COS 426: PRECEPT 2

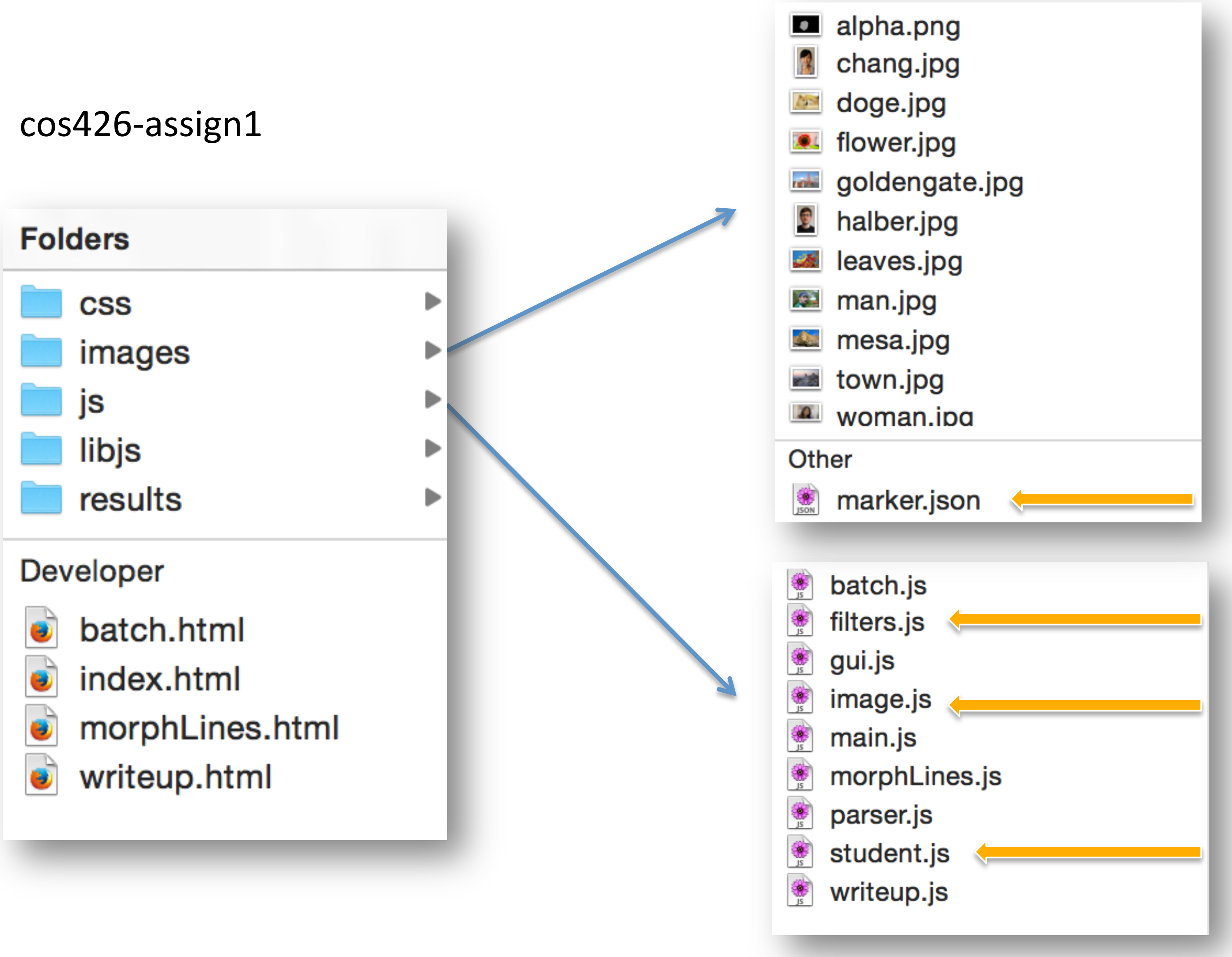
Huiwen chang

Assignment 1: Image Processing

- Structure of the assignment
- Implementation of filters operations
  - Luminance
  - Color
  - Filter
  - Dither
  - Resampling

Structure

cos426-assign1



Folders

- css
- images
- js
- libjs
- results

Developer

- batch.html
- index.html
- morphLines.html
- writeup.html

- alpha.png
- chang.jpg
- doge.jpg
- flower.jpg
- goldengate.jpg
- halber.jpg
- leaves.jpg
- man.jpg
- mesa.jpg
- town.jpg
- woman.jpg

