

Cartoonify

Artistic Image Filters & Effects

Dino Anastopoulos (1900661)
30 June 2021
APPM4058A: Digital Image Processing

Main Effects

Cartoonify

- Step 1:** Read in colour image.
- Step 2:** Convert to grayscale.
- Step 3:** Blur this gray image.
- Step 4:** Extract the edges.
- Step 5:** Perform colour quantization using K-means clustering with 10 colours.
- Step 6:** Blur this reduced colour image.
- Step 7:** Combine the extracted edges and the blurred reduced-colour image.





Pencil Sketch

- Step 1:** Read in colour image.
- Step 2:** Convert to grayscale.
- Step 3:** Invert the grayscale image.
- Step 4:** Blur the inverted grayscale image.
- Step 5:** Perform dodging.
- Step 6:** Perform burning. (Optional)
- Step 7:** Final pencil sketch is done.



Original



Gray



Inverted



Blurred



Dodge



Result

— — —



Pixelate

Step 1: Read in colour image.

Step 2: Calculate width and height of image.

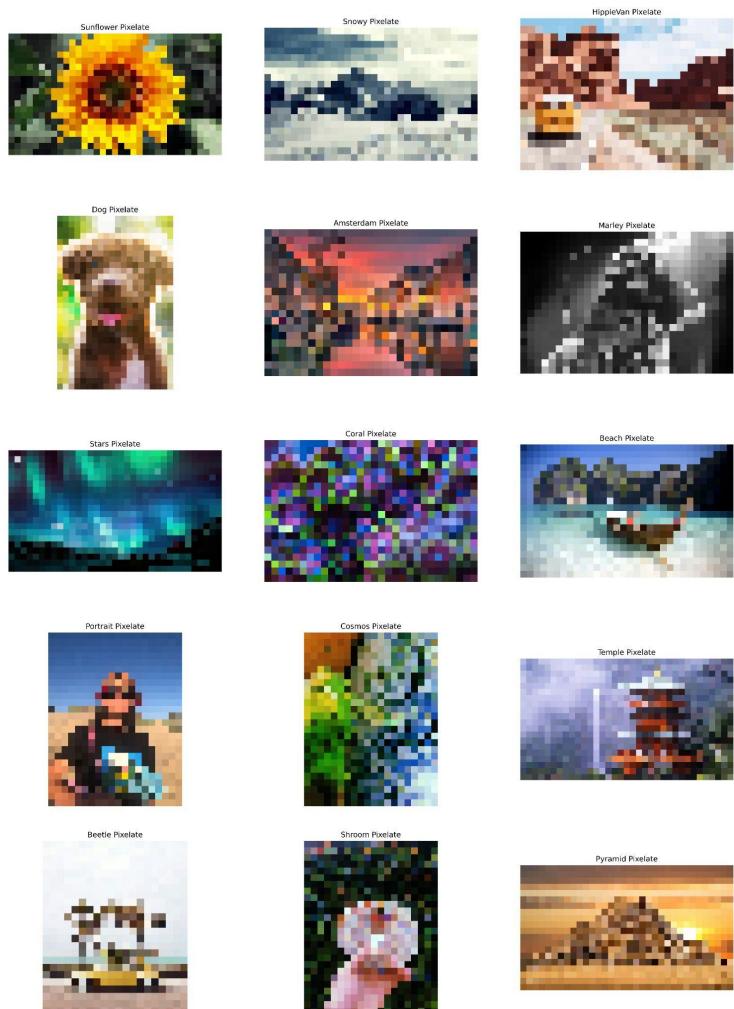
Step 3: Calculate aspect ratio (width/height).

Step 4: Set the new length of the shorter side equal to 20.

Step 5: Set the new length of the longer side to $20 * \text{AspectRatio}$ if the image is landscape or $20 * 1 / \text{AspectRatio}$ if the image is portrait.

Step 6: Resize the image to the new size using Linear Interpolation.





