OpenCV Python Documentation

Release 0.1

gramman

June 16, 2016

Contents

OpenCV-Python Tutorial .

Contents:

CHAPTER 1

1.1 Goal

- , , .
- `cv2.imread()`, `cv2.imshow()`, `cv2.imwrite()` .

1.2

openCV import.:

```
>>> import cv2
```

`cv2.imread()` . /.

```
>>> img = cv2.imread('lena.jpg', cv2.IMREAD_COLOR)
```

`cv2.`**`imread`**(*fileName*, *flag*)
 flag .

Parameters

- **`fileName`** (*`str`*) –
- **`flag`** (*`int`*) – Option.

Returns image

Return type numpy.ndarray

flag 3 .

- `cv2.IMREAD_COLOR` : Color . , Default.
- `cv2.IMREAD_GRAYSCALE` : Grayscale . .
- `cv2.IMREAD_UNCHANGED` : alpha channel .

Note: 3 flag 1, 0, -1 .

img numpy ndarray type. numpy python openCV . img .

```
>>> img.shape
(206, 207, 3)
```

3 return . 206 (Y), 207 (X), 3 . 207 X 206 .

3 . BGR. RGB , openCV B(lue), G(reen), R(ed) .

.

1.3

cv2.imshow() .

```
>>> c22.imshow('image', img)
>>> cv2.waitKey(0)
>>> cv2.destroyAllWindows()
```

cv2.**imshow**(*title*, *image*)
.

Parameters

- **title** (*str*) – Title
- **image** (*numpy.ndarray*) – cv2.imread() return

cv2.waitKey() keyboard 0 key milisecond .

cv2.destroyAllWindows() . 3 .

Sample Code

```
1  import cv2
2
3  fname = 'lena.jpg'
4
5  original = cv2.imread(fname, cv2.IMREAD_COLOR)
6  gray = cv2.imread(fname, cv2.IMREAD_GRAYSCALE)
7  unchange = cv2.imread(fname, cv2.IMREAD_UNCHANGED)
8
9  cv2.imshow('Original', original)
10 cv2.imshow('Gray', gray)
11 cv2.imshow('Unchange', unchange)
12
13 cv2.waitKey(0)
14 cv2.destroyAllWindows()
```

Fig. 1.1: Sample Image

flag .

Fig. 1.2: Original

Fig. 1.3: Grayscale

Fig. 1.4: Unchange

1.4

cv2.imwrite() .

```
>>> cv2.imwrite('lenagray.png', gray)
```

cv2.**imwrite**(*fileName*, *image*)
 image .

Parameters

- **fileName** (*str*) –
- **image** –

Sample Code

esc , 's' key grayscale Sample. cv2.waitKey() .:

```
import cv2

img = cv2.imread('lena.jpg', cv2.IMREAD_GRAYSCALE)
cv2.imshow('image',img)
k = cv2.waitKey(0)
if k == 27: # esc key
    cv2.destroyAllWindow()
elif k = ord('s'): # 's' key
    cv2.imwrite('lenagray.png',img)
    cv2.destroyAllWindow()
```

Warning: 64bit OS k = cv2.waitKey(0) & 0xFF bit .

1.5 Matplotlib

Matplotlib plot Python Plot Library. zoom .

Sample Code

```
1  #-*- coding:utf-8 -*-
2  import cv2
3  from matplotlib import pyplot as plt # as alias
4
5  img = cv2.imread('lena.jpg', cv2.IMREAD_COLOR)
6
7  plt.imshow(img)
8  plt.xticks([]) # x
9  plt.yticks([]) # y
10 plt.show()
```

Result

. , .

openCV BGR , Matplotlib RGB .

3 .

Sample .

