**Play a video using OpenCV**

**OpenCV (Open Source Computer Vision):**

is a computer vision library that contains various functions to perform operations on Images or videos. OpenCV library can be used to perform multiple operations on videos. Let’s see how to play a video using the OpenCV Python.

To capture a video, we need to create a VideoCapture object. VideoCapture have the device index or the name of a video file. Device index is just the number to specify which camera. If we pass 0 then it is for first camera, 1 for second camera so on. We capture the video frame by frame.



**Syntax:**

cv2.VideoCapture(0): Means first camera or webcam.

cv2.VideoCapture(1): Means second camera or webcam.

cv2.VideoCapture("file name.mp4"): Means video file

**Steps to Install OpenCV on Windows**

1. Install Python on your system
2. Install pip
3. Install OpenCV library using pip

After the installation of the Python and pip, we can directly install the OpenCV library and start using them. To install the library, we need to enter the given command in the terminal.

**Getting Started with Videos**

**Goal**

In this Blog, we will learn how to Read, Write and Display a video using OpenCV.

We will learn these functions: cv2.VideoCapture(), cv2.VideoWriter()

## Reading a Video:

In OpenCV, a video can be read either by using the feed from a camera connected to a computer or by reading a video file. The first step towards reading a video file is to create a **VideoCapture** object. Its argument can be either the device index or the name of the video file to be read.

In most cases, only one camera is connected to the system. So, all we do is pass ‘0’ and OpenCV uses the only camera attached to the computer. When more than one camera is connected to the computer, we can select the second camera by passing ‘1’, the third camera by passing ‘2’ and so on.

Create a VideoCapture object and read from input file

If the input is taken from the camera, pass 0 instead of the video file name.

cap = cv2.VideoCapture(file\_name.mp4')

**Code:**

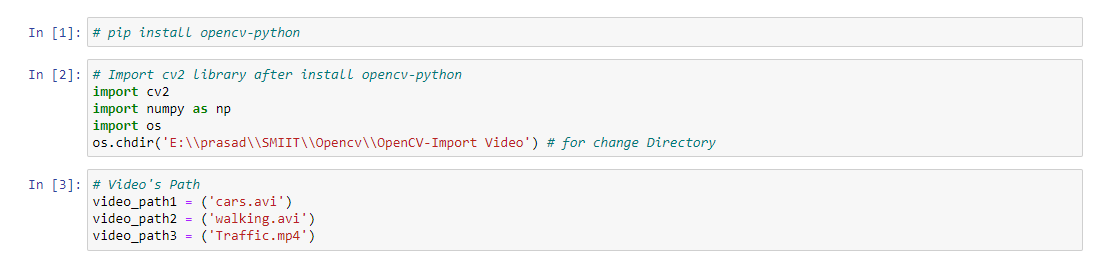
pip install opencv-python

import cv2

import numpy as np

import os

os.chdir('E:\\prasad\\SMIIT\\Opencv\\OpenCV-Import Video')



As per above code first of all we install opencv-python library by using pip install opencv-python command after that we import all important libraries like import cv2, import numpy as np and import os. Os library we install for change the directory for set our located video path using os.chdir() command.

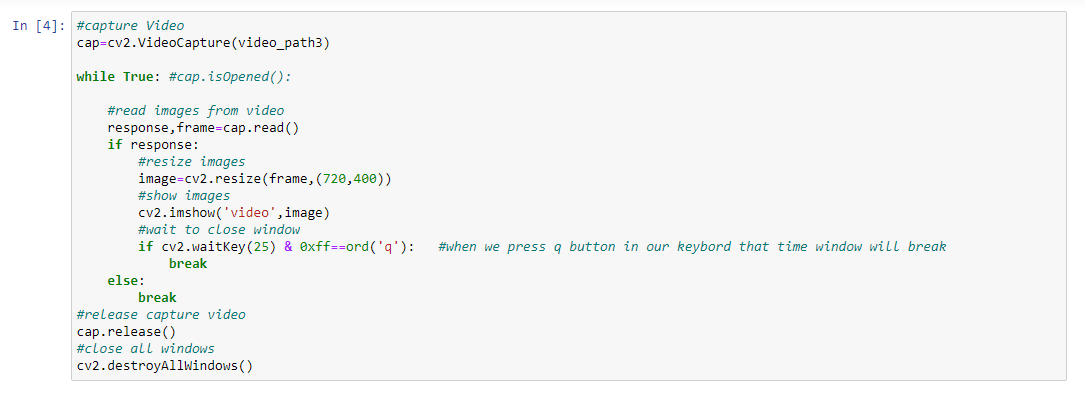
**Video's Path**

video\_path1 = ('cars.avi')

video\_path2 = ('walking.avi')

video\_path3 = ('Traffic.mp4')

Here we import three videos like cars.avi, walking.avi and Traffic.mp4 this 3 videos path store in different 3 variables like video\_path1, video\_path2 and video\_path3.



cap.read() returns a bool (True/False). If frame is read correctly, it will be True. So you can check end of the video by checking this return value.

