4. Image mirroring

4.1. Image mirroring

The principle of image mirroring

There are two types of image mirroring transformations: horizontal mirroring and vertical mirroring. Horizontal mirroring uses the vertical center line of the image as the axis to swap the pixels of the image, that is, swap the left and right halves of the image. Vertical mirroring uses the horizontal center line of the image as the axis to swap the upper and lower halves of the image.

Transformation principle:

Let the width of the image be width and the length be height. (x, y) are the transformed coordinates, (x0, y0) are the coordinates of the original image

Horizontal mirror transformation

Forward mapping: x=width-x0-1, y=y0

Backward mapping: x0=width-x-1, y0=y

Vertical mirror transformation

Upward mapping: x=x0, y=height-y0-1

Downward mapping: x0=x, y0=height-y-1

Summary:

During horizontal mirror transformation, the entire image is traversed, and then each pixel is processed according to the mapping relationship. In fact, horizontal mirror transformation is to swap the image coordinate column to the right and the right column to the left, which can be transformed in columns. The same is true for vertical mirror transformation, which can be transformed in rows.

4.2. Actual effect display

Let's take vertical transformation as an example to see how Python is written:

Code path:

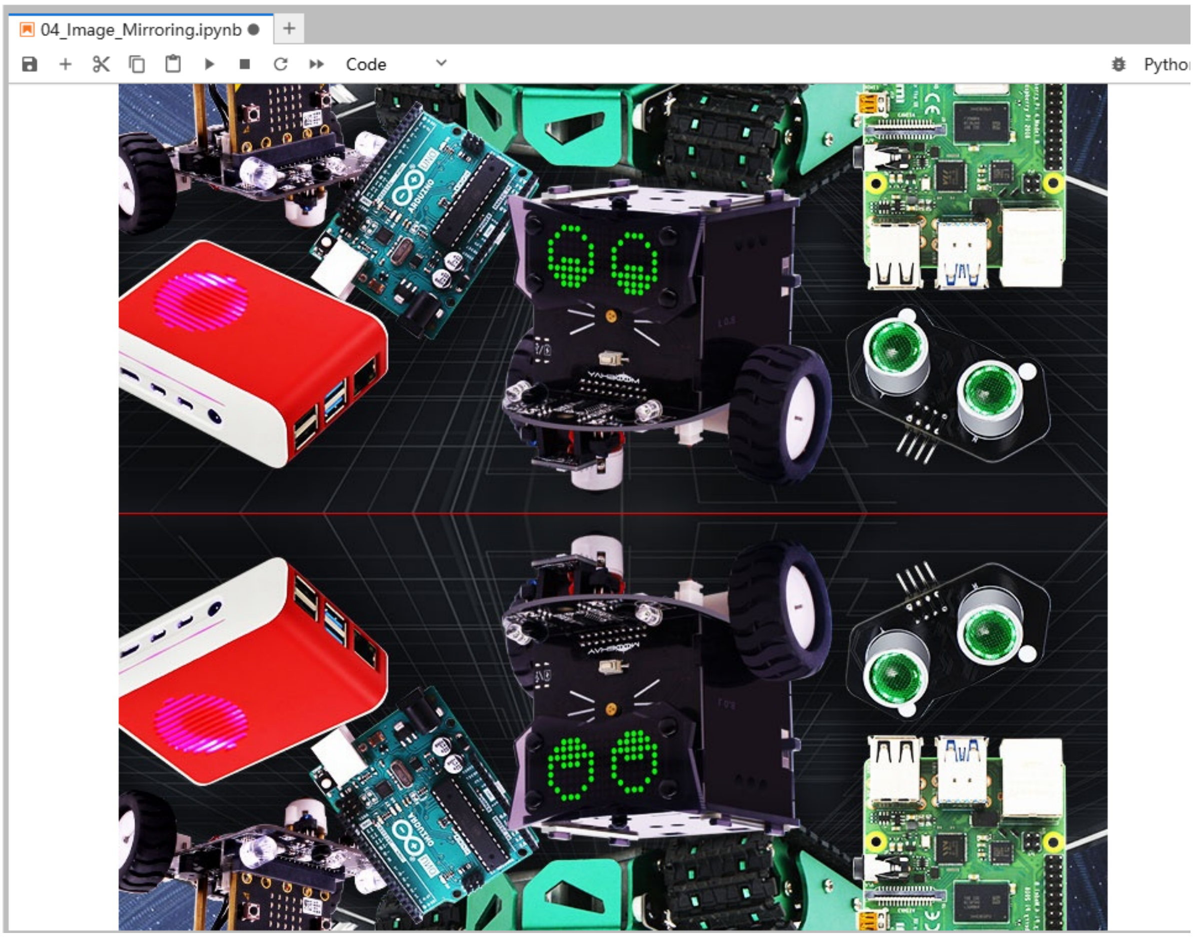
/home/pi/Rider-pi_class/4.Open Source CV/B.Geometric_Transformations/04_Image_Mirroring.ipynb

```
import cv2
import numpy as np
img = cv2.imread('yahboom.jpg',1)
#cv2.imshow('src',img)
imgInfo = img.shape
```

```python
height = imgInfo[0]
width = imgInfo[1]
deep = imgInfo[2]
newImgInfo = (height*2,width,deep)
dst = np.zeros(newImgInfo,np.uint8)#uint8
for i in range(0,height):
    for j in range(0,width):
        dst[i,j] = img[i,j]
        #x y = 2*h - y -1
        dst[height*2-i-1,j] = img[i,j]
for i in range(0,width):
    dst[height,i] = (0,0,255)#BGR
#cv2.imshow('dst',dst)
#cv2.waitKey(0)
```

```python
#bgr8转jpeg格式 bgr8 to jpeg format
import enum
import cv2

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

```python
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_widget1)
#display(image_widget2)

image_widget1.value = bgr8_to_jpeg(dst)
```


04_Image_Mirroring.ipynb
Code
Pytho