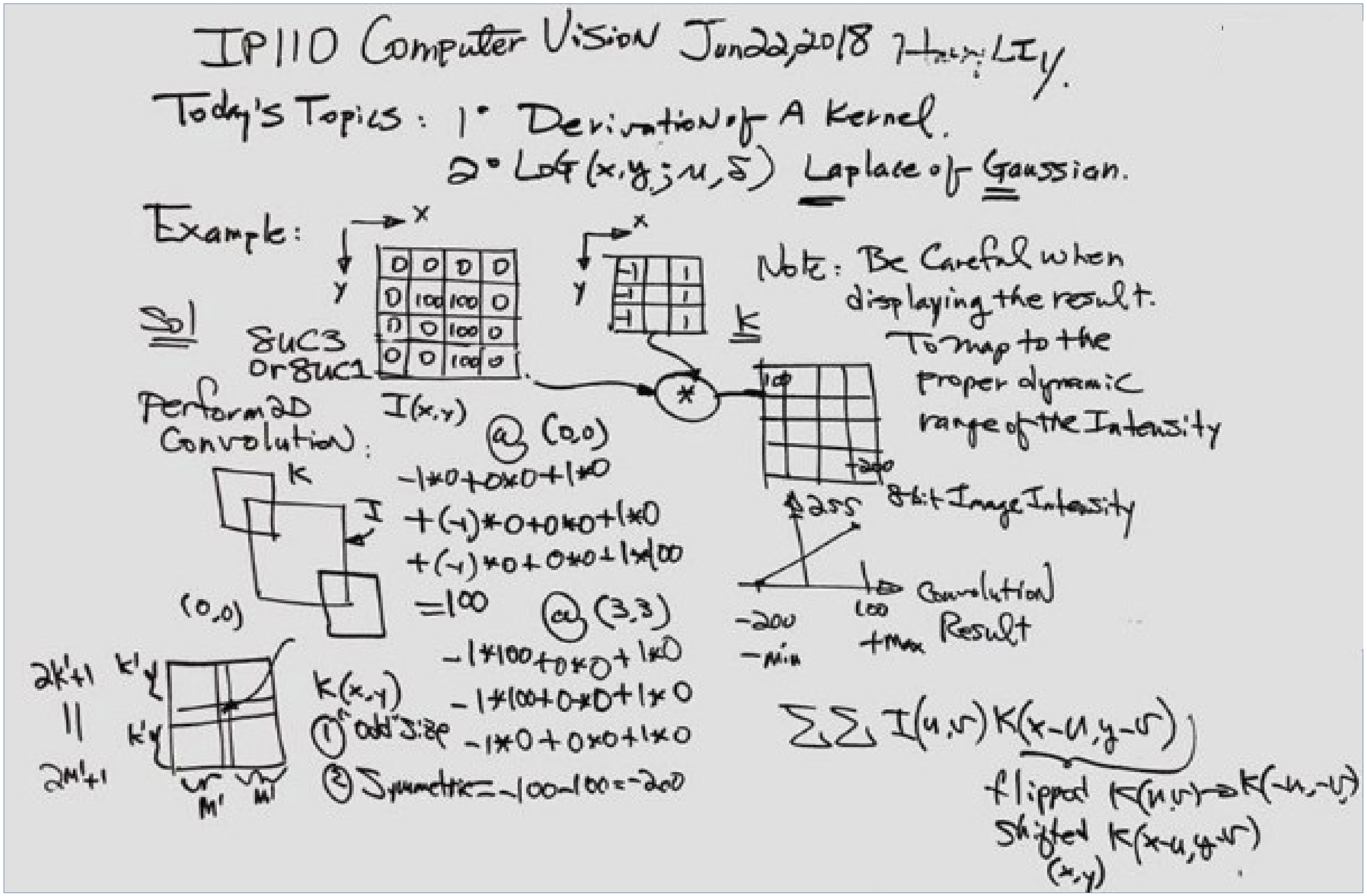