Emboss/Engrave

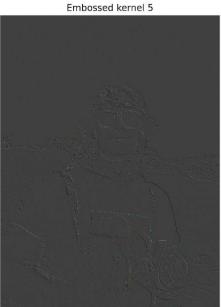
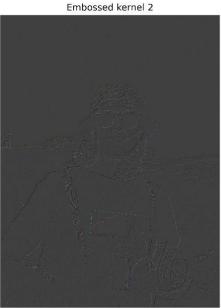
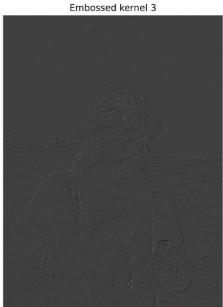
1) Randomly select a kernel:

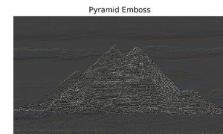
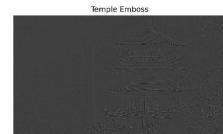
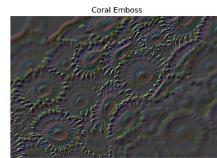
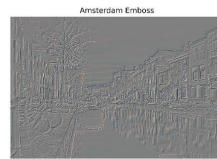
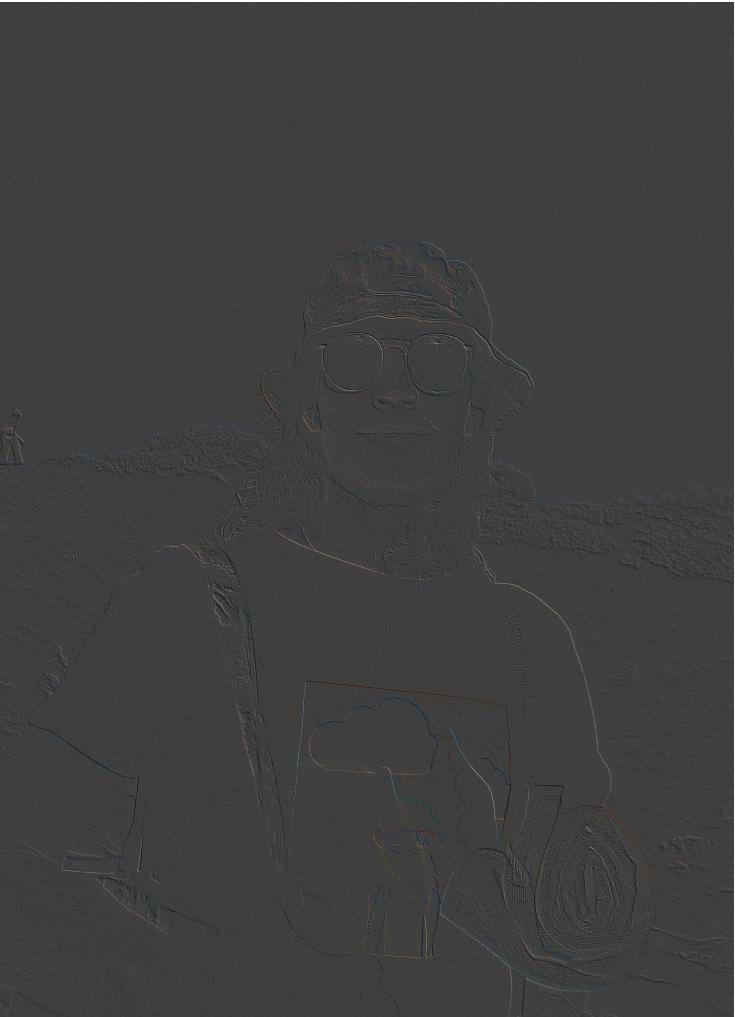
- $\begin{bmatrix} -2, -1, 0 \\ -1, 1, 1 \\ -1, 1, 2 \end{bmatrix}$
- $\begin{bmatrix} -1, 0 \\ 0, 1 \end{bmatrix}$
- $\begin{bmatrix} 1, 0 \\ 0, -1 \end{bmatrix}$
- $\begin{bmatrix} 0, 1 \\ -1, 0 \end{bmatrix}$
- $\begin{bmatrix} 0, -1 \\ 1, 0 \end{bmatrix}$
- $\begin{bmatrix} 1, 0, 0 \\ 0, 0, 0 \\ 0, 0, -1 \end{bmatrix}$
- $\begin{bmatrix} -1, 0, 0 \\ 0, 0, 0 \\ 0, 0, 1 \end{bmatrix}$

