

#### 4. Picture mirroring

There are two types of picture mirroring transformation: horizontal mirror image and vertical mirror image.

Horizontal mirroring uses the vertical center line of the picture as the axis to swap the pixels of the picture, that is, to swap the left half and the right half of the picture.

Vertical mirroring takes the horizontal center line of the image as the axis, and swaps the upper half of the image with the lower part.

Transformation principle:

Set the width of the image be width and the length be height.  $(x, y)$  is the transformed coordinates,  $(x_0, y_0)$  is the coordinates of the original picture.

##### Horizontal mirror transformation

$$\begin{aligned}x &= \text{width} - x_0 - 1 \\ y &= y_0\end{aligned}$$

Forward mapping

##### Inverse transformation

$$\begin{aligned}x_0 &= \text{width} - x - 1 \\ y_0 &= y\end{aligned}$$

Backward mapping

##### Vertical mirror transformation

$$\begin{aligned}x &= x_0 \\ y &= \text{height} - y_0 - 1\end{aligned}$$

##### Inverse transformation

$$\begin{aligned}x_0 &= x \\ y_0 &= \text{height} - y - 1\end{aligned}$$

We take the vertical transformation as an example.

Code path:

[/home/pi/Yahboom\\_Project/1.OpenCV\\_course/02Geometric\\_transformation/04\\_Picture\\_mirroring.ipynb](/home/pi/Yahboom_Project/1.OpenCV_course/02Geometric_transformation/04_Picture_mirroring.ipynb)

```
import cv2
```

```
import numpy as np

img = cv2.imread('yahboom.jpg',1)

#cv2.imshow('src',img)

imgInfo = img.shape

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
```

```
#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_widget1)

#display(image_widget2)

image_widget1.value = bgr8_to_jpeg(dst)
```

As shown below.