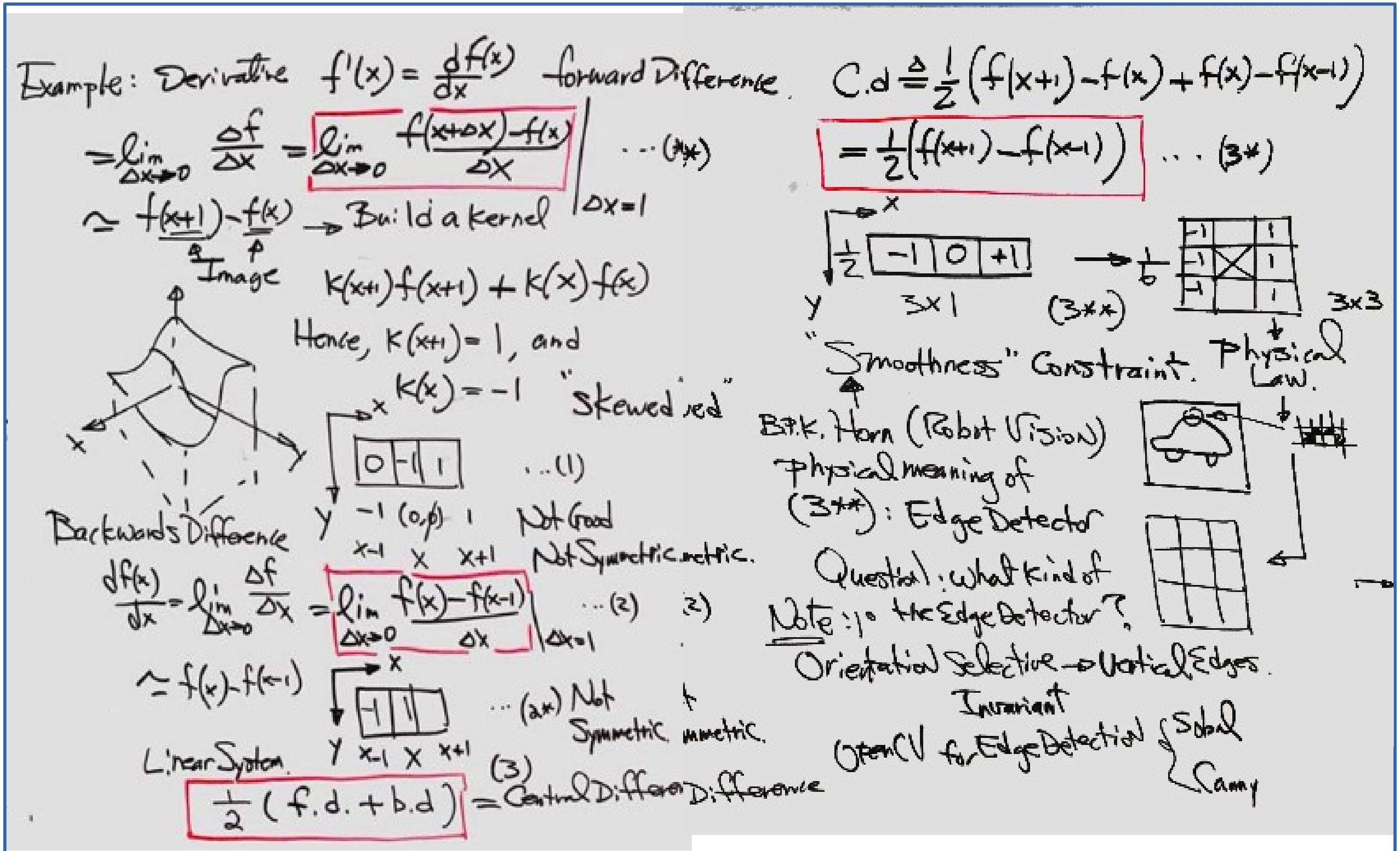
6-22-18 Kernel Design Background



