized_image)` points to "New image shape".

edureka! PYTHON PROGRAMMING CERTIFICATION www.edureka.co/python-programming-certification-training

Resizing The Image

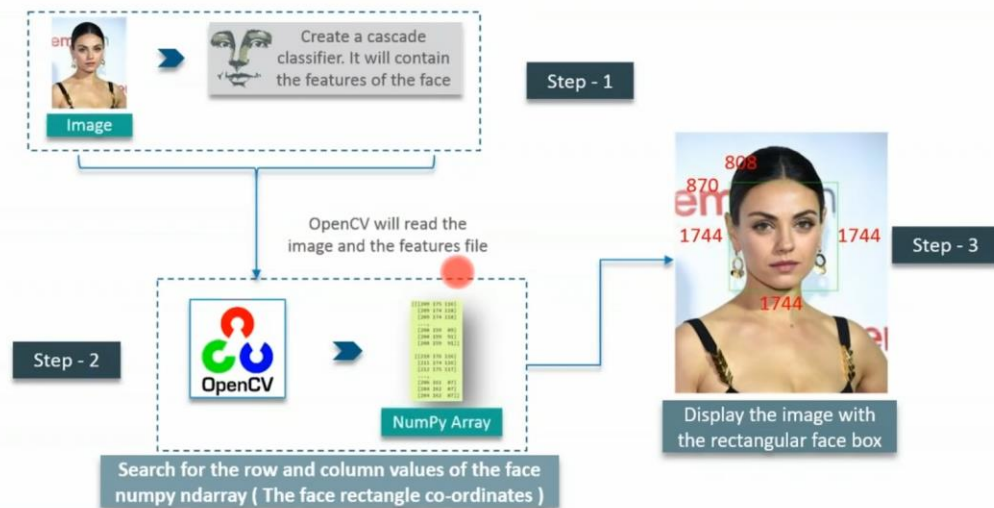
```
resized_image = cv2.resize(img, (intint))
```

New image shape = Old image shape/2



Face Detection

Face Detection Using OpenCV



Face Detection Using OpenCV

```
import cv2

# Create a CascadeClassifier Object
face_cascade = cv2.CascadeClassifier("haarcascade_frontalface_default.xml")

# Reading the image as it is
img = cv2.imread("photo.jpg")

# Reading the image as gray scale image
gray_img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

# Search the co-ordinates of the image
faces = face_cascade.detectMultiScale(gray_img, scaleFactor=1.05,
                                      minNeighbors=5)

print(type(faces))
print(faces)
```

1. Use your PIP pack manager to show the path of packages installed.

run in cmd: *pip show opencv-python*

Face Detection Using OpenCV

```
# Create a CascadeClassifier Object
face_cascade = cv2.CascadeClassifier("haarcascade_frontalface_default.xml")
```

Create a CascadeClassifier Object

Path to the xml file which contains the face features

```
# Reading the image as gray scale image
gray_img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
```

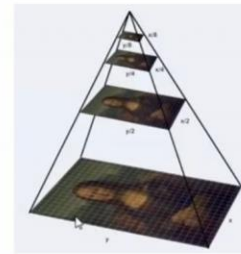
Converting colored image to gray scale

Face Detection Using OpenCV

```
# Search the co-ordinates of the image  
faces = face_cascade.detectMultiScale(gray_img, scaleFactor = 1.05,  
                                     minNeighbors=5)
```

Method to search for the face rectangle co-ordinates

Decreases the shape value by 5%, until the face is found.
Smaller this value, the greater is the accuracy



Face Detection Using OpenCV - Output

```
print(type(faces))  
print(faces)
```

Output

```
<class 'numpy.ndarray'>  
[[ 870  808 1744 1744]]
```



edureka!

PYTHON PROGRAMMING CERTIFICATION

www.edureka.co/python-programming-certification-trainings



Adding The Rectangular Face Box

Let's add the rectangular face box

edureka!

Copyright © 2018, edureka and/or its affiliates. All rights reserved.



Face Detection Using OpenCV

```
for x,y,w,h in faces:  
    img = cv2.rectangle(img, (x,y), (x+w,y+h), (0,255,0), 3)
```

Method to create
the face rectangle

Image object

(x,y)

Width of the
rectangle

RGB value of the
rectangle outline

(x+w,y+h)

edureka!

