4. OpenCV image quality

4.1 Function method: cv2.imwrite('yahboomTest.jpg', img, [cv2.IMWRITE_JPEG_QUALITY, 50])

Parameter meaning:

cv2.CV_IMWRITE_JPEG_QUALITY: Set the image quality of the image format to .jpeg or .jpg, the value is 0---100 (the larger the value, the higher the quality), the default is 95

cv2.CV_IMWRITE_WEBP_QUALITY: Set the image quality of the image format to .webp format, the value is 0--100

cv2.CV_IMWRITE_PNG_COMPRESSION: Set the compression ratio of the .png format, the value is 0--9 (the larger the value, the greater the compression ratio), the default is 3

4.2. Actual effect display

Code path:

/home/pi/DOGZILLA_Lite_class/4.Open Source CV/A.introduction/Introduction_to_OpenCV/04_OpenCV_Img_Quality.ipynb

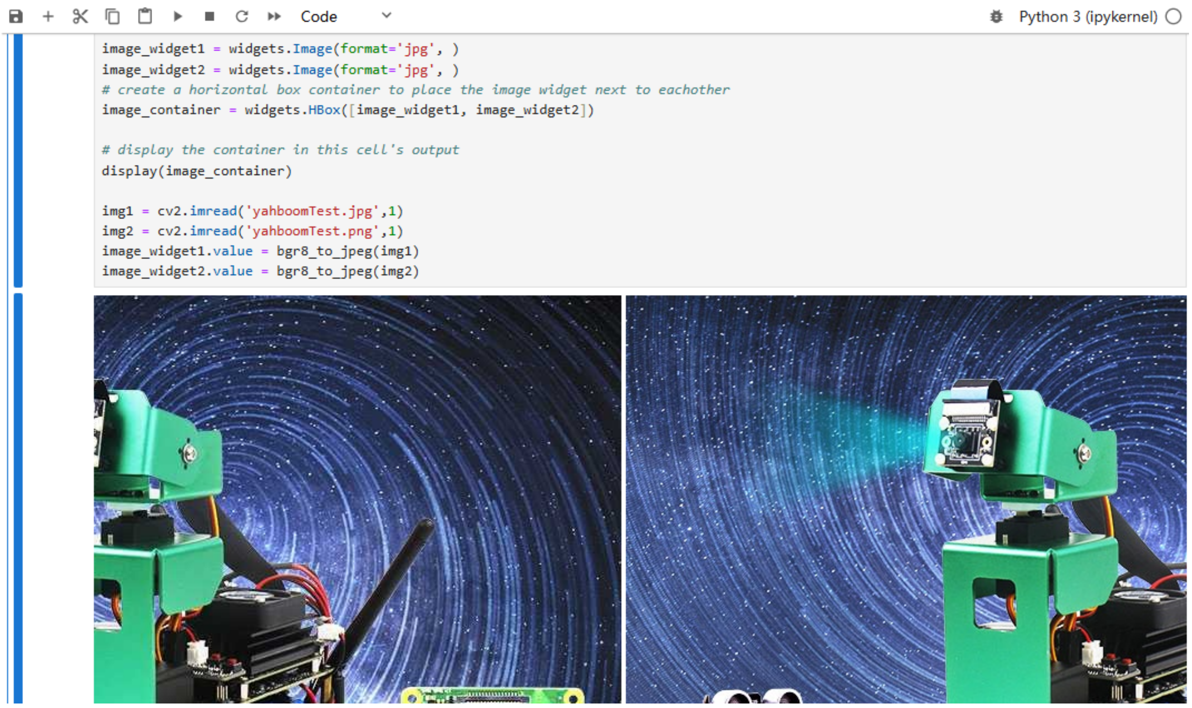
```
import cv2
img = cv2.imread('yahboom.jpg',1)
cv2.imwrite('yahboomTest.jpg', img, [cv2.IMWRITE_JPEG_QUALITY, 50])
#1M 100k 10k 0-100 lossy compression
```

```
# 1 Lossless 2 Transparency attribute
import cv2
img = cv2.imread('yahboom.jpg',1)
cv2.imwrite('yahboomTest.png', img, [cv2.IMWRITE_PNG_COMPRESSION,0])
# jpg 0 High compression ratio 0-100 png 0 Low compression ratio 0-9
```

```
#bgr8 to jpeg format
import enum
import cv2

def bgr8_to_jpeg(value, quality=75):
    return bytes(cv2.imencode('.jpg', value)[1])
```

```
import ipywidgets.widgets as widgets

image_widget1 = widgets.Image(format='jpg', )
image_widget2 = widgets.Image(format='jpg', )
# create a horizontal box container to place the image widget next to eachother
image_container = widgets.HBox([image_widget1, image_widget2])
```

```
# display the container in this cell's output
display(image_container)

img1 = cv2.imread('yahboomTest.jpg',1)
img2 = cv2.imread('yahboomTest.png',1)
image_widget1.value = bgr8_to_jpeg(img1)
image_widget2.value = bgr8_to_jpeg(img2)
```



```
image_widget1 = widgets.Image(format='jpg', )
image_widget2 = widgets.Image(format='jpg', )
# create a horizontal box container to place the image widget next to eachother
image_container = widgets.HBox([image_widget1, image_widget2])

# display the container in this cell's output
display(image_container)

img1 = cv2.imread('yahboomTest.jpg',1)
img2 = cv2.imread('yahboomTest.png',1)
image_widget1.value = bgr8_to_jpeg(img1)
image_widget2.value = bgr8_to_jpeg(img2)
```