Open Source CV image beautification

1、 OpenCV repair image

1.1、Image restoration is a type of algorithm in computer vision

1. Image inpainting is a class of algorithms in computer vision whose goal is to fill areas within an image or video. The area is identified using a binary mask, and filling is usually done according to the area boundary information that needs to be filled. The most common application of image restoration is to restore old scanned photos. It is also used to remove small unwanted objects in images.

1.2、In OpenCV, dst = cv2.inpaint(src, inpaintMask, inpaintRadius, flags) is provided to repair the image

Parameter meaning：

src：source image, which is the image that needs to be repaired

inpaintMask：Binary mask indicating the pixels to be inpainted.

dst：result image

inpaintRadius：Indicates the radius of the repair

flags : Repair algorithm, mainly INPAINT_NS (Navier-Stokes based method) or INPAINT_TELEA (Fast marching based method)

Navier-Stokes based fixes should be slower and tend to produce more ambiguous results than fast marching methods. In practice, we have not found this to be the case. INPAINT_NS produced better results in our tests and was also slightly faster than INPAINT_TELEA.

1.3、Code and actual effect display

（1）、First, we first add damage to the intact picture, which can be understood as modifying the pixel value of a specific part of it

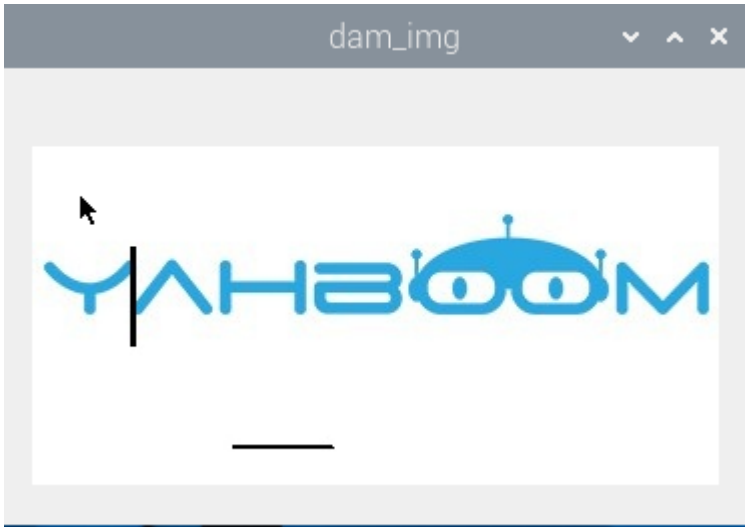
run the program

```
cd ~/yahboomcar_ws/src/yahboomcar_astra/scripts/opencv/
python3 4_1_1.py
```

```
import cv2
import numpy as np
if __name__ == '__main__':
    img = cv2.imread('yahboom.jpg')
    for i in range(50,100):
        img[i,50] =   (0,0,0)
        img[i,50+1] =   (0,0,0)
        img[i,50-1] =   (0,0,0)
    for i in range(100,150):
        img[150,i] =   (0,0,0)
```

```
        img[150,i+1] =   (0,0,0)
        img[150-1,i] =   (0,0,0)
    cv2.imwrite("damaged.jpg",img)
    dam_img = cv2.imread('damaged.jpg')
    while True :
        cv2.imshow("dam_img",dam_img)
        action = cv2.waitKey(10) & 0xFF
        if action == ord('q') or action == 113:
            break
    img.release()
    cv2.destroyAllWindows()
```

After running, a picture will be generated, which is regarded as a damaged picture of the original picture.