PYTHON PROGRAMMING CERTIFICATION

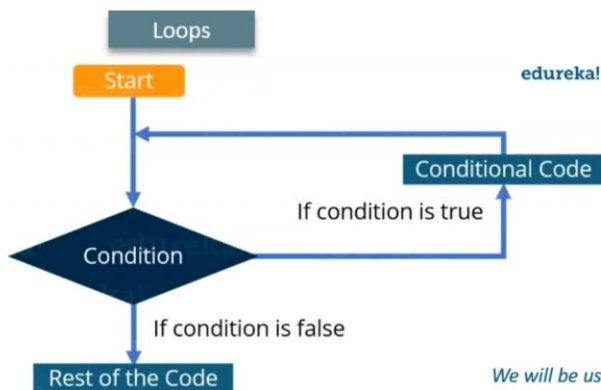
www.edureka.co/python-programming-certification-tra



Capturing a Video

Capturing Video

We will be using OpenCV for reading frames/images one-by-one.



We will be using loops to build a window where images
will appear really fast, so that you can see it as a video

edureka!

PYTHON PROGRAMMING CERTIFICATION

www.edureka.co/python-programming-certification-tra



Capturing Video

```
import cv2
```

```
video = cv2.VideoCapture(0)
```

```
video.release()
```

This will release the camera
in some milliseconds

Either give the path to the video file or use
numbers. Numbers specify that you will be
using the webcam to capture video

Method to create VideoCapture
object. It will trigger the camera

'0' is to specify that
use built-in camera

Note: If I have an external cam, I can put '1' to use that. Similarly, if I have two external cams and I want to use the third cam I can put '3'.

edureka!

PYTHON PROGRAMMING CERTIFICATION

www.edureka.co/python-programming-certification-training



Capturing Video

When you execute the code, you will notice that your cam light
switches on for split seconds, and then it turns off



*Let's go ahead
and add time
delay, using
Time module*

edureka!

PYTHON PROGRAMMING CERTIFICATION

www.edureka.co/python-programming-certification-training



Capturing Video

```
import cv2,time
```

```
video = cv2.VideoCapture(0)
```

```
time.sleep(3)
```

```
video.release()
```

Import the time module

This will stop the script for 3 seconds

When you execute the above code, the web cam will be on for 3 seconds

edureka!

PYTHON PROGRAMMING CERTIFICATION

www.edureka.co/python-programming-certification-trainings



Now, Let's Add a Window That Shows The Video

edureka!

Copyright © 2018, edureka and/or its affiliates. All rights reserved.



Capturing Video

```
import cv2,time
```

```
video = cv2.VideoCapture(0)
```

```
check, frame = video.read()
```

```
print(check)
```

```
print(frame)
```

```
time.sleep(3)
```

```
video.release()
```

It is a NumPy array, it represents the first image that video captures

It is bool data type, returns true if Python is able to read the VideoCapture object



Capturing Video

- ❑ We need to create a frame object, which will read the images of the VideoCapture object.
- ❑ We will recursively show each frame of the video being captured.

```
import cv2,time
```

```
video = cv2.VideoCapture(0)
```

```
check, frame = video.read()
```

```
time.sleep(3)
```

```
cv2.imshow('Capturing', frame)
```

```
cv2.waitKey(0)
```

```
video.release()
```

```
cv2.destroyAllWindows
```

This will read the first frame/image of the video

imshow method is used to capture the first image/frame of the video



How to Capture the Video, Instead of First Image/Frame of the Video?

edureka!

Copyright © 2018, edureka and/or its affiliates. All rights reserved.



Capturing Video

In order to capture the video, we will be using 'while' loop. While condition will be such that, until unless 'check' is True, Python will display the frames.

```
import cv2,time

video = cv2.VideoCapture(0)

a = 1

while True:
    a = a + 1
    check, frame = video.read()
    print (frame)
    gray = cv2.cvtColor(frame,cv2.COLOR_BGR2GRAY)
    cv2.imshow('Capturing',gray)
    key = cv2.waitKey(1)
    if key == ord('q'):
        break

print(a) # This will print the number of frames
video.release()
cv2.destroyAllWindows
```

Convert each frame into a gray scale image

This will iterate through the frames and display the window

This will generate a new frame after every 1 milliseconds

Once you enter 'q' the window will be destroyed

edureka!

