

1.4.6 Gaussian mean filtering

Image processing-filtering

Filtering: It is a basic operation in signal and image processing. The purpose is to selectively extract certain information in the image that is considered important according to the application environment. By filtering, we can remove noise in the image, extract some visual features, allow image re-sampling, and so on.

Frequency domain analysis: The image is divided into different parts from low frequency to high frequency. Low frequencies correspond to areas with small changes in image intensity. High frequencies correspond to areas with very large changes in image intensity.

In the framework of frequency analysis, filtering is used to enhance a certain band or frequency in an image and block (or reduce) other frequency bands.

The low-pass filter eliminates the high-frequency part of the image and keeps the low-frequency part.

The high-pass filter eliminates the low-frequency part of the image and retains the high-frequency part.

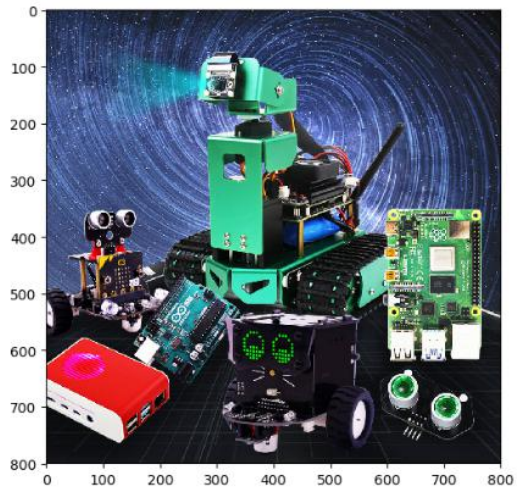
Code path:

[/home/pi/Yahboom_Project/1.OpenCV_course/04image_beautification/06_Gaussian_mean_filtering.ipynb](#)

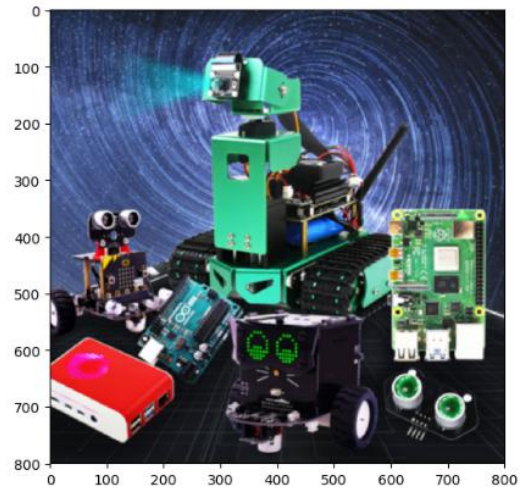
```
import cv2
import numpy as np
import matplotlib.pyplot as plt

img = cv2.imread('yahboom.jpg',1)
# cv2.imshow('src',img)
dst = cv2.GaussianBlur(img,(5,5),1.5)
# cv2.imshow('dst',dst)
# cv2.waitKey(0)
img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
dst = cv2.cvtColor(dst, cv2.COLOR_BGR2RGB)
plt.figure(figsize=(14, 6), dpi=100) # Set the size and pixels of the drawing area
plt.subplot(121) # The first in a row and two columns
plt.imshow(img)
plt.subplot(122) # The second in a row and two columns
plt.imshow(dst)
plt.show()
```

After running the above program, two pictures will be displayed in the jupyterLab control interface, as shown below.



[Original picture]



[Gaussian mean filtering picture]