Other

- marker.json

- batch.js
- filters.js
- gui.js
- image.js
- main.js
- morphLines.js
- parser.js
- student.js
- writeup.js

Structure

- Interactive Mode
  - Photolist (edit it in Gui.js)
  - morphLines
- Batch Mode
  - Gui to Batch
  - Brightness Animation
  - newTab
    - Multiple parameters
    - Multiple images
    - Gif
    - Art

Morph Lines

- Read two images and create your own morph lines correspondence.
- You could modify your morphlines by including &marker = yourmakerfile to load it in.
- Read JSON in your code

```
linek = lines.initial[k]
```

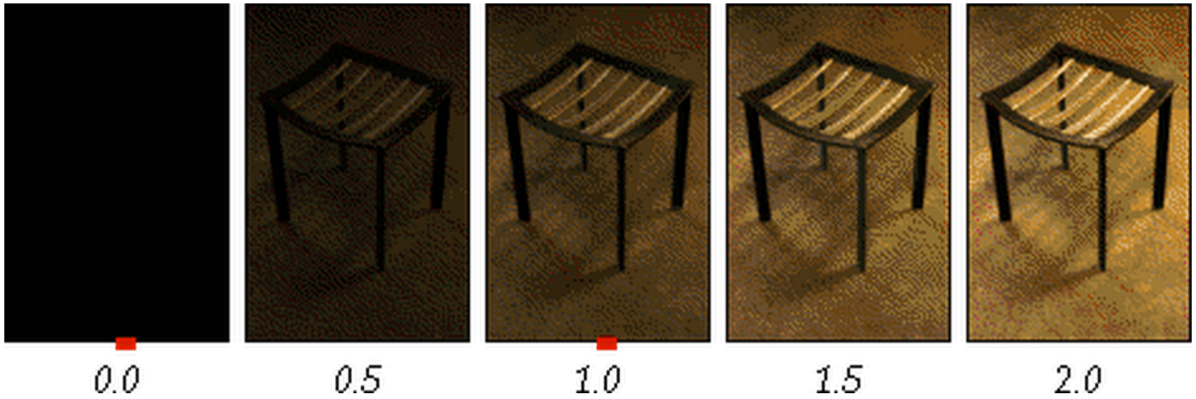
  – linek.x0, linek.y0, linek.x1, linek.y1,

Implementation

- Graphica Obscura

  out = (1 - alpha)*in0 + alpha*in1

  - brightness:

