we're going to cover some basic operations with video and webcams. Aside from the beginning lines, handling frames from a video is identical to handling for images. Let's show some examples:

import numpy as np

import cv2

cap = cv2.VideoCapture(0)

while(True):

ret, frame = cap.read()

cv2.imshow('frame', frame)

if cv2.waitKey(1) & 0xFF == ord('q'):

break

cap.release()

cv2.destroyAllWindows()

Little bit change now

import numpy as np

import cv2

cap = cv2.VideoCapture(0)

while(True):

ret, frame = cap.read()

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

cv2.imshow('frame',frame)

cv2.imshow('gray',gray)

if cv2.waitKey(1) & 0xFF == ord('q'):

break

cap.release()

cv2.destroyAllWindows()

First, we import numpy and cv2, nothing fancy there. Next, we cay cap = cv2.VideoCapture(0). This will return video from the first webcam on your computer.

while(True):

ret, frame = cap.read()

This code initiates an infinite loop (to be broken later by a break statement), where we have ret and frame being defined as the cap.read(). Basically, ret is a boolean regarding whether or not there was a return at all, at the frame is each frame that is returned. If there is no frame, you wont get an error, you will get None.

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

Here, we define a new variable, gray, as the frame, converted to gray. Notice this says BGR2GRAY. It is important to note that OpenCV reads colors as BGR (Blue Green Red), where most computer applications read as RGB (Red Green Blue). Remember this.

cv2.imshow('frame',gray)

Notice that, despite being a video stream, we still use imshow. Here, we're showing the converted-to-gray feed. If you wish to show both at the same time, you can do imshow for the original frame, and imshow for the gray and two windows will appear.

if cv2.waitKey(1) & 0xFF == ord('q'):

break

This statement just runs once per frame. Basically, if we get a key, and that key is a q, we will exit the while loop with a break, which then runs:

cap.release()

cv2.destroyAllWindows()

This releases the webcam, then closes all of the imshow() windows.

In some cases, you may actually want to record, and save the recording to a new file. Here's an example of doing this on Windows:

import numpy as np

import cv2

cap = cv2.VideoCapture(1)

fourcc = cv2.VideoWriter\_fourcc(\*'XVID')

out = cv2.VideoWriter('output.avi',fourcc, 20.0, (640,480))

while(True):

ret, frame = cap.read()

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

out.write(frame)

cv2.imshow('frame',gray)

if cv2.waitKey(1) & 0xFF == ord('q'):

break

cap.release()

out.release()

cv2.destroyAllWindows()

Mainly to note here is the codec being used, and the output information defined before the while loop. Then, within the while loop, we use out.write() to output the frame. Finally, outside the while loop, after we release our webcam, we also release the out.