Sample Code

```
1  #-*- coding:utf-8 -*-
2  import cv2
3  from matplotlib import pyplot as plt # as alias
```

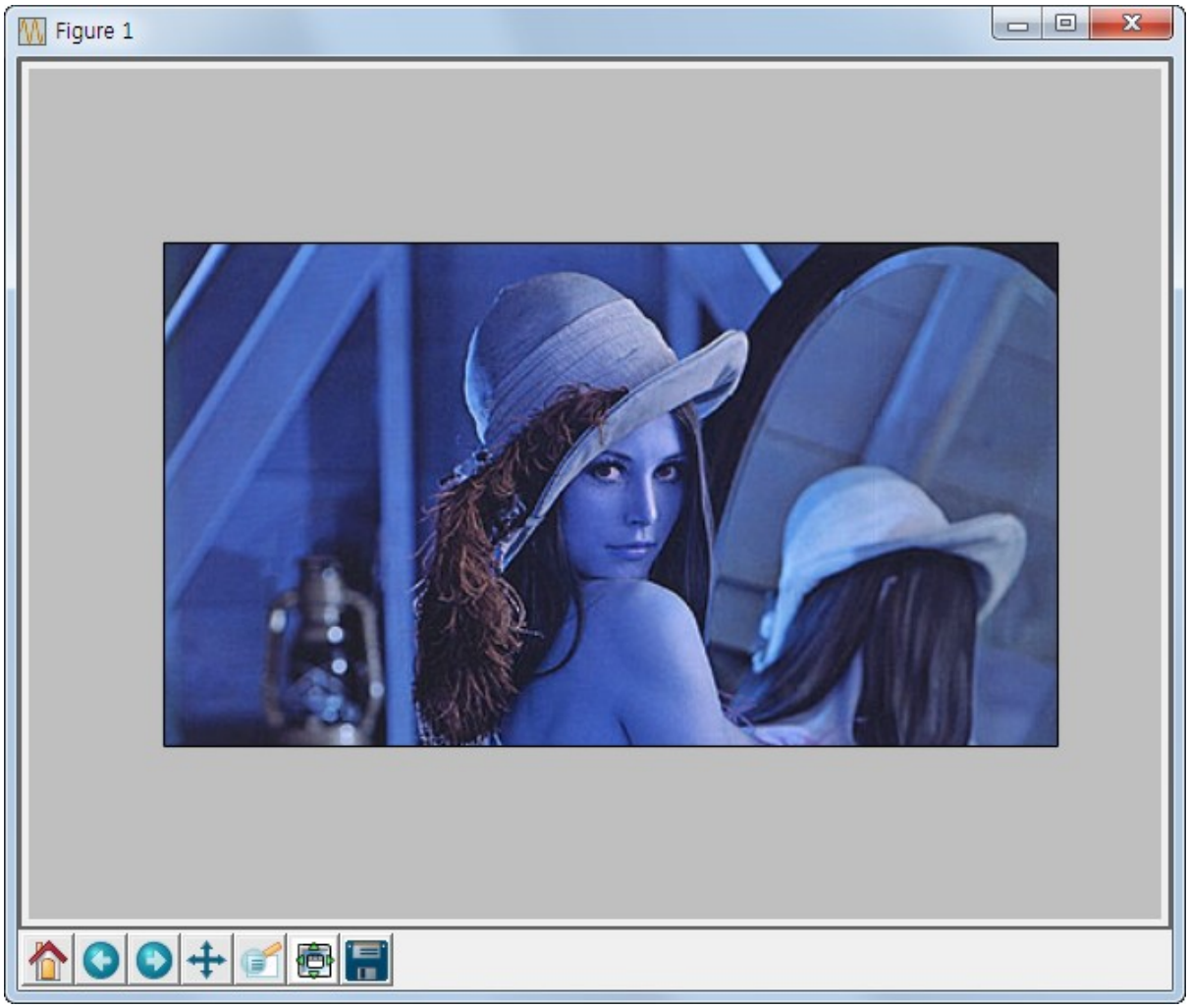


Fig. 1.5: Matplotlib Result

```
4
5  img = cv2.imread('lena.jpg', cv2.IMREAD_COLOR)
6
7  b, g, r = cv2.split(img)    # img b,g,r
8  img2 = cv2.merge([r,g,b]) # b, r  Merge
9
10 plt.imshow(img2)
11 plt.xticks([]) # x
12 plt.yticks([]) # y
13 plt.show()
```