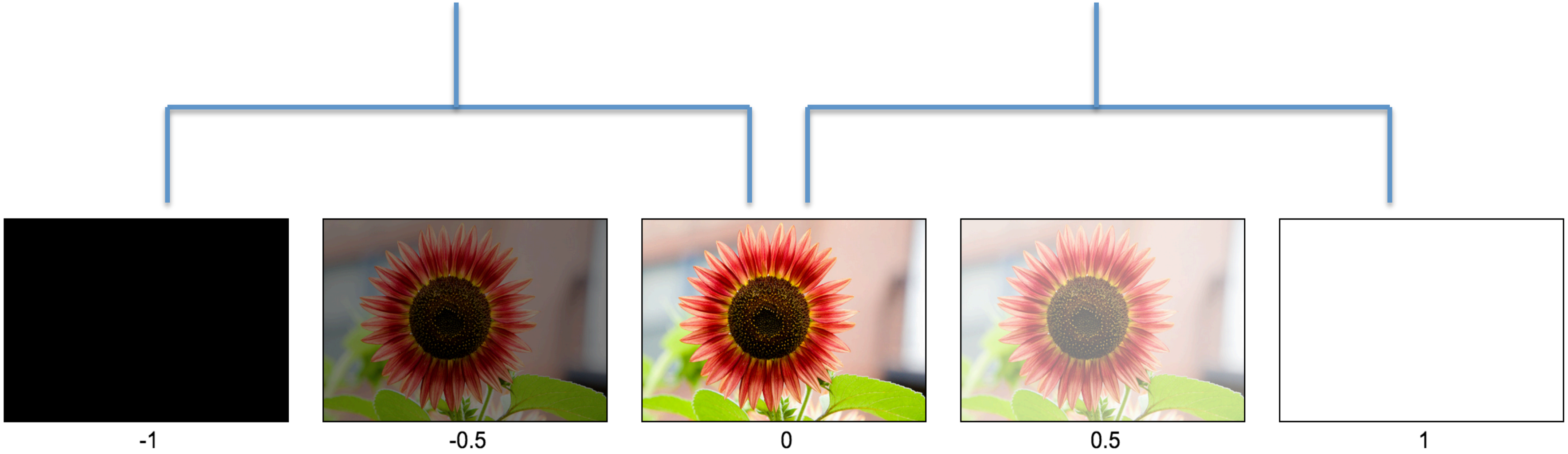


- problem: it does not make great use of the full range of the slider

Brightness

Ratio < 0:
interpolate with black

Ratio > 0:
interpolate with white

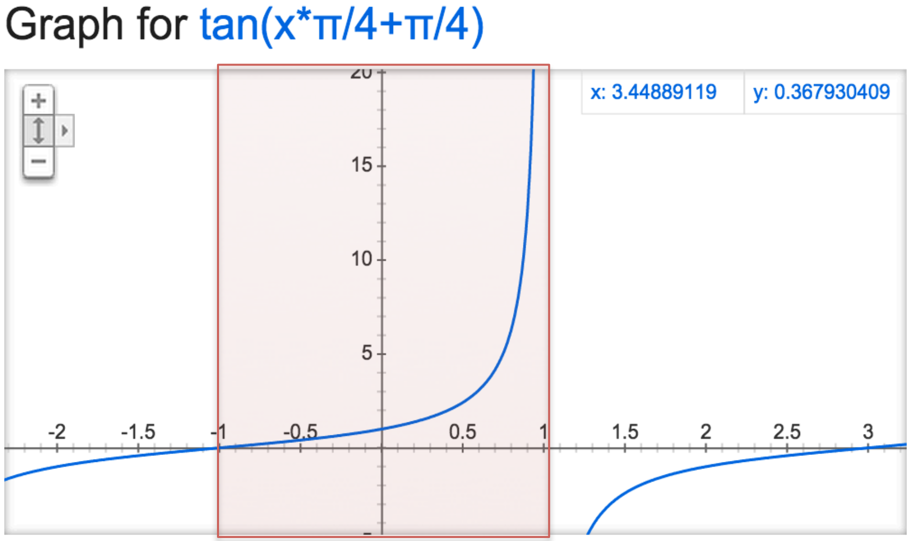


See wiki_GIMP_contrast_brightness

Contrast

- Map [-1, 1) to [0,∞) by

```
Ratio = tan((Ratio+1)*PI/4)
```



Graph for tan(x*π/4+π/4)

interpolate with gray

-1

-0.5

0

0.5

1

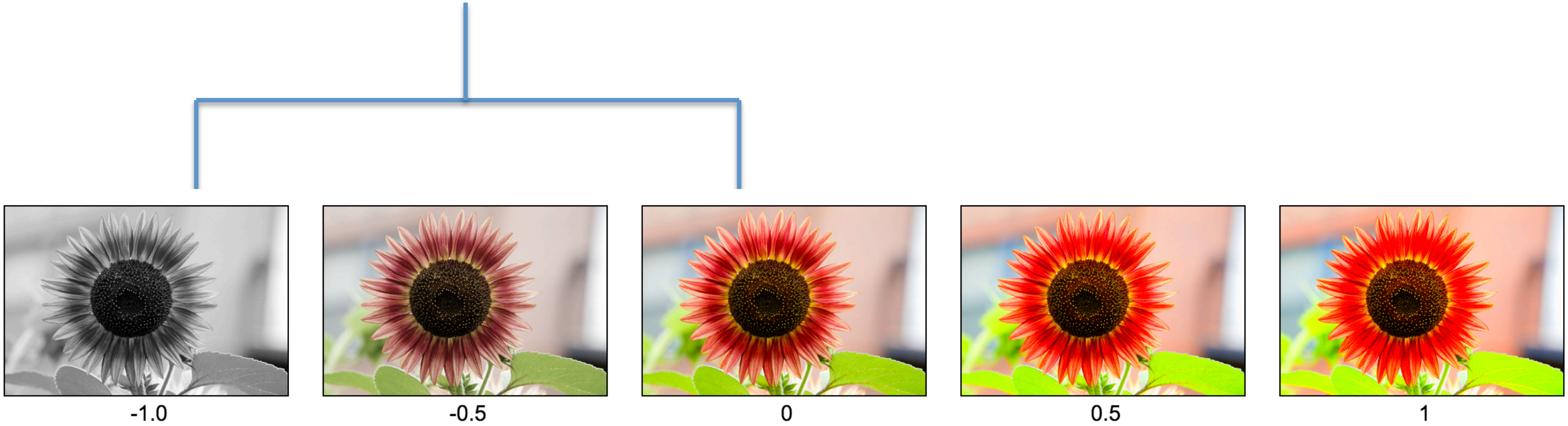
See [wiki_GIMP_contrast_brightness]