2) Randomly select offset:

- 64 (Darker Gray)
- 128 (Lighter Gray)

3) Convolve kernel with image, and offset all gray level pixels

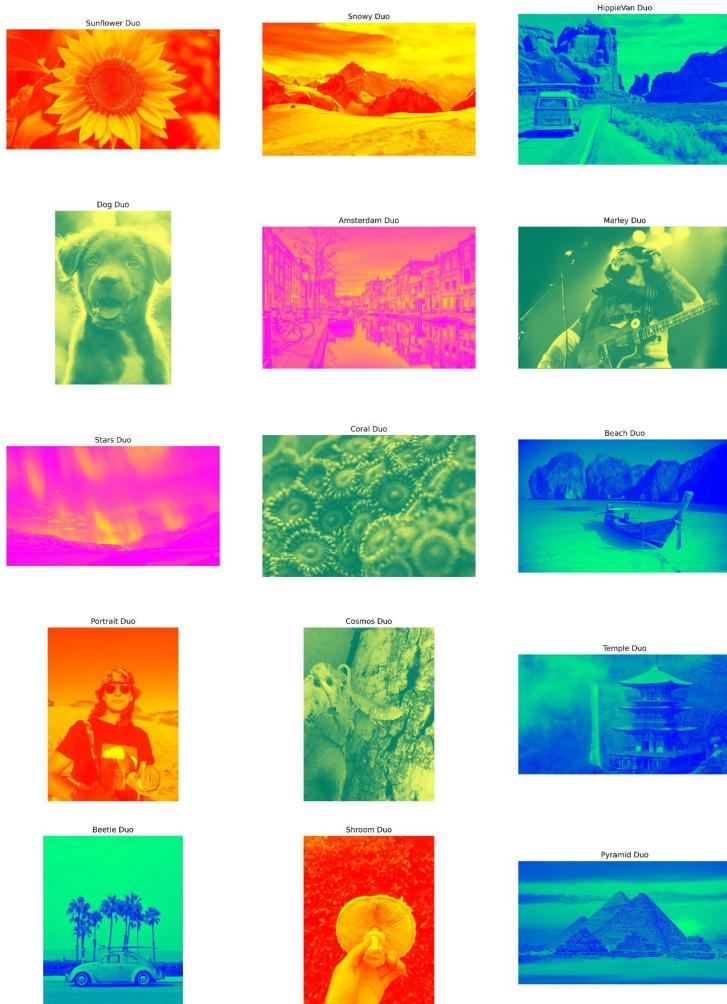


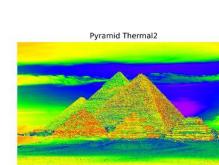
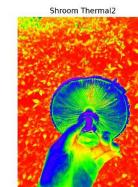
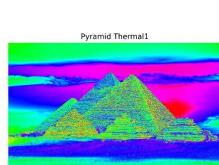
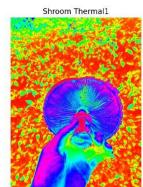
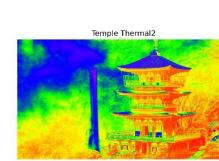
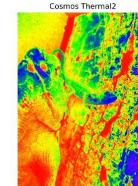
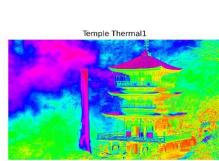
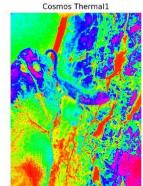
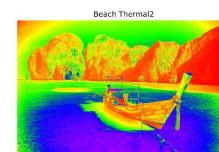
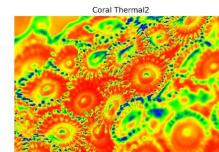
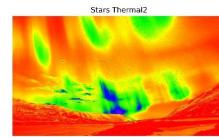
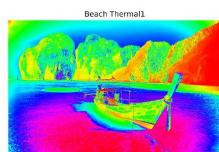
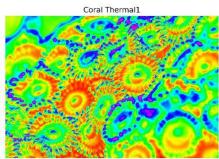
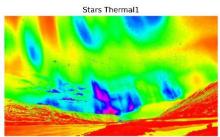
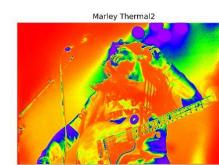
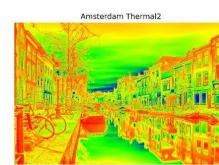
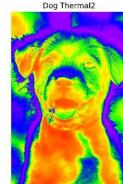
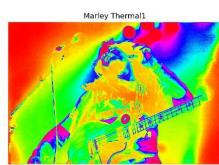
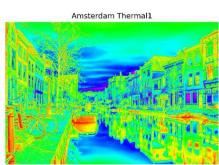
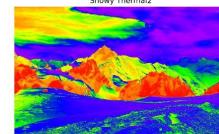