Result

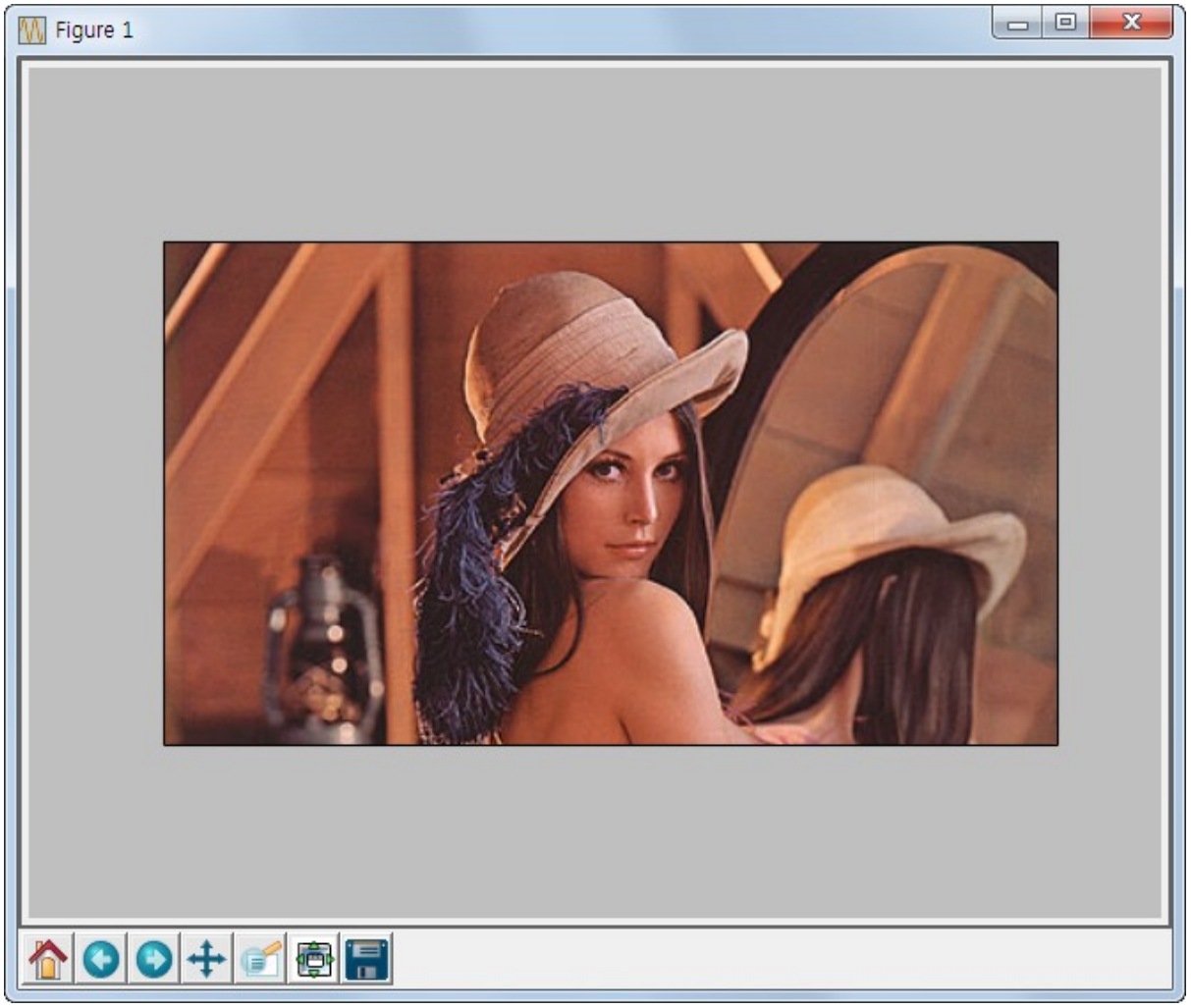


Fig. 1.6: RGB

2.1 Goal

- , , .
- `cv2.VideoCapure()` ,`cv2.VideoWriter()` .

2.2 Camera

Camera , .

- VideoCapture Object . camera device index . 0 Camera .
- Loop frame .
- frame , .
- , VideoCapure Object release window .

grayscale .