Saturation

- Map [-1, 1] to [0, 2] by

```
Ratio = Ratio + 1;
```

interpolate with grayscale image



See wiki GIMP contrast brightness

Gray

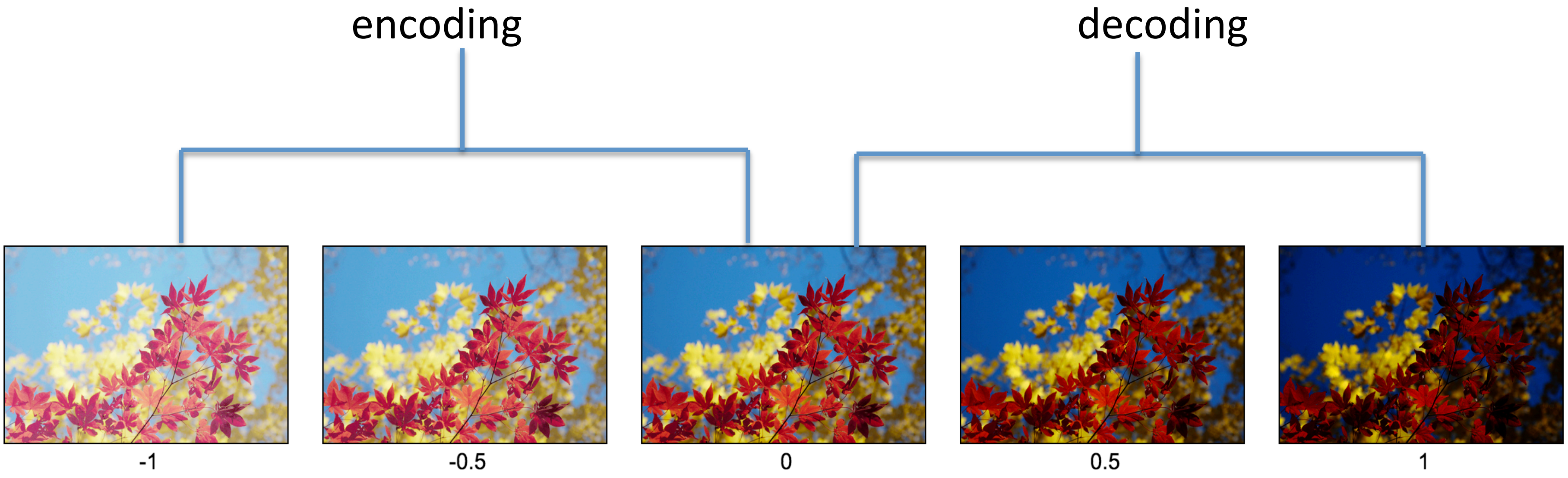
- Either way is ok:
  - Luminance (standard for certain color spaces):
    `0.2126*R + 0.7152*G + 0.0722*B`
  - Luminance (perceived option 1):
    `0.299*R + 0.587*G + 0.114*B`