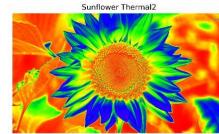
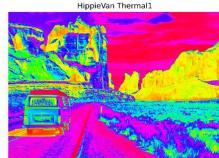
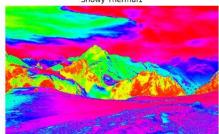
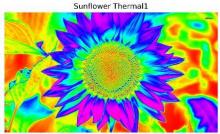


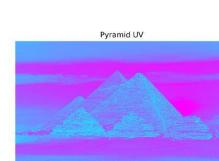
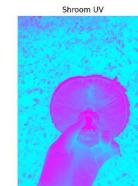
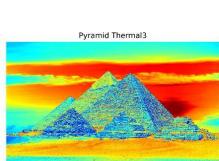
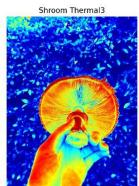
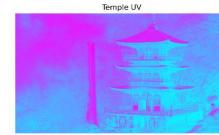
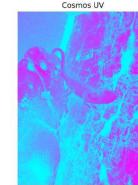
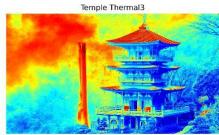
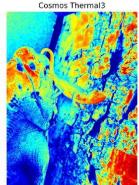
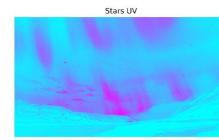
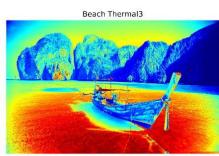
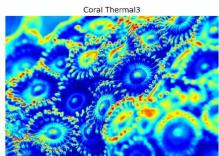
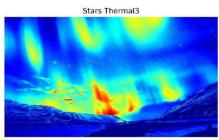
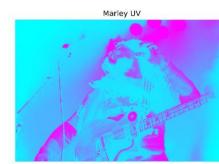
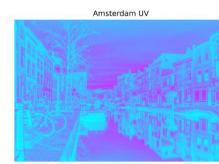
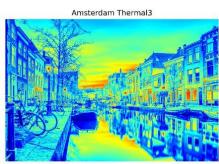
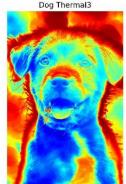
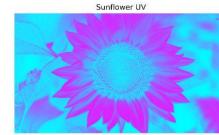
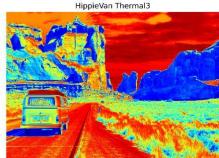
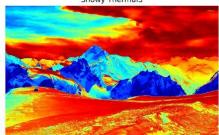
Extra Effects

