

### 1.2.1 Picture Zoom

In OpenCV, the function to achieve picture zoom:

`cv2.resize(InputArray src, OutputArray dst, Size, fx, fy, interpolation)`

*Path:*

[\*/home/jetson/Dofbot\4.opencv\2.Geometric\\_transformation\01\\_picture\\_zoom.ipynb\*](#)

About parameter:

InputArray src	Input picture
OutputArray dst	Output picture
Size	Size of output picture
fx, fy	Zoom factor along x-axis and y-axis
interpolation	Interpolation method

The interpolation method used by the interpolation option:

INTER_NEAREST	Nearest neighbor interpolation
INTER_LINEAR	Bilinear interpolation(default setting)
INTER_AREA	Using the pixel region relationship to complete to re-sampling
INTER_CUBIC	Bicubic interpolation of 4x4 pixel neighborhood
INTER_LANCZOS4	Lanczos interpolation of 8x8 pixel neighborhood

Note:

1.Size format: (width, height)

2.The default interpolation method: bilinear interpolation

```
# 1 load 2 info 3 resize 4 check
import cv2
import matplotlib.pyplot as plt    #2D drawing library for Python

# Read the original picture
img = cv2.imread('yahboom.jpg')
# Print out the picture size
print(img.shape)
# Assign the image height and width to x and y respectively
x, y = img.shape[0:2]

# Display original picture
```

```

# cv.imshow('OriginalPicture', img)

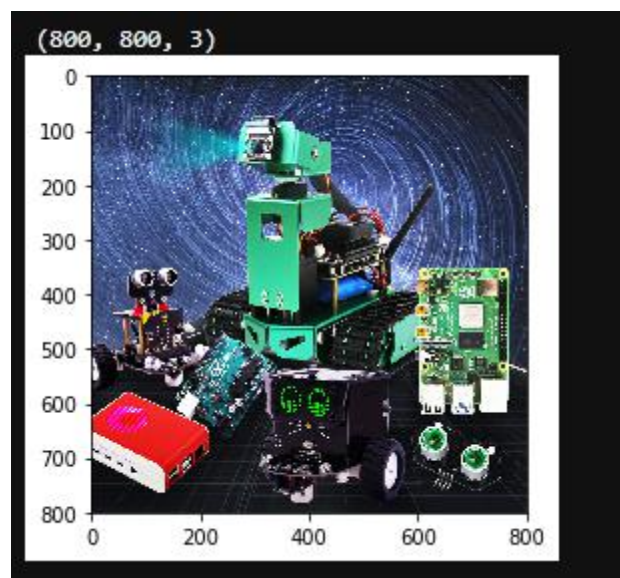
# Zoom to the original half, the output size format is (width, height)
img_test1 = cv2.resize(img, (int(y / 2), int(x / 2)))
# cv2.imshow('resize0', img_test1)
# cv2.waitKey()

# Nearest neighbor interpolation to zoom
# Zoom to the original quarter
img_test2 = cv2.resize(img, (0, 0), fx=0.25, fy=0.25,
interpolation=cv2.INTER_NEAREST)
# cv.imshow('resize1', img_test2)
# cv.waitKey()
# cv.destroyAllWindows()
img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
dst1 = cv2.cvtColor(img_test1, cv2.COLOR_BGR2RGB)
dst2 = cv2.cvtColor(img_test2, cv2.COLOR_BGR2RGB)

# Display original picture
plt.imshow(img)
plt.show()

```

After the program is executed, we can see that the image is 800 \* 800. As shown below.



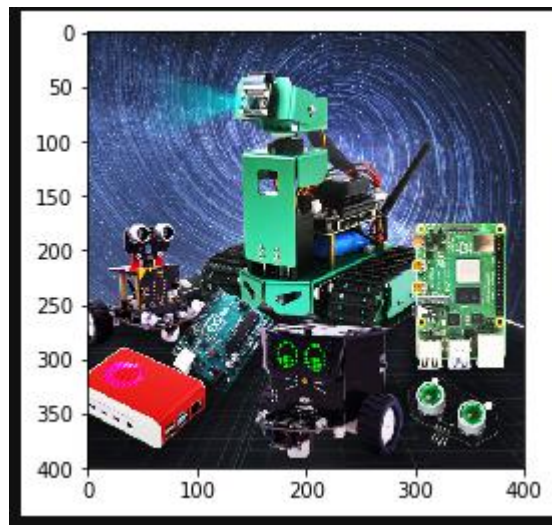
```

# Display zoom 1/2
plt.imshow(dst1)
plt.show()

```

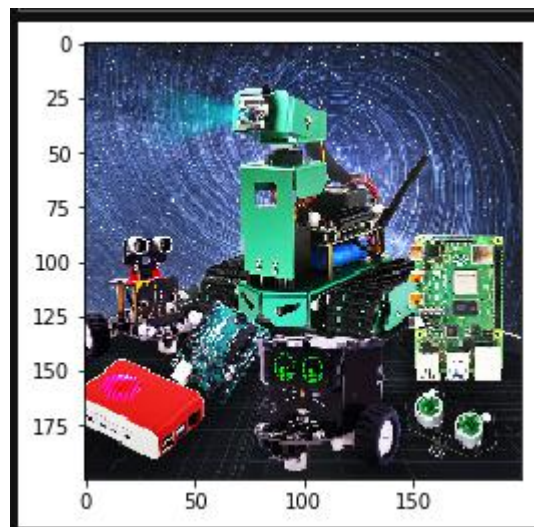
After the program is executed, we can see that the image is 400 \* 400. zoom 1/2 As

shown below.



```
# Display zoom 1/4 Nearest neighbor interpolation to zoom
plt.imshow(dst2)
plt.show()
```

After the program is executed, we can see that the image is 200 \* 200, zoom 1/4. As shown below.



Next, we will introduce matplotlib: 2D plotting library for Python  
Tutorial: <https://www.runoob.com/numpy/numpy-matplotlib.html>

```
import numpy as np
from matplotlib import pyplot as plt

x = np.arange(1,11)
y = 2 * x + 5
plt.title("Matplotlib demo")
plt.xlabel("x axis caption")
plt.ylabel("y axis caption")
```

```
plt.plot(x,y)  
plt.show()
```