From piazza@11

Gamma Correction

```
V_out = Math.pow(V_in, \gamma)
```

- `V_in` is the rgb values in [0,1], the result pixel is `V_out` x 255



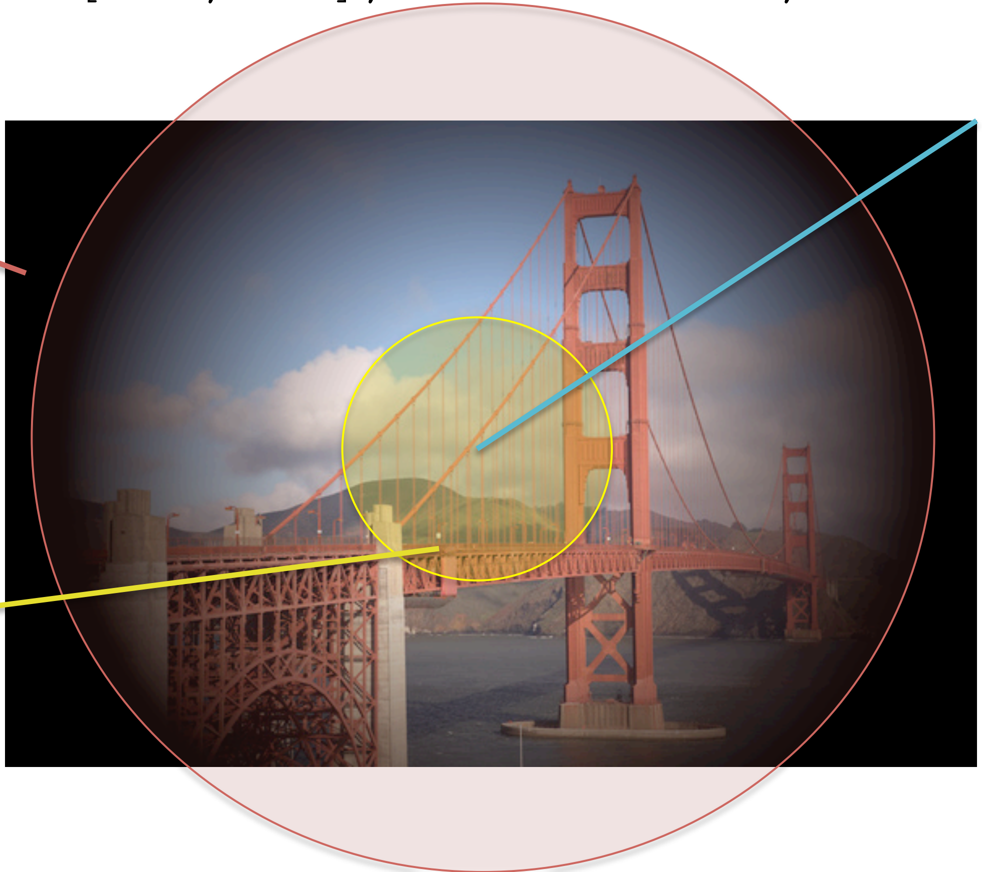
encoding

decoding

Vignette

```
innerR = 0.5 - 0.5 * value[0];
outerR = 0.5 + 0.5 * value[1];
```

Example: `value=[0.5,0.5], innerR=0.25, outerR=0.75`



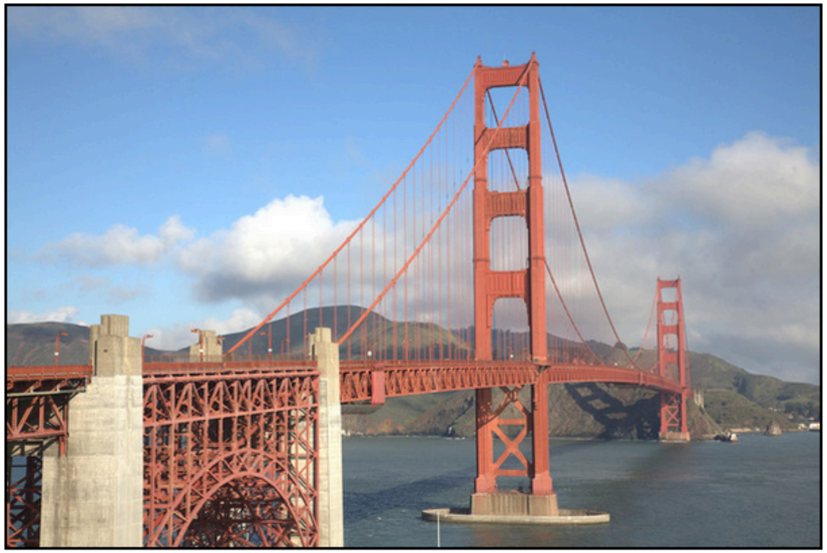
Pixel outside outerR is black

