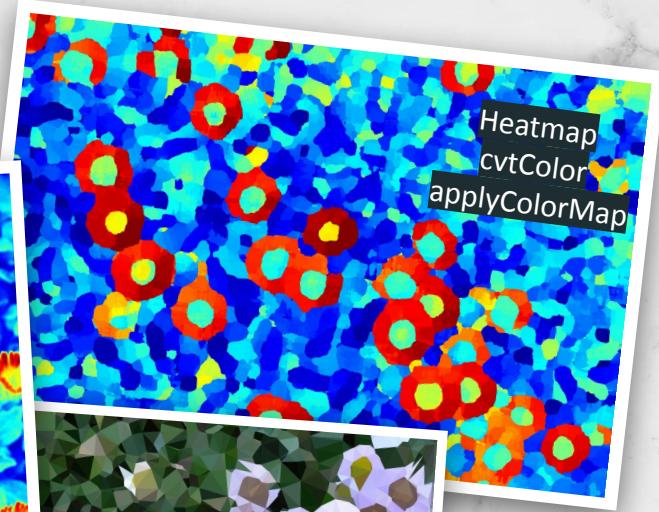
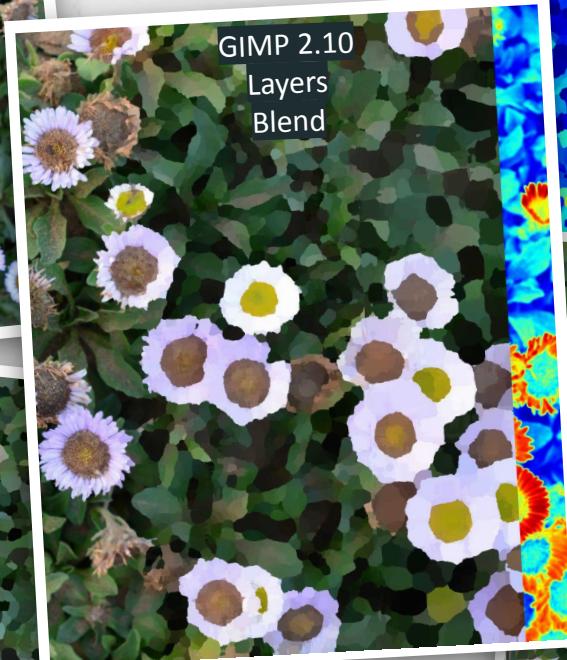