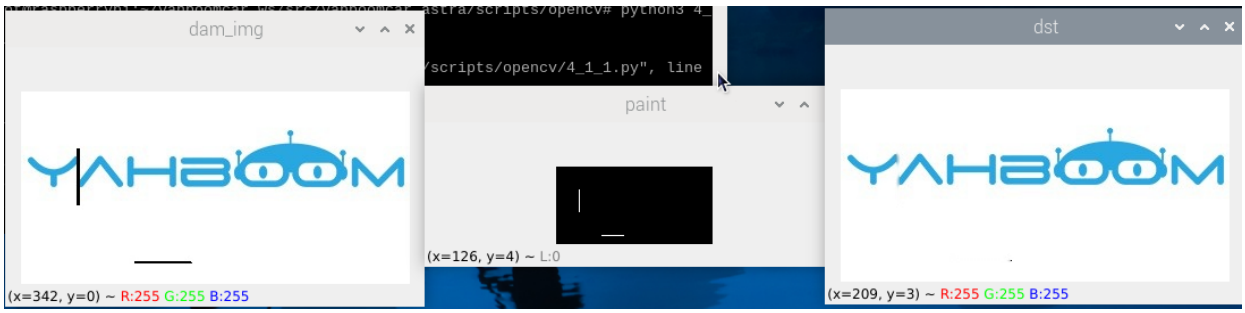


（2）、repair the photo you just created, first read, then create the mask, and finally use the function to repair it

```
# run the program
cd ~/yahboomcar_ws/src/yahboomcar_astra/scripts/opencv/
python3 4_1_2.py
```

```
import cv2
import numpy as np
if __name__ == '__main__':
    dam_img = cv2.imread('damaged.jpg')
    imgInfo = dam_img.shape
    height = imgInfo[0]
    width = imgInfo[1]
    paint =   np.zeros((height,width,1),np.uint8)
    for i in range(50,100):
        paint[i,50] = 255
        paint[i,50+1] = 255
        paint[i,50-1] = 255
    for i in range(100,150):
        paint[150,i] = 255
```

```
        paint[150+1,i] = 255
        paint[150-1,i] = 255
    dst_img =  cv2.inpaint(dam_img,paint,3,cv2.INPAINT_TELEA)
    while True :
        cv2.imshow("dam_img",dam_img)
        cv2.imshow("paint",paint)
        cv2.imshow("dst",dst_img)
        action = cv2.waitKey(10) & 0xFF
        if action == ord('q') or action == 113:
            break
    img.release()
    cv2.destroyAllWindows()
```



如图所示，左边为修复前，中间的是掩码图片，右边是修复后的原图。

2、 OpenCV image brightness enhancement

2.1、Implementation process：

1. Implementation process: Amplify the three-channel value of each pixel synchronously, while keeping the channel value between 0-255. In fact, it is to traverse each pixel, add and subtract values to them, and then judge the three channels. Whether rgb is in the range of 0-255, if it is greater or less than 255 or 0.

2.2、Code and actual effect display

```
cd ~/yahboomcar_ws/src/yahboomcar_astra/scripts/opencv/
python3 4_2.py
```

```
import cv2
import numpy as np
if __name__ == '__main__':
    img = cv2.imread('yahboom.jpg')
    imgInfo = img.shape
    height = imgInfo[0]
    width = imgInfo[1]
    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) + 100
            gg = int(g) + 100
```

```
            rr = int(r) + 100
            if bb > 255:
                bb = 255
            if gg > 255:
                gg = 255
            if rr > 255:
                rr = 255
            dst[i,j] = (bb,gg,rr)
    while True :
        cv2.imshow("dst",dst)
        action = cv2.waitKey(10) & 0xFF
        if action == ord('q') or action == 113:
            break
    img.release()
    cv2.destroyAllWindows()
```

图片是增加亮度后的照片

3、OpenCV image microdermabrasion and whitening

3.1、OpenCV realizes the function of microdermabrasion and whitening of pictures

1. OpenCV realizes the function of microdermabrasion and whitening of pictures. The principle of implementation is basically the same as the principle of "1.20 OpenCV picture brightness enhancement", but here we do not need to process the r value, just follow this formula, p = p(x)*1.4+ y, where p(x) represents the b channel or g channel, and y represents the value that needs to be increased or decreased. Similarly, after adding the value, we need to judge the value.

3.2、Code and actual effect display

```
cd ~/yahboomcar_ws/src/yahboomcar_astra/scripts/opencv/
python3 4_3.py
```

```
import cv2
import numpy as np
```

```python
if __name__ == '__main__':
    img = cv2.imread('yahboom.jpg')
    imgInfo = img.shape
    height = imgInfo[0]
    width = imgInfo[1]
    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*1.4) + 5
            gg = int(g*1.4) + 5
            if bb > 255:
                bb = 255
            if gg > 255:
                gg = 255
            dst[i,j] = (bb,gg,r)
    while True :
        cv2.imshow("origin",img)
        cv2.imshow("dst",dst)
        action = cv2.waitKey(10) & 0xFF
        if action == ord('q') or action == 113:
            break
    img.release()
    cv2.destroyAllWindows()
```