1) ColourMaps

- Duo Chromatic
- Thermal 1
- Thermal 2
- Thermal 3
- UV

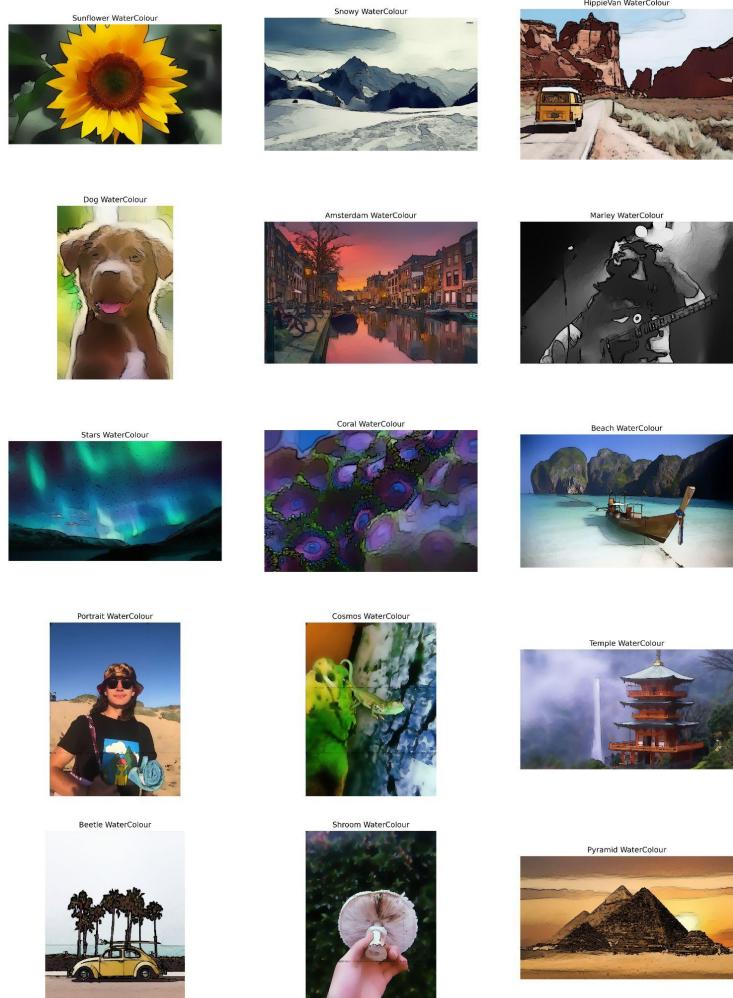






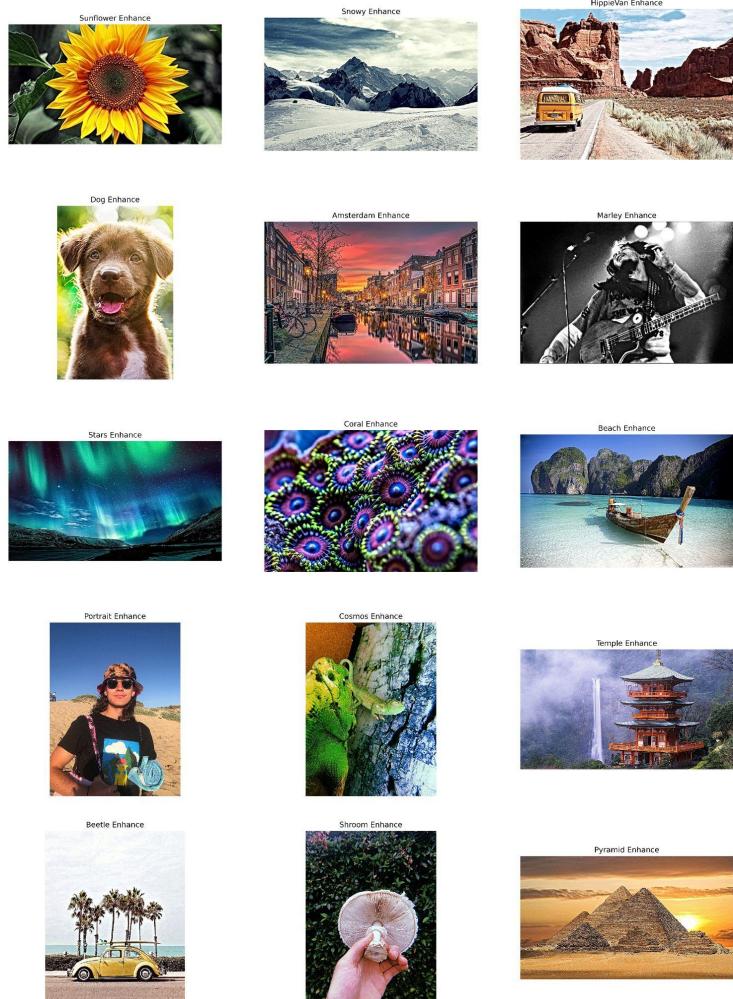
2) WaterColour