Sample Code

```
# -*-coding: utf-8 -*-
import cv2

# cap   open    cap.isOpen()
cap = cv2.VideoCapture(0)

# cap.get(prodId)/cap.set(propId, value)    .
# 3 width, 4 heigh

print 'width: {0}, height: {1}'.format(cap.get(3),cap.get(4))
cap.set(3,320)
cap.set(4,240)

while(True):
    # ret : frame capture(boolean)
    # frame : Capture frame
    ret, frame = cap.read()

    if (ret):
        # image Grayscale Convert.
        gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

        cv2.imshow('frame', gray)
        if cv2.waitKey(1) & 0xFF == ord('q'):
            break
```

```

cap.release()
cv2.destroyAllWindows()
```

2.3 File

File Camera .

Sample Code

```
import cv2

cap = cv2.VideoCapture('vtest.avi')

while(cap.isOpened()):
    ret, frame = cap.read()
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    cv2.imshow('frame',gray)

    if cv2.waitKey(1) & 0xFF == ord('q'):
        break
cap.release()
cv2.destroyAllWindows()
```

Note: Codec .

2.4

`cv2.VideoWriter` Object .

`cv2.`**`VideoWriter`**(*outputFile*, *fourcc*, *frame*, *size*)
Object

Parameters

- **`outputFile`** (*`str`*) –
- **`fourcc`** – Codec. cv2.VideoWriter_fourcc()
- **`frame`** (*`float`*) – frame
- **`size`** (*`list`*) – (ex; 640, 480)

fourcc `cv2.VideoWriter_fourcc('M','J','P','G')` `cv2.VideoWriter_fourcc(*'MJPG)` . OS codec .(Windows DIVX)

Sample Code

```
# -*-coding: utf-8 -*-

import cv2

cap = cv2.VideoCapture(0)

fourcc = cv2.VideoWriter_fourcc(*'DIVX')
out = cv2.ViewoWriter('output.avi', fourcc, 25.0, (640,480))

while (cap.isOpend()):
    ret, frame = cap.read()

```

```
    if ret:
        #  ,  0:, 1 :
        frame = cv2.flip(frame, 0)

        out.write(frame)

        cv2.imshow('frame', frame)

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

cap.release()
out.release()
cv2.destroyAllWindows()
```

3.1 Goal

- .
- `cv2.line()`, `cv2.circle()`, `cv2.rectangle()`, `cv2.putText()` .

Match .

3.2 Line

Start End .

`cv2.`**`line`**(*img*, *start*, *end*, *color*, *thickness*)

Parameters

- **`img`** –
- **`start`** – (ex; (0,0))
- **`end`** – (ex; (500. 500))
- **`color`** – BGR Color(ex; (255, 0, 0) -> Blue)
- **`thickness`** (*int*) – . pixel

Sample Code

```
import numpy as np
import cv2

# 0    Canvas()
img = np.zeros((512, 512, 3), np.uint8)
img = cv2.line(img, (0, 0), (511, 511), (255, 0, 0), 5)

cv2.imshow('image',img)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

3.3

top-left corner bottom-right corner .

`cv2.`**`rectangle`**(*img*, *start*, *end*, *color*, *thickness*)

Parameters

- **img** –
- **start** – (ex; (0,0))
- **end** – (ex; (500. 500))
- **color** – BGR Color(ex; (255, 0, 0) -> Blue)
- **thickness** (*int*) – . pixel

Sample Code

```
img = cv2.rectangle(img, (384, 0), (510, 128), (0,255,0), 3)
```

3.4

cv2.**circle**(*img*, *center*, *radian*, *color*, *thickness*)

Parameters

- **img** –
- **center** – (x, y)
- **radian** –
- **color** – BGR Color
- **thickness** – , -1

Sample Code

```
img = cv2.circle(img, (447,63), 63, (0,0,255), -1)
```

3.5

cv2.ellipse(img, center, axes, angle, startAngle, endAngle, color[, thickness[, lineType

Parameters

- **img** – image
- **center** –
- **axes** –
- **angle** –
- **startAngle** –
- **endAngle** –
- **color** –
- **thickness** – -1

Sample Code

```
img = cv2.ellipse(img, (256,256), (100,50), 0, 0, 180, 255, -1)
```

3.6 Polygon

cv2.**polylines**(*img*, *pts*, *isClosed*, *color*, *thickness*)

Parameters

- **img** – image
- **pts** (*array*) –
- **isClosed** –
- **color** – Color
- **thickness** –

Sample Code