Pixel inside innerR is clear

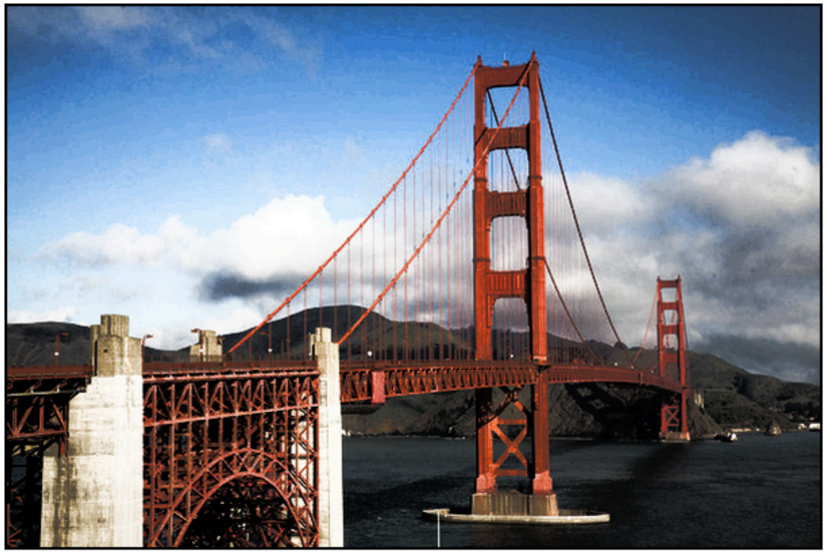
White balance

- First, map RGB to [0,1]
- RGB -> LMS
- divided by L_w M_w S_w
- LMS -> RGB
- Map back to 0-255

Histgram equalization



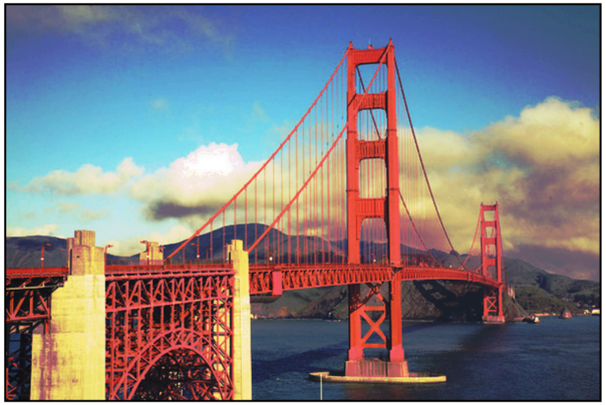
Before



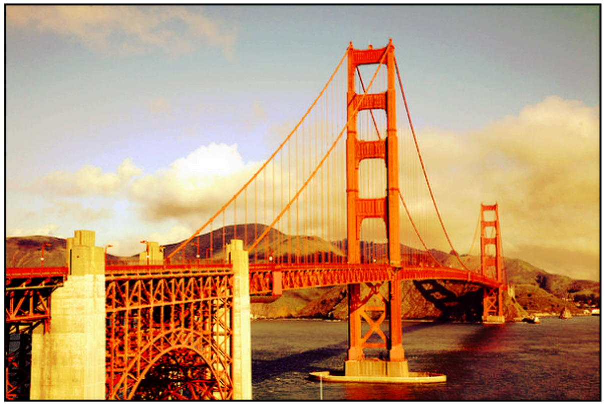
After

Histogram Matching

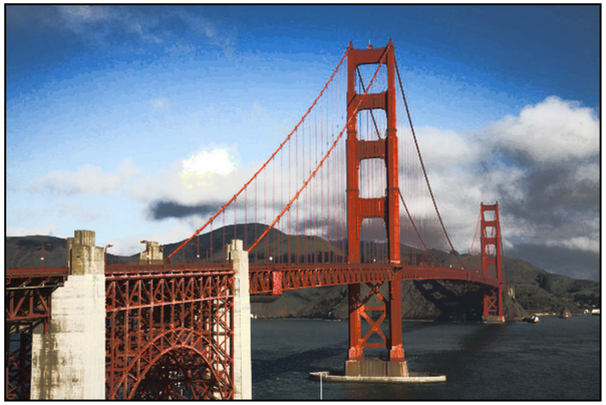
- Tips: Choose a reasonable reference image