---

Image  
oilPainting  
Heatmap  
Pixellate  
Triangulate



Python Image Processing + GIMP



R \_\_\_\_\_ G \_\_\_\_\_ B \_\_\_\_\_  
White \_\_\_\_\_ Black \_\_\_\_\_



Created by Python + GIMP 2.10

# DSC\_2202.jpg

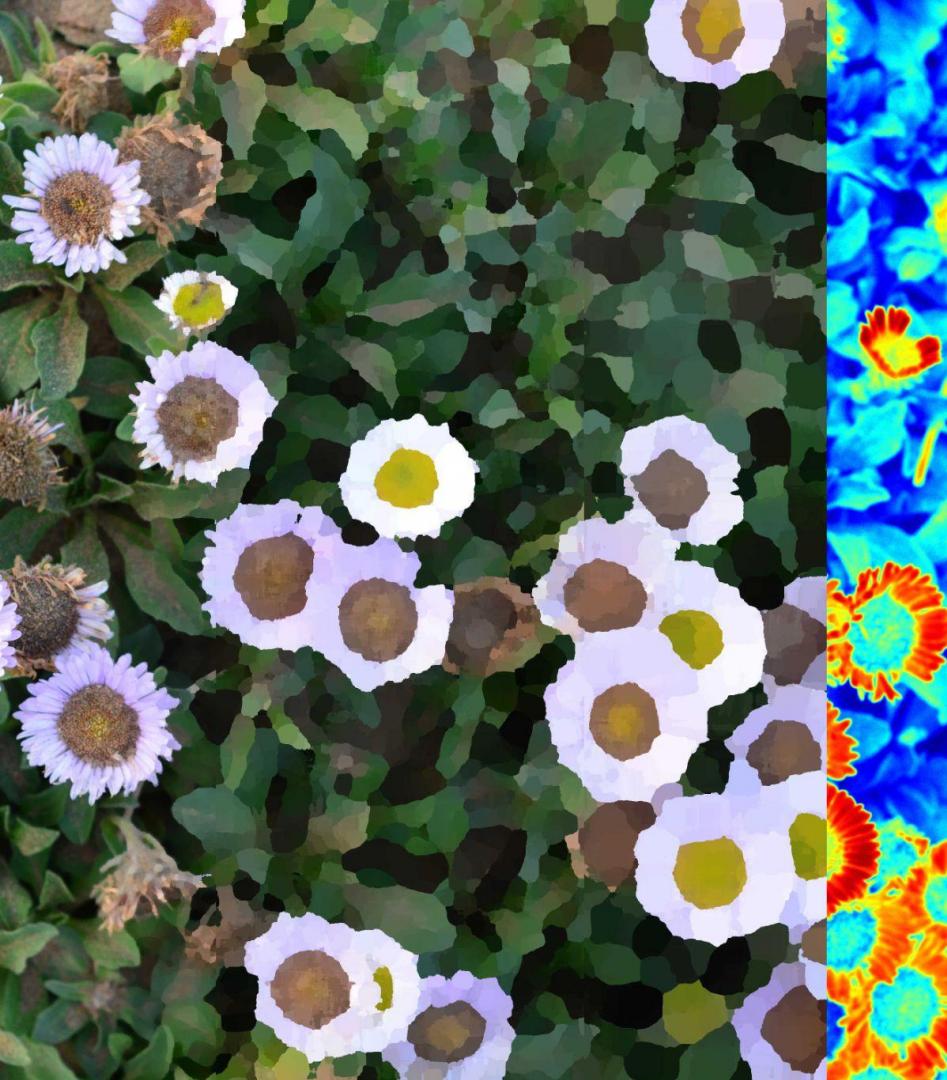
```
python imageFilterOptions.py --image PXL_20220414_015637240_2.jpg
```

oilPainting:

```
image = cv2.imread(args["image"])
oilpaint25 = cv2.xphoto.oilPainting(image, 25, 1)
cv2.imwrite(str(filename + '-oilpainting_02_25.jpg'), oilpaint25)
```

Heatmap:

```
image = cv2.imread(args["image"])
img_RGB_Gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY) ### RGB to Gray
filename_RGB_Gray = str(filename + '-RGB-GRAY.jpg')
cv2.imwrite(filename_RGB_Gray, img_RGB_Gray)
### RGB to Gray Scale to Heatmap
img_RGB_Heatmap = cv2.applyColorMap(img_RGB_Gray, cv2.COLORMAP_JET)
filename_RGB_Heatmap = str(filename + '-RGB-Heatmap.jpg')
cv2.imwrite(filename_RGB_Heatmap, img_RGB_Heatmap)
```





A screenshot of a desktop environment showing a Jupyter Notebook, a file explorer, and a file viewer.

The Jupyter Notebook displays Python code for applying an "Oil Painting Effect" to images:

```
!git clone https://github.com/Logozx/OilPaintImageFilterEffect
cd OilPaintImageFilterEffect
python oilpainting.py <image> <output>
# or
python oilpainting.py <image> <output> <size>
```

The file explorer shows a directory structure with numerous files named like "IMG\_20210813\_102738.jpg" and "IMG\_20210813\_102738-oilpainting\_01\_50.jpg".

The file viewer shows the original image "DSC\_6034.JPG" and several versions of it with the "Oil Painting Effect" applied, labeled "Oil Painting Effect 01", "Oil Painting Effect 02", and "Oil Painting Effect 03".