

Assignment#1 Q2 ReadMe

Name: Wen Sheng
E-Mail: wen.sheng@yale.edu

Operating System: Mac OS 10.10.5

Browser: Firefox 40.0.3 / Safari 8.0.8

(For chrome 45.0, this demo needs a local server to get rid of a Security Error)

Q2 478/578 Bilateral

I. Visual Effect (n, m)

This filter makes each pixel affected by surrounding $n * n - 1$ pixels (All $n * n$ pixels will potentially affect the center point including the center pixel) to get a more continuous/obscure and not unique/intense in each pixel. The value of m is between 0 and 255.

An surrounding Pixel (i, j) will be applied to center pixel (x, y) only when $|Intensity(i, j) - Intensity(x, y)| \leq m$. When m increases, more pixels in the $n * n$ filter will be countered and the image becomes increasingly obscure and continuous. More importantly, using this method could make the image with less noise spots, while in the same time distorting the edges in the picture.

II. Input

- Filter Size n: 2
- Filter Intensity M: 20
- Image source: tigereye.jpg

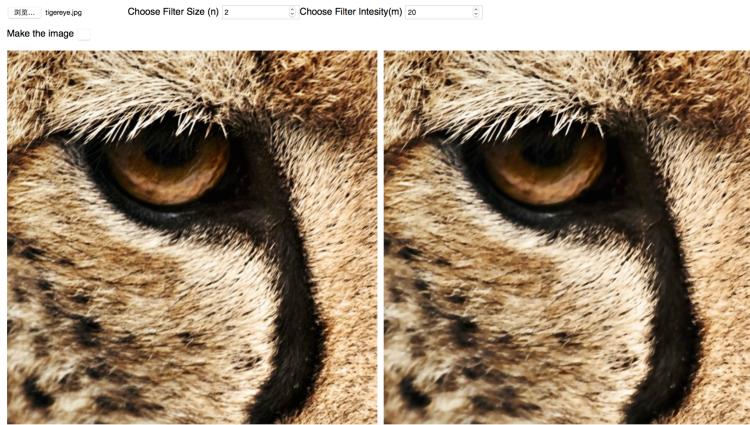
III. Special Issues

Fixed fake path error in example code when using Chrome. I modified the path expression of the image by using the relative path:

```
img.src = "./" + document.getElementById("imagefilename").value.replace(/^.*(\\"|\\\:)/, "");
```

IV. Screenshot

Q2 478/578 Filtering, ID: ws362



Q2 578 Other

I. Visual Effect (n, m)

Find the center points that are different/conspicuous with surrounding pixels. In other words, this will find potential edges, outline, and noise spots.

II. Input

- Filter Size n: 2

- Filter Intensity M: 20
- Image source:sunset.jpg

III. Special Issues

To make those different/conspicuous points more visible, I magnified its (r,g,b) value by multiply 20 then get the remainder from 255.

```
diffData.data[4 * index] = Math.abs(imoutData.data[4 * index] - imageData.data[4 * index]) * 20 % 255;  
diffData.data[4 * index + 1] = Math.abs(imoutData.data[4 * index + 1] - imageData.data[4 * index + 1]) * 20 % 255;  
diffData.data[4 * index + 2] = Math.abs(imoutData.data[4 * index + 2] - imageData.data[4 * index + 2]) * 20 % 255;
```

IV. Screenshot