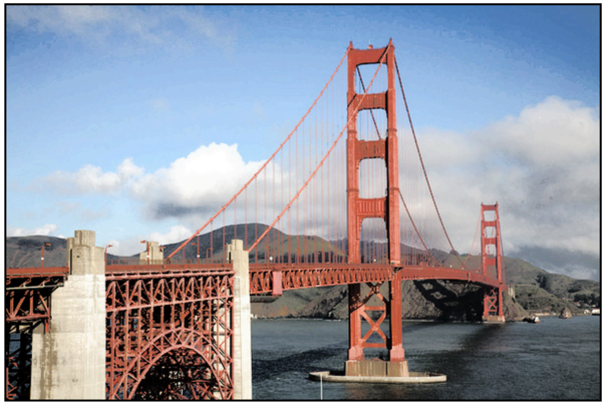
reference image: town



reference image: flower



reference image: town



reference image: flower

Gaussian Filter

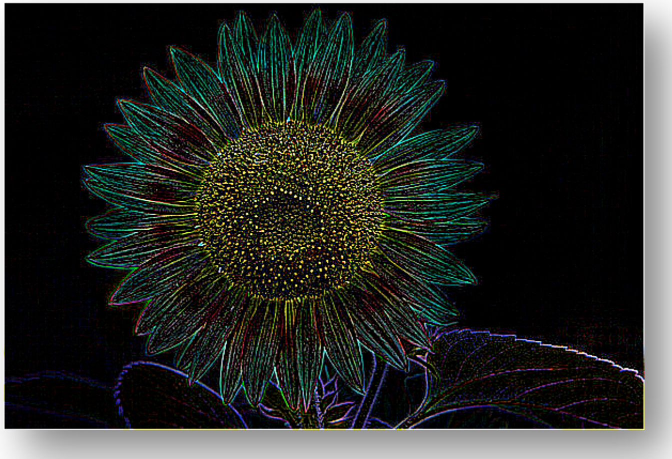
- Tips:
  - Weight should be normalized.
  - Border pixels
  - Create new image

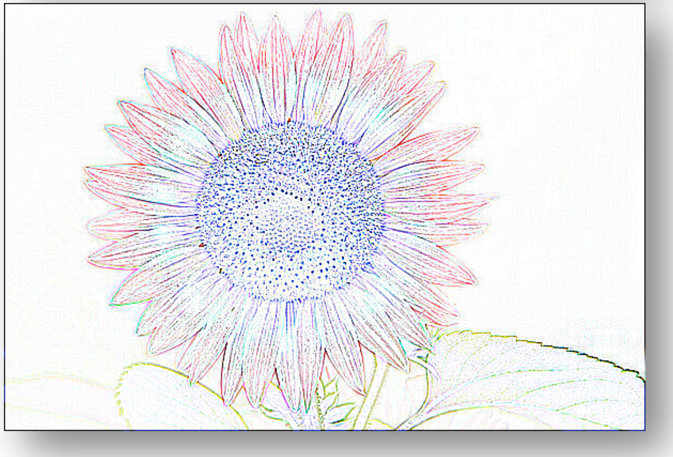
Edge

- Tips:
  - Weight should not be normalized.
  - Border pixels
  - Create new image

-1	-1	-1
-1	8	-1
-1	-1	-1

	3	-1
	-1	-1



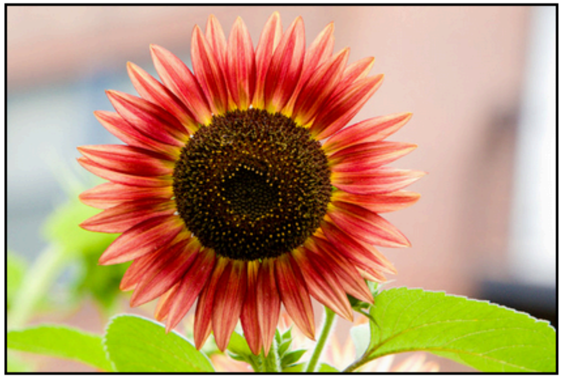


Sharpen

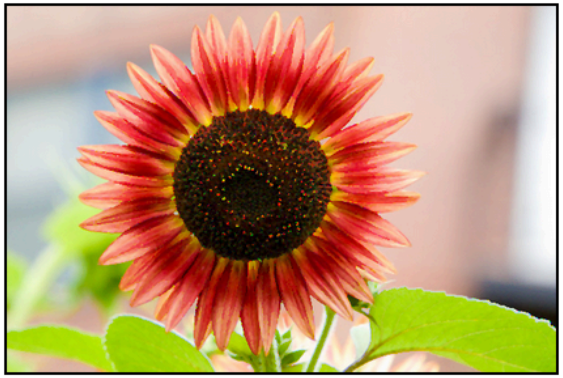
- Tips:
  - Weight could be normalized.
  - Border pixels
  - Create new image