PYTHON PROGRAMMING CERTIFICATION

www.edureka.co/python-programming-certification-training

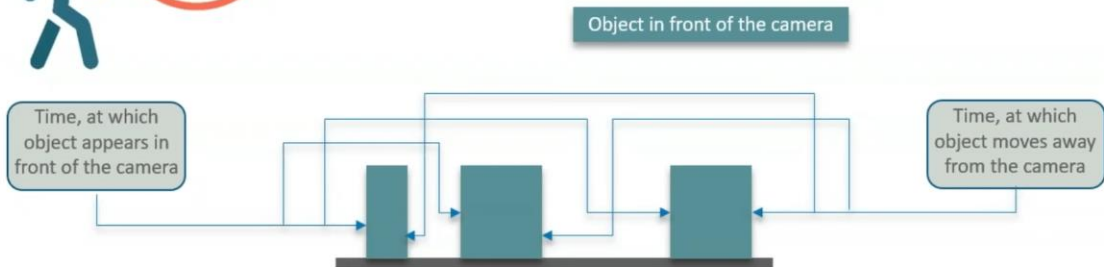


Using Case – Motion Detection

Problem Statement



You have been approached by a company that is studying human behavior. Your task is to give them a webcam, that can detect the motion or any movement in front of it. This should return a graph, this graph should contain for how long the human/object was in front of the camera



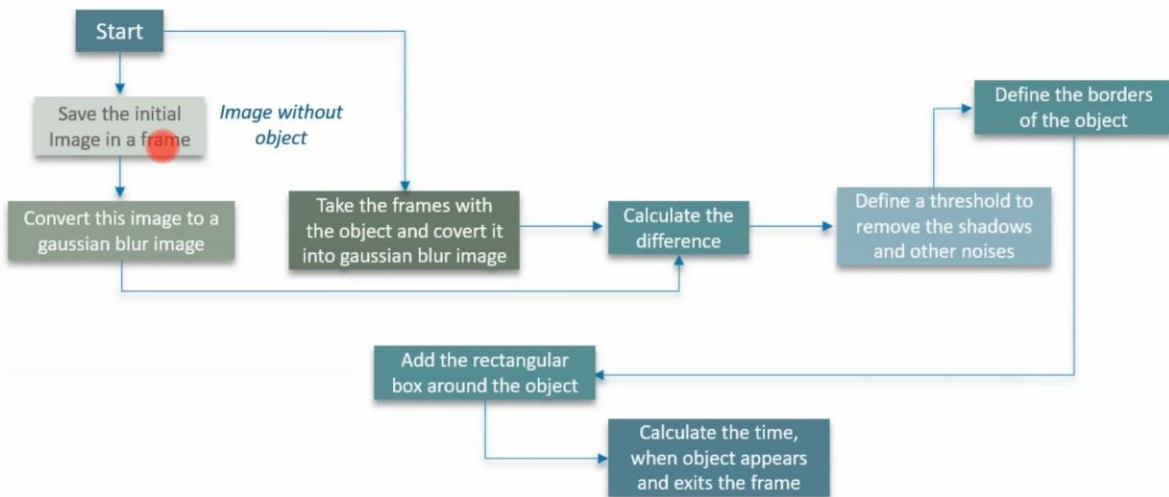
edureka!

PYTHON PROGRAMMING CERTIFICATION

www.edureka.co/python-programming-certification-training



Solution Logic



edureka!

PYTHON PROGRAMMING CERTIFICATION

www.edureka.co/python-programming-certification-training



Motion Detector

```
import cv2,time  
  
first_frame = None
```

```
video = cv2.VideoCapture(0)
```

Create a VideoCapture object to record video using web cam

```
while True:
```

```
    check, frame = video.read()
```

```
    gray = cv2.cvtColor(frame,cv2.COLOR_BGR2GRAY)
```

Convert the frame color to gray scale

```
    gray = cv2.GaussianBlur(gray,(21,21),0)
```

Convert the gray scale frame to GaussianBlur

```
    if first_frame is None:  
        first_frame = gray  
        continue
```

This is used to store the first image/frame of the video

edureka!

PYTHON PROGRAMMING CERTIFICATION

www.edureka.co/python-programming-certification-tra



Motion Detector

```
delta_frame = cv2.absdiff(first_frame,gray)
```

Calculates the difference between the first frame and other frames

```
thresh_delta = cv2.threshold(delta_frame, 30, 255, cv2.THRESH_BINARY)[1]
```

```
thresh_delta = cv2.dilate(thresh_delta, None, iterations=0 )
```

```
(_,cnts,_) = cv2.findContours(thresh_delta.copy(),cv2.RETR_EXTERNAL,  
                             cv2.CHAIN_APPROX_SIMPLE)
```

Provides a threshold value, such that it will convert the difference value with less than 30 to black. If the difference is greater than 30 it will convert those pixels to white

```
for contour in cnts:  
    if cv2.contourArea(contour) < 1000:  
        continue  
    (x, y, w, h) = cv2.boundingRect(contour)  
    cv2.rectangle(frame, (x,y), (x+w,y+h),(0,255,0), 3)
```

Define the contour area. Basically, add the borders

```
cv2.imshow('frame',frame)  
cv2.imshow('Capturing',gray)  
cv2.imshow('delta',delta_frame)  
cv2.imshow('thresh',thresh_delta)
```

Removes noises and shadows. Basically, it will keep only that part white, which has area greater than 1000 pixels

Creates a rectangular box around the object in the frame

edureka!

