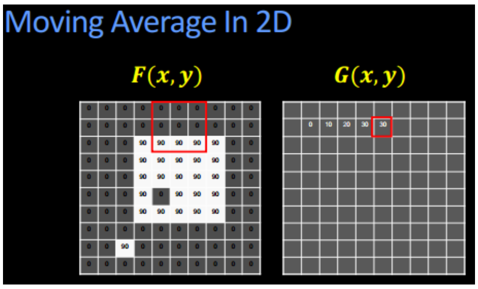
2A-L2-Filtering

1. 2017/11/03 20:23
2. Sum
    a. remove noise in an image - Correlation Filtering
        i. average filter
            1. average assumptions
        ii. Gaussian filter
3. Remove the Noise
    a. intro
        i. if we know the noise, we can certainly subtract it but we don't know. Here are some ways to realize it.
    b. Alternative 1
        i. Replace each pixel with an average of all the values in its neighborhood - a moving average:
        ii. Averaging Assumptions
            1. 1. The "true" value of pixels are similar to the true value of pixels nearby.
            2. 2. The noise added to each pixel is done independently
                a. so the sum is 0;
    c. Alternative 2: Weighted Moving Average
        i. generate a smoother result than alternative 1
        ii. The basic idea is that nearby pixels have similar true underlying values. the closer a pixel is to some reference pixel, the more similar it would be. So the more it should contribute to an average.
        iii. the To do the moving average computation the number of weights should be Odd and symmetric - makes it easier to have a middle pixel
        iv. the sum of the weight should be 1
4. Moving Average In 2D
    a. use a squared region to calculate the average

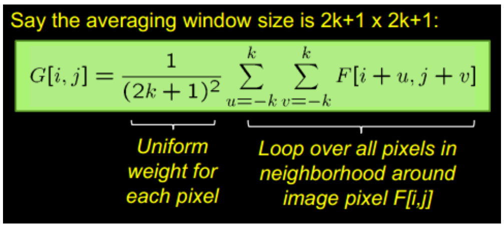


b. Questions

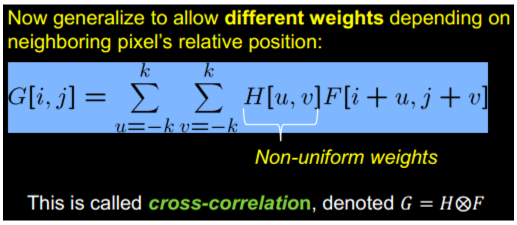
i. how to deal with the edge?

c. this is called Correlation Filtering

i. uniform weights



ii. non-uniform weights



iii. The filter “kernel” or “mask” H[u,v] is the matrix of weights in the linear combination.

1. this kernel is different from the one in ML

1. what makes a good kernel?

a. Alternative 1: Averaging Filter - uniform one

i. the result is really bad

ii. what’s the problem

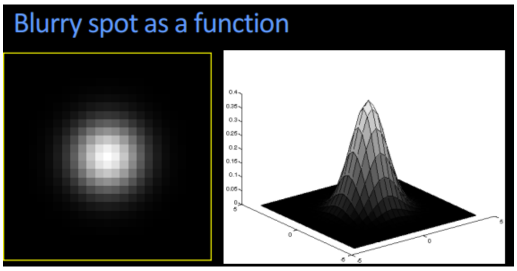
1. squares aren’t smooth

2. And filtering an image with a filter that is not “smooth” seems wrong if we’re trying to “blur” the image.

iii. analogy

1. think about what a single spot of light

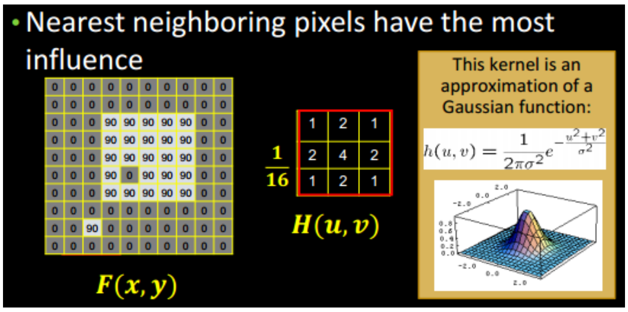
viewed by an out of focus camera would look like.



2. so To blur a single pixel into a "blurry" spot, we would need to need to filter the spot with Something that looks like a blurry spot - higher values in the middle, falling off to the edges

a. Alternative 2 - Gaussian Filter

i. we get sth so much better - no clear edges in the image



ii. Key

1. nearest neighboring pixels have the most influence

iii. called: circularly symmetric Gassian function or isotropic

iv. formula

$$h(u,v) = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$$

v. The amount of smoothing is define by the Variance or Standard Deviation (sigma), the only parameter in the isotropic funciton

vi. another influential parameter, the size of the kernel/square

1. the bigger one with the same sigma has better performance
2. the kernel has to be big enough, it’s a default parameter. So often a "big kernel" means a "big sigma" actually

vii.

1. Matlab code

```
sigmav = [3, 13, 23, 33];
for sigma = sigmav
  hsize = 31;
  h = fspecial('gaussian', hsize, sigma);
  out = imfilter(nmona, h);
  figure(sigma);imshow(out);
end
```

7. Quiz: Remove Noise

i. the Gaussian filter can smooth/blur the image but it affect the original image, too. so you don't get back exactly the same as the original one.

ii. even through the smoothed image doesn't look good virtually, but it benefits the image process a lot.

8. Quiz: Gaussian Filter Quiz

a. When filtering with a Gaussian, which is true:

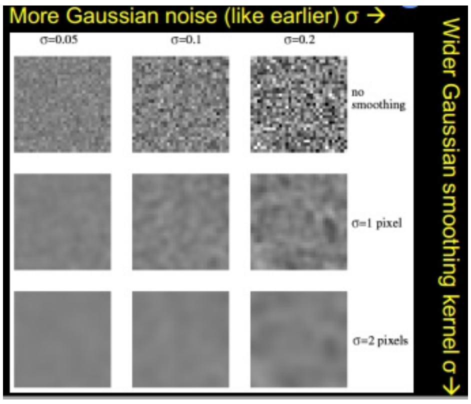
i. The sigma is most important – it defines the blur kernel’ s scale with respect to the image

ii. Altering the normalization coefficient does not effect the blur, only the brightness.

9. Keeping the Two Gaussians Straight

a. when talking about Gaussian filter, sigma defines a width of a the Gaussian filter.

b. when talking about Gaussian noise, sigma defines the variance of a noise function or the value of the noise. The bigger the noise sigma was the more likely that large values of noise can be created.


More Gaussian noise (like earlier) σ →
σ=0.05
σ=0.1
σ=0.2
no smoothing
σ=1 pixel
σ=2 pixels
Wider Gaussian smoothing kernel σ →

10.