Sometimes, cap may not have initialized the capture. In that case, this code shows error. You can check whether it is initialized or not by the method cap.isOpened(). If it is True, OK. Otherwise open it using cap.open().

**Displaying a video**

After reading a video file, we can display the video frame by frame. A frame of a video is simply an image and we display each frame the same way we display images, i.e., we use the function cv2.imshow().

It is same as capturing from Camera, just change camera index with video file name. Also while displaying the frame, use appropriate time for cv2.waitKey(). If it is too less, video will be very fast and if it is too high, video will be slow (Well, that is how you can display videos in slow motion). 25 milliseconds will be OK in normal cases.

**Code:**

cap=cv2.VideoCapture(video\_path3)

while True: #cap.isOpened():

#read images from video

response,frame=cap.read()

if response:

#resize images

image=cv2.resize(frame,(720,400))

#show images

cv2.imshow('video',image)

#wait to close window

if cv2.waitKey(25) & 0xff==ord('q'):

#when we press q button in our keybord that time window will break

break

else:

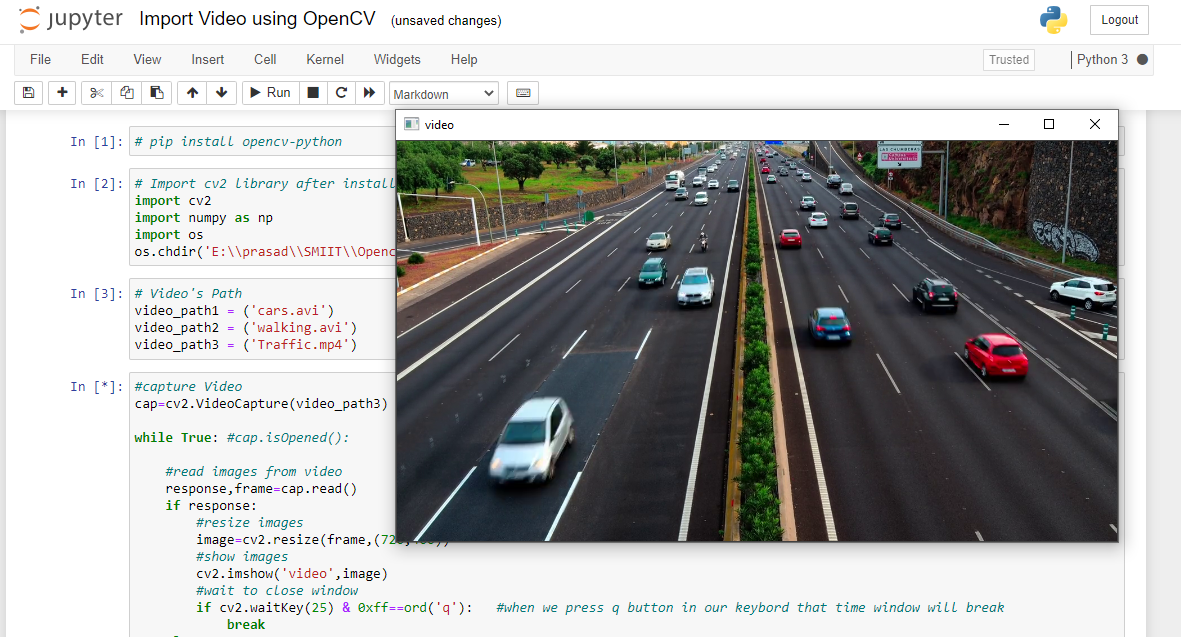
break

#release capture video

cap.release()

#close all windows

cv2.destroyAllWindows()



## Show Video in Multiple Frame

**Code:**

#capture Video

cap=cv2.VideoCapture(video\_path2)

while True: #cap.isOpened()

#read images from video

response,frame=cap.read()

if response:

#resize images

frame=cv2.resize(frame,(450,320))

frame2=np.hstack((frame,frame,frame)) #Create 3 frames horizontaly

frame4=np.vstack((frame2,frame2)) #Create 2 frames Vertical

#show images

cv2.imshow('Car\_Video',frame4)

#wait to close window

if cv2.waitKey(5) & 0xff ==ord('q'): #when we press q button in our keybord that time window will break

break

else:

break

#release capture video

cap.release()

#close all windows

cv2.destroyAllWindows()



