Image binarization

1. Implementation principle

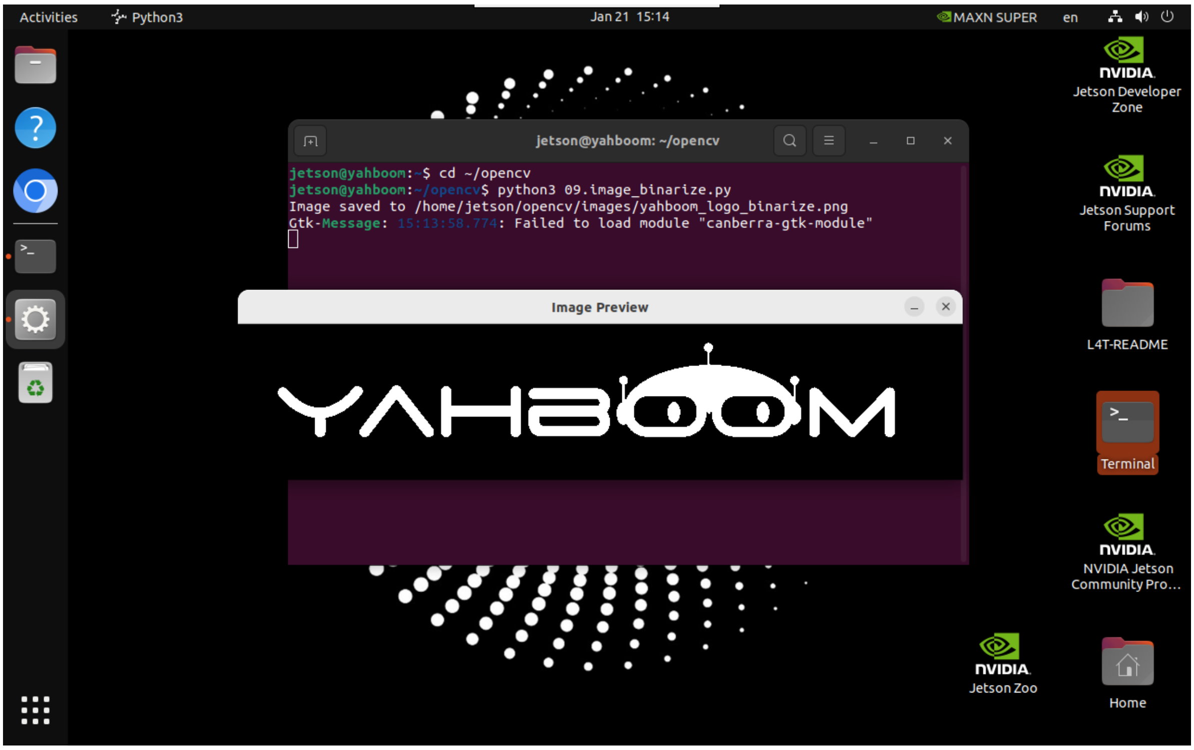
Use `cv2.threshold()` function to binarize the image.

2. Implementation effect

```
cd ~/opencv
```

```
python3 09.image_binarize.py
```

Note: Select the image and press `q` to exit the program!



```
jetson@yahboom:~$ cd ~/opencv
jetson@yahboom:~/opencv$ python3 09.image_binarize.py
Image saved to /home/jetson/opencv/images/yahboom_logo_binarize.png
Gtk-Message: 15:13:58.774: Failed to load module "canberra-gtk-module"
```

3. Implementation code

```
import cv2

def binarize_image(input_path, output_path, threshold):
    image = cv2.imread(input_path, cv2.IMREAD_GRAYSCALE)
    if image is None:
        print("Error: Unable to open image file.")
        return
    _, binary_image = cv2.threshold(image, threshold, 255, cv2.THRESH_BINARY)
    if cv2.imwrite(output_path, binary_image):
        print(f"Image saved to {output_path}")
        cv2.imshow('Image Preview', cv2.imread(output_path))
        cv2.waitKey(0)
        cv2.destroyAllWindows()
    else:
        print("Error: Unable to save image file.")

binarize_image('/home/jetson/opencv/images/yahboom_logo.png', \
               '/home/jetson/opencv/images/yahboom_logo_binarize.png', \
               127)
```