

Lesson 3 Basic Operations of OpenCV

1. Image reading and writing

Read the image: `cv2.imread(Location, Model)`

Location——Want to read the image address. The address can be an absolute path or a relative path, but note that the usage of path slashes in different operating systems.

Model——Image loading mode, the first is `cv2.IMREAD_COLOR`, which is used to load a color image, but does not load its own Alpha channel (recording transparency); the second is `cv2.IMREAD_GRAYSCALE`, which is used to load one Grayscale image; the third type is `cv2.IMREAD_UNCHANGED`, which loads the Alpha channel while loading the image.

Display image: `cv2.imshow('Name' ,Pic)`

Name——The name of the window displaying the image.

Pic——The image to be displayed (the image object that has been read in using `cv2.imread()` before).

Example: Create a new py file, and put a picture with the full name "camera.png" in the same folder of the py file, enter the following code, you can see the image displayed after running, press any key to display the image hide.

```
import cv2
a=cv2.imread("camera.png")
cv2.imshow("test",a)
cv2.waitKey()
cv2.destroyAllWindows()
```

Note: The `cv2.waitKey()` function will wait for any key of the keyboard to be pressed, and the `cv2.destroyAllWindows()` function will close all windows.

2. Video reading and writing

Video can be seen as a fast switching image, so video reading and writing can actually be seen as a continuation of image reading and writing.

Camera initialization: `cv2.VideoCapture(Number)`.

Number--The camera number, usually 0, but because our Raspberry Pi camera transmits data through the local area network, it must be filled

in: `http://127.0.0.1:8080/?action=stream?dummy=param.mjpg`

Read camera frame: `Capture.read()`

Capture-refers to the camera object defined previously

```
import cv2
cap=cv2.VideoCapture('http://127.0.0.1:8080/?action=stream?dummy=param.mjpg')
while(cap.isOpened()):
    ret,frame=cap.read()
    cv2.imshow('capture',frame)
    key=cv2.waitKey(1)
    if key & 0xFF==ord('q'):
        break
cap.release()
cv2.destroyAllWindows()
```

Example: Display the camera screen on the desktop, and stop the display when the pressed q key.

Note: `cv2.waitKey(delay)` is to wait for keyboard input, it has the function of refreshing the image in the video, the delay in the brackets

It is the waiting time. After displaying a frame of image, the program waits for "delay" ms before displaying the next frame of the video.