Median

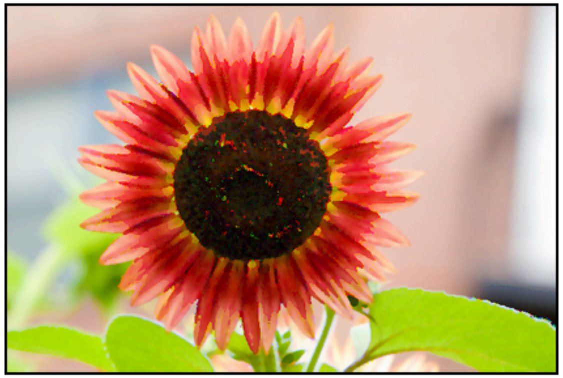
- RGB vs Luminance



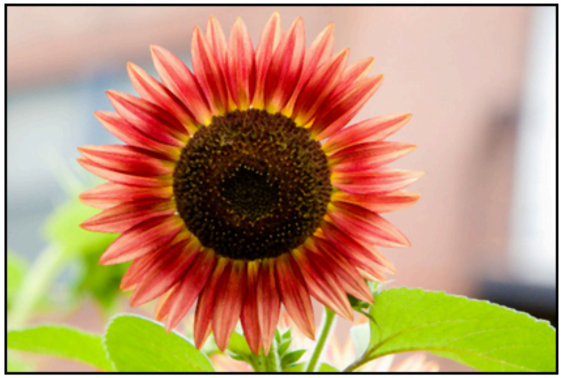
1



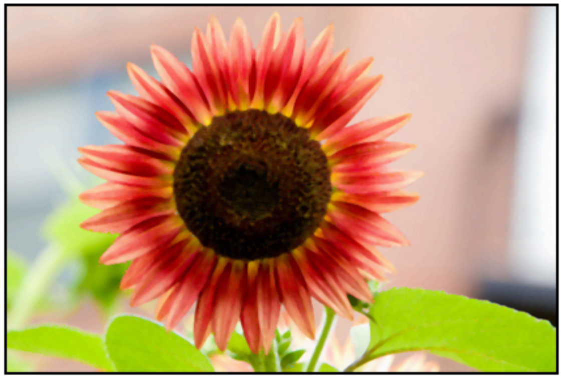
2



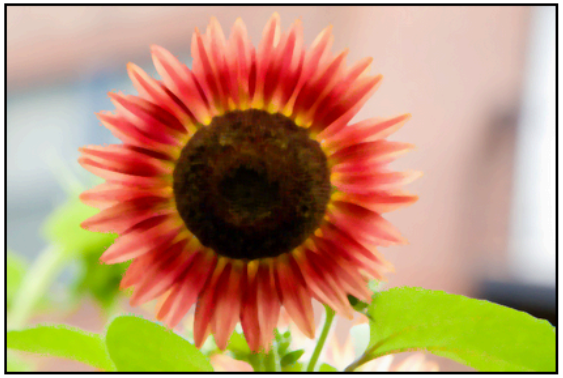
3



1



2



3

Bilateral

- Color sigma
  - calculate the distance in rgb [0,1]
- Weighted should be normalized
- Make two sigmas more equalized

Dither

- First check your results in grayscale.
- Random – make sure you have positive and negative random noise.
- Floyd – border pixels

Sampling

- Create a new image
- Rotation:
  - Set the alpha of outside pixel as 0
- Swirl:
  - For the outside pixels, find its nearest pixel inside the photo.

More tips

- Don’t worry about minor difference with results in example page.
  - contrast, quantize, random…
  - Just make sure your results are reasonable.
- Which rgb range this operation should process in. [0,1] or 0-255?
- Need to create new images?
- No ~~256~~

Q&A