```
pts = np.array([[10,5], [20,30], [70,20], [50,10]], np.int32) #   2
#    3  .     .
# -1     .
pts = pts.reshape((-1, 1, 2))
img = cv2.polylines(img, [pts], True, (0,255,255))
```

3.7 Text

cv2.**putText**(*img*, *text*, *org*, *font*, *fontSacle*, *color*)

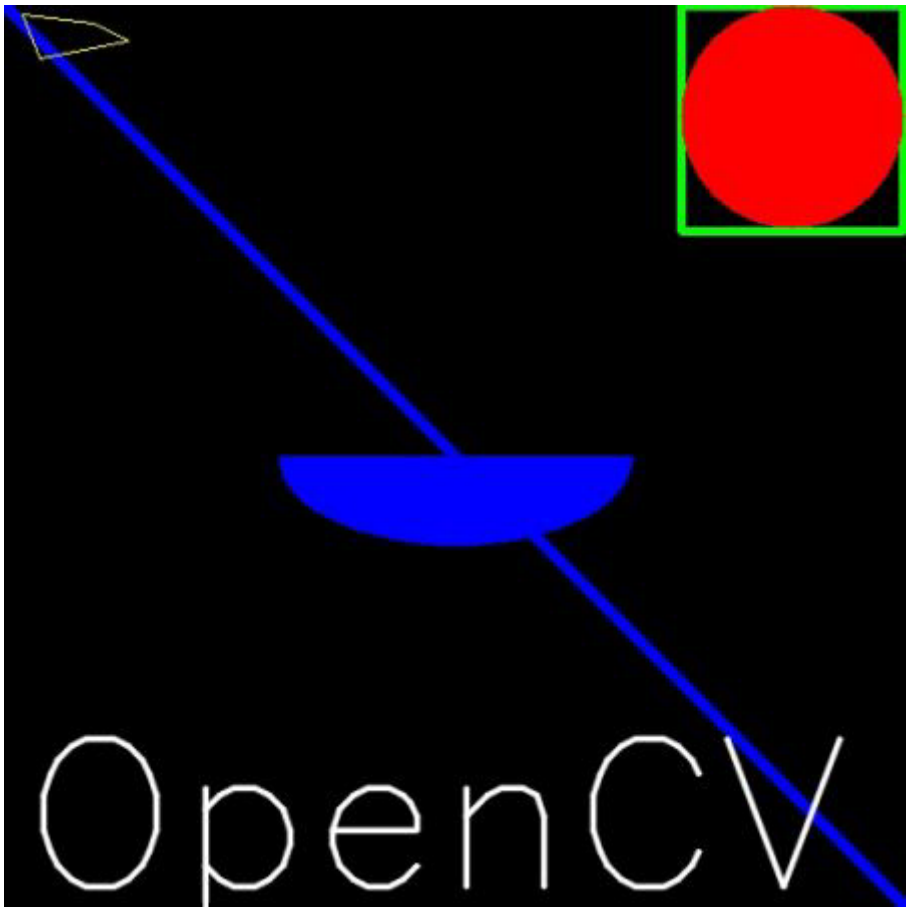
Parameters

- **img** – image
- **text** –
- **org** – . bottom-left corner
- **font** – font type. CV2.FONT_XXX
- **fontSacle** – Font Size
- **color** – fond color

Sample Code

```
cv2.putText(img, 'OpenCV', (10,500), cv2.FONT_HERSHEY_SIMPLEX, 4, (255,255,255), 2)
```

Sample Code .


OpenCV

CHAPTER 4

Mouse

4.1 Goal

- Mouse Event .
- `cv2.setMouseCallback()` .

4.2

OpenCV Mouse Event . Python Terminal .