- cv2.stylization()



3) Enhance

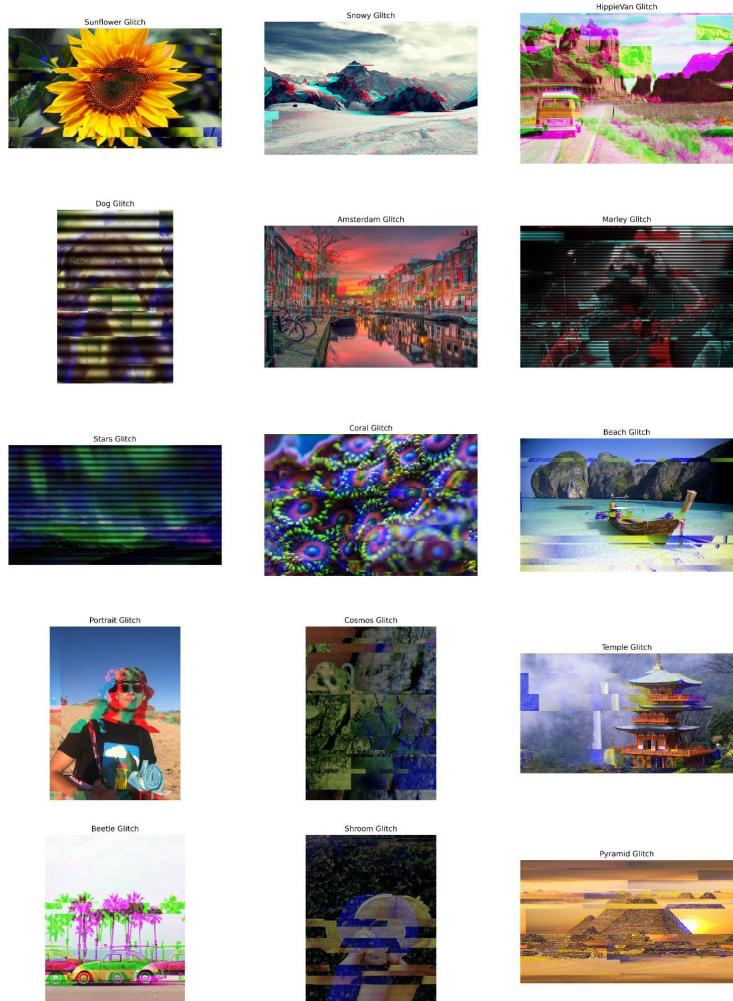
- `cv2.detailEnhance()`



4) Glitch

- `glitch_this` library

<https://github.com/TotallyNotChase/glitch-this>



THE END