1.4.5 Skin whitening

Picture whitening formula:

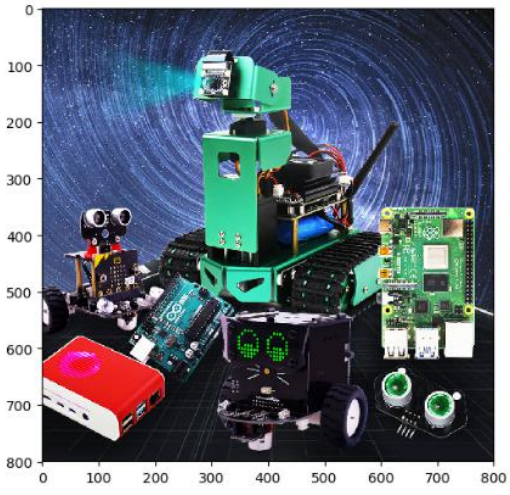
p = P*1.4(a)+ b

Code:

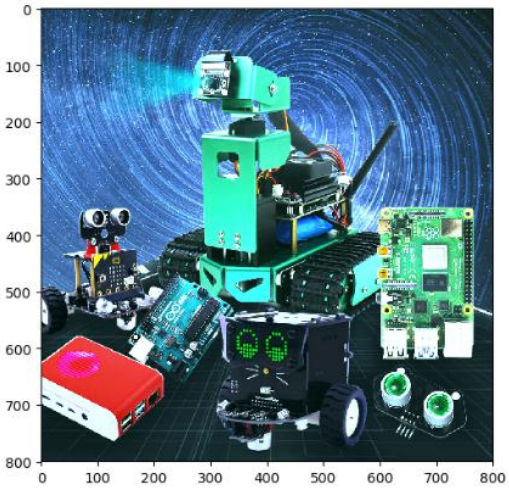
```
import cv2

import numpy as np

import matplotlib.pyplot as plt



img = cv2.imread('image0.jpg',1)

imgInfo = img.shape

height = imgInfo[0]

width = imgInfo[1]

#cv2.imshow('src',img)

dst = np.zeros((height,width,3),np.uint8)

for i in range(0,height):

    for j in range(0,width):

        (b,g,r) = img[i,j]

        bb = int(b*1.3) + 10

        gg = int(g*1.2) + 15



        if bb>255:

            bb = 255
```

```
            if gg>255:
                gg = 255


            dst[i,j] = (bb,gg,r)
# 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]

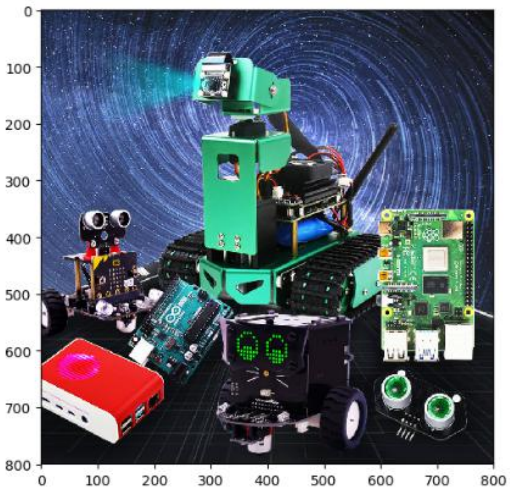


[Skin whitening picture]

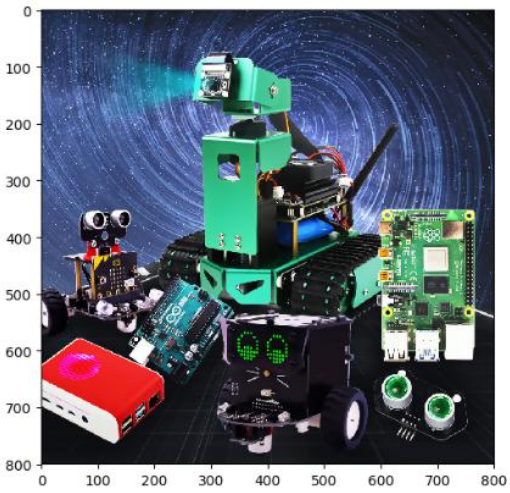
Bilateral filtering is a nonlinear filtering method. This method can only filter out low-frequency information better.
The code is as follows,

```
import cv2

import matplotlib.pyplot as plt



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

#cv2.imshow('src',img)

dst = cv2.bilateralFilter(img,15,35,35)

# 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]



[Skin whitening picture]