**Output:**

****

## Multiple Video in One Frame

**Code:**

# Video's Path

video\_path1 = ('cars.avi')

video\_path2 = ('walking.avi')

video\_path3 = ('Traffic.mp4')

#capture Video

cap1=cv2.VideoCapture(video\_path1)

cap2=cv2.VideoCapture(video\_path2)

cap3=cv2.VideoCapture(video\_path3)

cap4=cv2.VideoCapture(0)

while True: #cap.isOpened()

#read images from video

response1,frame1=cap1.read()

response2,frame2=cap2.read()

response3,frame3=cap3.read()

response4,frame4=cap4.read()

if response1:

#resize images

frame1=cv2.resize(frame1,(400,330))

frame2=cv2.resize(frame2,(400,330))

frame3=cv2.resize(frame3,(400,330))

frame4=cv2.resize(frame4,(400,330))

frame\_2=np.hstack((frame1,frame2)) #Create frames horizontaly

frame\_02=np.hstack((frame3,frame4)) #Create frames horizontaly

frame\_4=np.vstack((frame\_2,frame\_02)) #Create 2 frames Vertical

#show images

cv2.imshow('mult\_video',frame\_4)

#wait to close window

if cv2.waitKey(5) & 0xff ==ord('q'): #when we press q button in our keybord that time window will break

break

else:

break

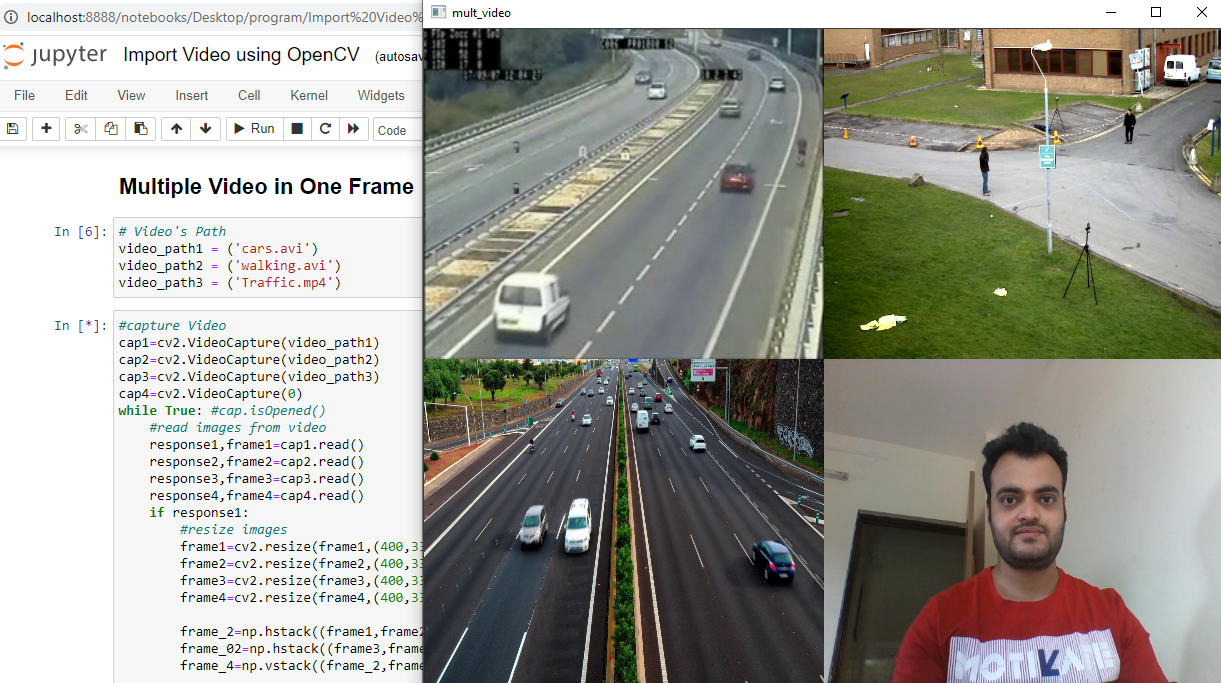
#release capture video

cap4.release()

#close all windows

cv2.destroyAllWindows()

**Output:**

****

**Blog Site:** <https://www.smiit.xyz/opencv-import-videos/>

## GitHub Link: <https://github.com/SMIIT-Projects/OpenCV-Import-Video>

## Tools and Technologies:

**The Code is written in Python 3.8.5.**

## Used libraries:

**opencv-python==4.5.1.48**

**matplotlib==3.4.1**

**numpy==1.20.2**

**pandas==1.2.4**

**scipy==1.6.2**

**seaborn==0.11.1**

Thank You!