```
>>> import cv2
>>> events = [i for i in dir(cv2) if 'EVENT' in i]
>>> print events
```

Mouse Event . Event .:

```
'EVENT_FLAG_ALTKEY', 'EVENT_FLAG_CTRLKEY', 'EVENT_FLAG_LBUTTON', 'EVENT_FLAG_MBUTTON', 'EVENT_FLAG
```

Mouse Event Callback `cv2.setMouseCallback()` .

`cv2.`**`setMouseCallback`**(*windowName*, *callback*, *param=None*)

Parameters

- **`windowName`** – windowName
- **`callback`** – callback. callback (event, x, y, flags, param) .
- **`param`** – callback Data

4.3 Demo

Demo Double-Click .

```
1  import cv2
2  import numpy as np
3
4  # callback
5  def draw_circle(event, x, y, flags, param):
6      if event == cv2.EVENT_LBUTTONDBLCLK:
7          cv2.circle(img,(x,y), 100,(255,0,0),-1)
8
9  #  Image
10 img = np.zeros((512,512,3), np.uint8)
11 cv2.namedWindow('image')
```

```
cv2.setMouseCallback('image', draw_circle)

while(1):
    cv2.imshow('image', img)
    if cv2.waitKey(0) & 0xFF == 27:
        break

cv2.destroyAllWindows()
```

4.4 Advanced Demo

Demo. Segmentaion .(ex;)

```
#-*- coding:utf-8 -*-
import cv2
import numpy as np

drawing = False #Mouse
mode = True # True , false
ix,iy = -1,-1


# Mouse Callback
def draw_circle(event, x,y, flags, param):
        global ix,iy, drawing, mode

        if event == cv2.EVENT_LBUTTONDOWN: #
                drawing = True
                ix, iy = x,y
        elif event == cv2.EVENT_MOUSEMOVE: #
                if drawing == True:             #
                        if mode == True:
                                cv2.rectangle(img,(ix,iy),(x,y),(255,0,0),-1)
                        else:
                                cv2.circle(img,(x,y),5,(0,255,0),-1)

        elif event == cv2.EVENT_LBUTTONUP:
                drawing = False;              #
                if mode == True:
                        cv2.rectangle(img,(ix,iy),(x,y),(255,0,0),-1)
                else:
                        cv2.circle(img,(x,y),5,(0,255,0),-1)


img = np.zeros((512,512,3), np.uint8)
cv2.namedWindow('image')
cv2.setMouseCallback('image',draw_circle)

while True:
        cv2.imshow('image', img)

        k = cv2.waitKey(1) & 0xFF

        if k == ord('m'):     # ,  Mode
                mode = not mode
        elif k == 27:         # esc
                break

