1 Bilateral Filtering

1.1 Approach

$$g(i,j) = \frac{\sum_{(k,l)\in S(i,j)} f(k,l)w(i,j,k,l)}{\sum_{(k,l)\in S(i,j)} w(i,j,k,l)}$$

g(i, j) : the output point.
S(i, j) : the size range of (2N+1)(2N+1) centered on (i,j).
f(k, l): input points(multiple).
w(i, j, k, l) The value calculated by two gaussian functions (not the weight).

We transform the above formula, assuming that w(I,j,k,l) in the formula is m, then:

$$= \frac{f1 * m1 + f2 * m2 \cdots + fn * mn}{m1 + m2 \cdots + mn}$$

Set m1+m2+m3 ⋯ +mn = M, then:

$$= f1\frac{m1}{M} + f2\frac{m2}{M} + \cdots fn\frac{mn}{M} = g(i,j)$$

For w(i, j, k, l): ws is spatial adjacent gaussian function, and wr is pixel value similarity gaussian function.

$$w = ws * wr$$

$$ws = e^{(-\frac{(i-k)^2+(j-l)^2}{2\sigma_s^2})}$$

$$wr = e^{(-\frac{\|f(i,j)-f(k,l)\|^2}{2\sigma_r^2})}$$

Where, when calculating the difference of color pictures, the sum of RGB three-channel values at (I, j) should be subtracted from the sum of RGB three-channel values at (k, l). This is the color space calculation, not as a single channel, but in the final convolution of the matrix, it's a single channel multiplied by the weights instead of the sum of the three channels.

Implements bilateral filter——function B = bfltColor (A,w,sigma_d,sigma_r)
A is the matrix normalized to [0,1] for the given image
W is the edge length /2 of the bilateral filter (core)
The domain sigma is SIGMA(1), range sigma is SIGMA(2).

1.2 Results

Parameters:
In the test program, the size of three original color images is 200*200pixels;

Bilateral filter half-width w=5;
Bilateral filter standard deviations, sigma = [3 0.1];
The output images are as follows,

original image

original image

original image

bilteredimage

bilteredimage

bilteredimage

2 Halloween Horror Show via Pyramid Blending

2.1 Approach

(1) Create Gaussian pyramids for the two images and the mask: function [gaussout] = gauss_pyramid (im, levels). Essentially this uses reduce for a total of levels times. Each image in one scale is half the size of the previous scale First scale is simply the original image.

(2) Create the Laplacian pyramids for the two images: function [pyramid] = lapl_pyramid(gauss_pyr). Take an image in a Gaussian pyramid at a smaller scale (i+1) and expand it. Then subtract it with the i'th scale. Image at the end of the pyramid is simply the last image in the Gaussian pyramid.

(3) Blend the two Laplacian pyramids together: function [blended_pyr] = blend (pyr_white, pyr_black, pyr_mask). Given the Laplacian pyramids of two images as well as the mask image, for each scale apply an alpha blending equation.

(4) Create final blended image: function [col_output] = collapse(lapl_pyr). Given a (blended) Laplacian pyramid, starting from the smallest scale, expand it, then add this previous scale. Repeat this until we get to the original scale and this is the output.

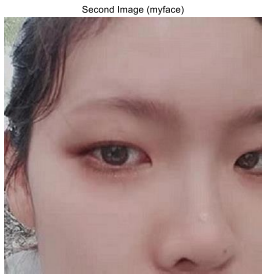
(5) Key procedures:

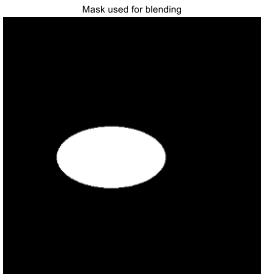
(i) Reduce: 1) Gaussian kernel (n=5, sigma=1.5) convolution is performed on the image. 2) Divide out all even rows and columns.

(ii)Expand: 1) Enlarge the image twice in each direction, and fill the new rows and columns with 0. 2) The same kernel (multiplied by 4) was used to convolve with the enlarged image to obtain an approximation of the added pixel.

2.2 Results

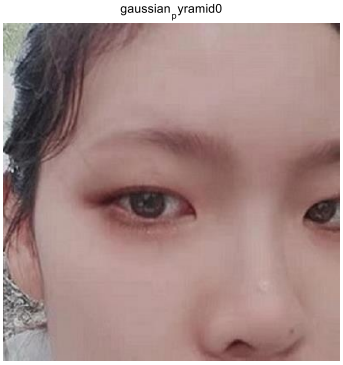
(i) the original images (my face, my palm and the binary mask)





(ii) three images showing the images from the low, mid and high levels of the Gaussian and Laplacian pyramids in two rows. Top row showing the Gaussian and the bottom row showing the Laplacian pyramid images.

(a) "My face of the Gaussian and Laplacian pyramids":



laplacian$_p$yramid

laplacian$_p$yramid

laplacian$_p$yramid

(b) “My hand of the Gaussian and Laplacian pyramids”:

gaussian$_p$yramid0

gaussian$_p$yramid

gaussian$_p$yramid

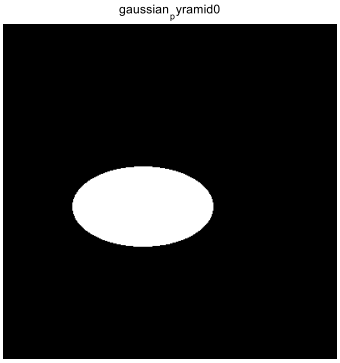
laplacian$_p$yramid

laplacian$_p$yramid

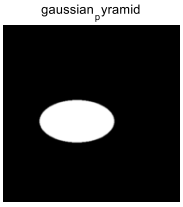
laplacian$_p$yramid

(c) “My mask of the Gaussian and Laplacian pyramids”:

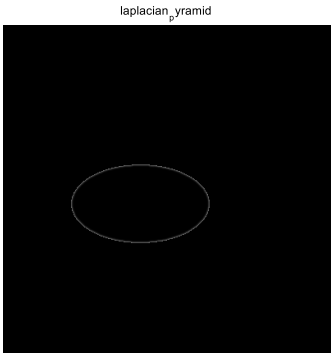
gaussian$_p$yramid0

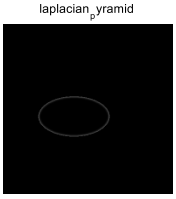


gaussian$_p$yramid



gaussian$_p$yramid





(iii) The belnded image at the original resolution of the face and palm images.