OpenCV in Python

Dr. Zoran Duric

Department of Computer Science
George Mason University

Office: Nguyen Engineering Building 4443
Email: zduric@cs.gmu.edu

URL: http://cs.gmu.edu/∼zduric/
Lab URL: http://cs.gmu.edu/∼vislab/

OpenCV Resources

1. OpenCV documentation etc.
2. OpenCV API Reference
3. OpenCV Safari Books Free for GMU students
4. OpenCV Computer Vision with Python A good source for installation in various OS, code examples, etc.
5. Programming Computer Vision with Python Not OpenCV, but a lot of examples

Read an Image

Use the function *cv2.imread()* to read an image. First argument is the image name. The image should be in the working directory or a full path of image should be given. Second argument is a flag which specifies the way image should be read.

- *cv2.IMREAD_COLOR* : Loads a color image.
- *cv2.IMREAD_GRAYSCALE* : Loads image in grayscale mode
- *cv2.IMREAD_UNCHANGED* : Loads image as such including alpha channel

Note Instead of these three flags, you can simply pass integers 1, 0 or -1 respectively.

See the code below:

```
import numpy as np
import cv2
# Load an color image in grayscale
img = cv2.imread('messi.jpg',0)
```

Display an Image

Use the function *cv2.imshow()* to display an image in a window. The window automatically fits to the image size.
First argument is a window name which is a string. Second argument is our image. You can create as many windows as you wish, but with different window names.

```
cv2.namedWindow('image', cv2.WINDOW_NORMAL) # not required
cv2.imshow('image',img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

cv2.waitKey() is a keyboard binding function. Its argument is the time in milliseconds. 0 – wait indefinitely
cv2.destroyAllWindows() simply destroys all the windows we created. To destroy any specific window, use the function *cv2.destroyWindow()* where you pass the exact window name.

Write an Image

Use the function *cv2.imwrite()* to save an image. First argument is the file name, second argument is the image you want to save.

A simple program using all functions:

```
import numpy as np
import cv2
img = cv2.imread('messi.jpg',0)
cv2.imshow('image',img)
k = cv2.waitKey(0) & 0xFF
if k == 27: # wait for ESC key to exit
    cv2.destroyAllWindows()
elif k == ord('s'): # wait for 's' key to save and exit
    cv2.imwrite('messigray.png',img)
    cv2.destroyAllWindows()
```

In addition, you can use *Matplotlib* to display images.

Capture Video from Camera

```
import numpy as np
import cv2
cap = cv2.VideoCapture(0)
while(True):
    # Capture frame-by-frame
    ret, frame = cap.read()
    # Our operations on the frame come here
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    # Display the resulting frame
    cv2.imshow('frame',gray)
    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
# When everything done, release the capture
cap.release()
cv2.destroyAllWindows()
```

Playing Video from File

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()*. 25 milliseconds will be OK in normal cases.

```
import numpy as np
import cv2
cap = cv2.VideoCapture('/Users/zduric/Desktop/008a014s00R.dv')

while(cap.isOpened()):
    ret, frame = cap.read()
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    cv2.imshow('frame',gray)
    if cv2.waitKey(25) & 0xFF == ord('q'):
        break
cap.release()
cv2.destroyAllWindows()
```

Saving a Video

```
import numpy as np
import cv2
cap = cv2.VideoCapture(0)
size =(int(cap.get(cv2.cv.CV_CAP_PROP_FRAME_WIDTH)),
       int(cap.get(cv2.cv.CV_CAP_PROP_FRAME_HEIGHT)))
# out = cv2.VideoWriter('output.avi',cv2.cv.CV_FOURCC('I','Y','U','V'), 30, size,True)
out = cv2.VideoWriter('output.avi',cv2.cv.CV_FOURCC('X','V','I','D'), 30, size,True)
while(cap.isOpened()):
    ret, frame = cap.read()
    if ret:
        out.write(frame)
        cv2.imshow('frame',frame)
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
    else:
        break
 cap.release()
out.release()
cv2.destroyAllWindows()
```