cv2.destroyAllWindows()

```

CHAPTER 5

Trackbar

5.1 Goal

- trackbar OpenCV .
- `cv2.getTrackbarPos()`, `cv2.createTrackbar()` .

5.2 Demo

Trackbar Demo . Demo 4 Tranckbar . 3 RGB , .

Demo `cv2.getTrackbarPos()`, `cv2.createTrackbar()` .

`cv2.`**`createTrackbar`**(*trackbarName*, *windowName*, *value*, *count*, *onChange*)

Parameters

- **`trackbarName`** – trackbar Name
- **`windowName`** – Named Window
- **`value`** (*int*) – Trackbar
- **`count`** – Tracbar Max. Min 0
- **`onChange`** – Slide Callback. Paramter trackbar Position

`cv2.`**`getTrackbarPos`**(*trackbarName*, *windowName*)

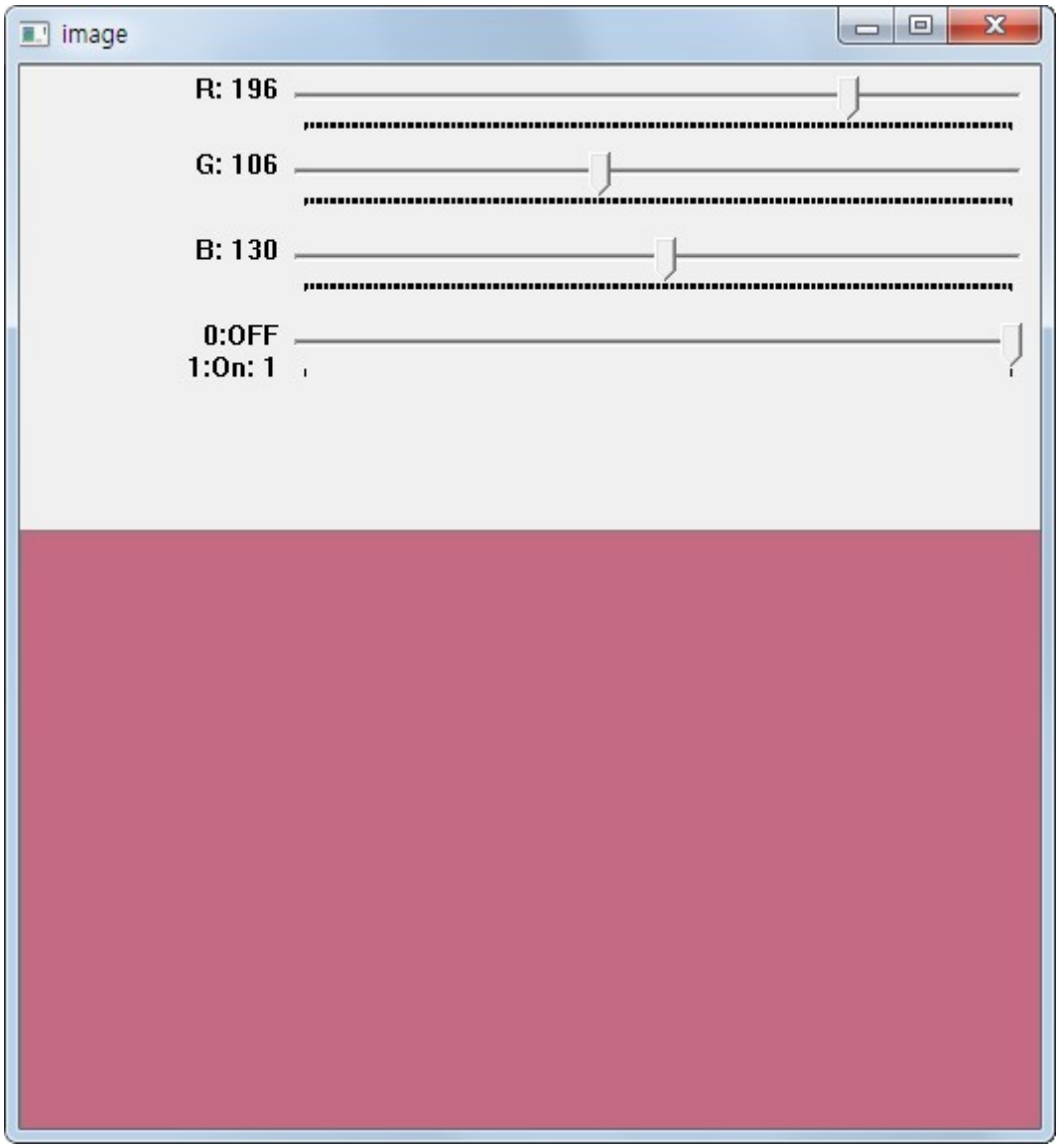
Parameters

- **`trackbarName`** – trackbar Name
- **`windowName`** – Trackbar Named Window

Sample Code

```
#-*- coding: utf-8 -*-
import cv2
import numpy as np

def nothing(x):
    pass

img = np.zeros((300,512,3), np.uint8)
cv2.namedWindow('image')

# trackbar  named window
cv2.createTrackbar('R', 'image', 0, 255, nothing)
cv2.createTrackbar('G', 'image', 0, 255, nothing)
```

```
cv2.createTrackbar('B', 'image', 0, 255, nothing)

switch = '0:OFF\n1:On'
cv2.createTrackbar(switch, 'image', 1, 1, nothing)

while(1):
    cv2.imshow('image', img)

    if cv2.waitKey(1) & 0xFF == 27:
        break

    r = cv2.getTrackbarPos('R','image')
    g = cv2.getTrackbarPos('G', 'image')
    b = cv2.getTrackbarPos('B', 'image')
    s = cv2.getTrackbarPos(switch, 'image')

    if s == 0:
        img[:] = 0 #  /   0 .
    else:
        img[:] = [b,g,r] #  /  [b,g,r]

cv2.destroyAllWindows()
```



CHAPTER 6

Indices and tables

- genindex
- modindex
- search

C