PYTHON PROGRAMMING CERTIFICATION

www.edureka.co/python-programming-certification-tra



Now, We Can Calculate The Time For Which The Object Was In Front Of The Camera

edureka!

Copyright © 2018, edureka and/or its affiliates. All rights reserved.



Storing Time Values

```
first_frame=None
status_list=[None,None]
times=[]
df=pandas.DataFrame(columns=["Start","End"])

video=cv2.VideoCapture(0)

while True:
    check, frame = video.read()
    status=0
    gray=cv2.cvtColor(frame,cv2.COLOR_BGR2GRAY)
    gray=cv2.GaussianBlur(gray, (21,21), 0)
```

DataFrame to store the time values during which object detection and movement appears

Status at the beginning of the recording is zero as the object is not visible

edureka!

PYTHON PROGRAMMING CERTIFICATION

www.edureka.co/python-programming-certification-training



Storing Time Values

```
(_, cnts, _) = cv2.findContours(thresh_frame.copy(), cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)

for contour in cnts:
    if cv2.contourArea(contour) < 10000:
        continue
    status = 1
```

Change in status when the object is being detected



Storing Time Values

```
(x, y, w, h) = cv2.boundingRect(contour)
cv2.rectangle(frame, (x, y), (x+w, y+h), (0, 255, 0), 3)
status_list.append(status)
```

List of status for every frame

```
status_list = status_list[-2:]

if status_list[-1] == 1 and status_list[-2] == 0:
    times.append(datetime.now())
if status_list[-1] == 0 and status_list[-2] == 1:
    times.append(datetime.now())
```

Record datetime in a list when change occurs



Storing Time Values

```
print(status_list)
print(times)

for i in range(0, len(times), 2):
    df=df.append({"Start":times[i], "End":times[i+1]}, ignore_index=True)

df.to_csv("Times.csv")

video.release()
cv2.destroyAllWindows
```

Store time values
in a DataFrame

Write the DataFrame
to a CSV file



Plotting the Motion Detection Graph

```
from motion_detector import df
from bokeh.plotting import figure, show, output_file
from bokeh.models import HoverTool, ColumnDataSource

df["Start_string"]=df["Start"].dt.strftime("%Y-%m-%d %H:%M:%S")
df["End_string"]=df["End"].dt.strftime("%Y-%m-%d %H:%M:%S")

cds=ColumnDataSource(df)

p=figure(x_axis_type='datetime', height=100, width=500, responsive=True, title="Motion Graph")
p.yaxis.minor_tick_line_color=None
p.ygrid[0].ticker.desired_num_ticks=1

hover=HoverTool(tooltips=[("Start", "@Start_string"), ("End", "@End_string")])
p.add_tools(hover)

q=p.quad(left="Start", right="End", bottom=0, top=1, color="red", source=cds)

output_file("Graph1.html")
show(p)
```

Import the DataFrame from the
motion_detector.py

Convert time to a
string format

The DataFrame of time
values is plotted on the
browser using Bokeh plots



Tutorials

1. Using OpenCV Episode 1 to 16

<https://www.youtube.com/playlist?list=PLQVvvaa0QuDdtJXlLtAJxJetJcgmqlQg>

2. Creating your own Haar Cascade OpenCV Python Tutorial Episode 17 to 21

<https://www.youtube.com/playlist?list=PLQVvvaa0QuDdtJXlLtAJxJetJcgmqlQg>

3. Faster Video FPS

<https://www.pyimagesearch.com/2017/02/06/faster-video-file-fps-with-cv2-videocapture-and-opencv/>

4. Download Haar Cascades

<http://alereiimondo.no-ip.org/OpenCV/34>

5. Face Recognition and Deep Learning

<https://www.pyimagesearch.com/2018/06/18/face-recognition-with-opencv-python-and-deep-learning/>

To fully detect a face

1. Viola-Jones Face Detection (At least 3)

- i. Frontal Face
- ii. Left Eye (Optional)
- iii. Right Eye (Optional)
- iv. Nose (Optional)
- v. Mouth (Optional)