3. The convolution result should be mapped to the dynamic range of the display device, see Figure illustration here, e.g., map to 0 to 255.

1. Computer 2D convolution with padding 0s outside the image, or position the kernel such a way so all its coefficients are mapped inside the image; 2. Kernel design guidelines, e.g., cover all the image pixels in a even/unbiased fashion, so leads to the size of the kernel is always K-by-K, where K is an odd number (2p+1)-by-(2p+1) so the kernel is always centered.

6-22-18 From Derivative To Kernels

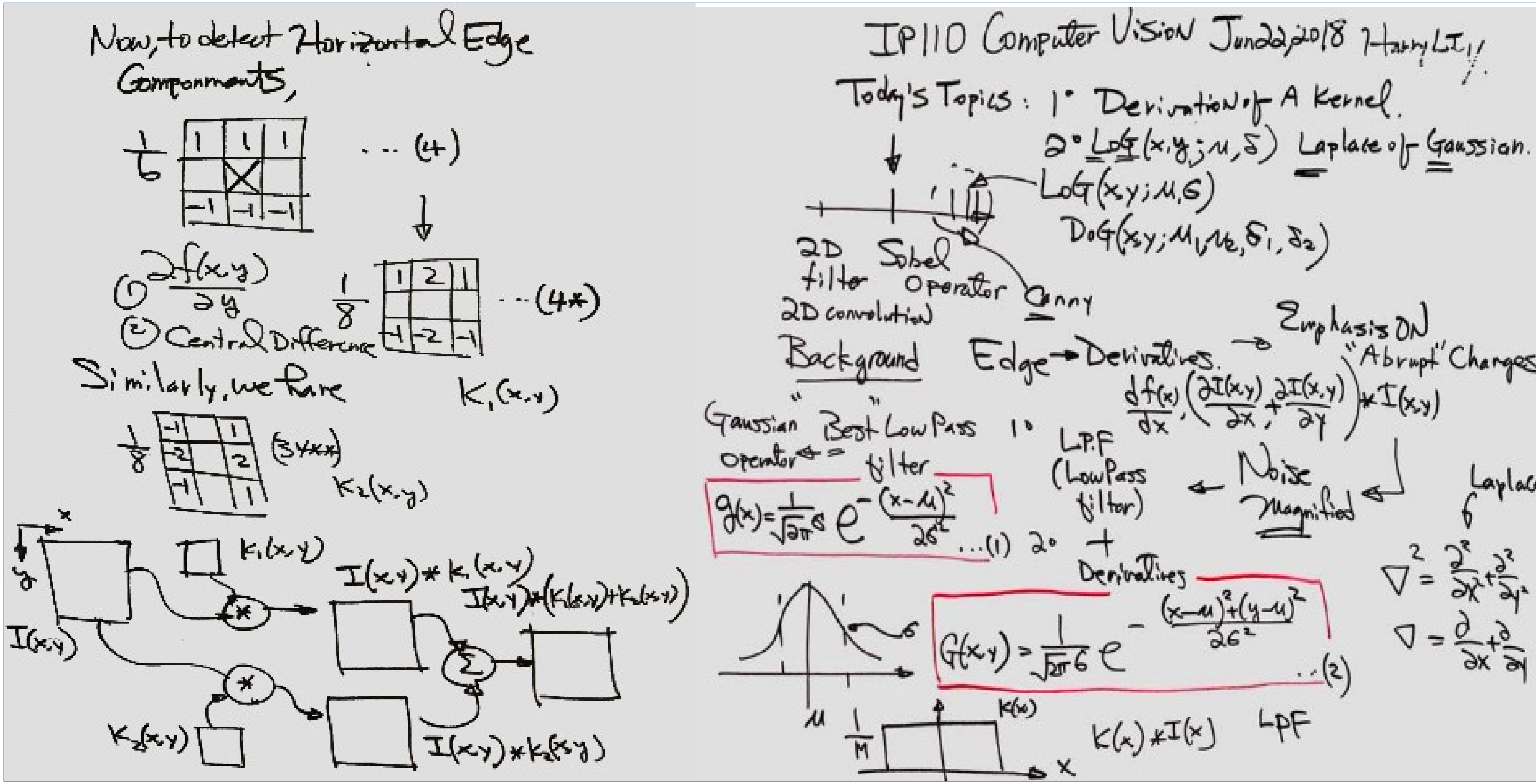


4. Why is an edge detector derived from derivatives? 5. What is orientation selective, or orientation invariant edge detector? 6. what is "smoothness" constraint? What role does it play in developing K-by-K kernels?

1. Backwards difference, forward difference, central difference calculation? 2. Why is backwards difference and forward difference biased in the sense of image processing? 3. How do you connect the finite difference formula of derivative computation to convolution definition? 4. How do you map convolution to kernel based image processing operation?

Harry Li, Ph.D. 2018

6-22-18 Sobel and LoG Edge Detectors



1.How to define the scaling factor of an edge detector? Such as 1/6, 1/8 etc.? 2. How do you use multiple orientation selective kernels to generate edge map as if it is coming from an orientation invariant kernel? 3. What are the Gaussian functions g(x) and g(x,y) ? 4. why is Gaussian function performing low pass operation? What is the difference between Gaussian function based low pass filtering and averaging? 5. Why is the Gaussian function g(x,y) best in mathematic sense? (Taylor expansion theory).

Harry Li, Ph.D. 2018

6-22-18 LoG(x,y) Edge Detectors

Recall, Derivation of A Kernel.

$\frac{d}{dx}f(x)$, $\frac{\partial}{\partial x}f(x,y)$, $\frac{\partial}{\partial y}f(x,y)$ "Edmap" $\nabla I(x,y)$

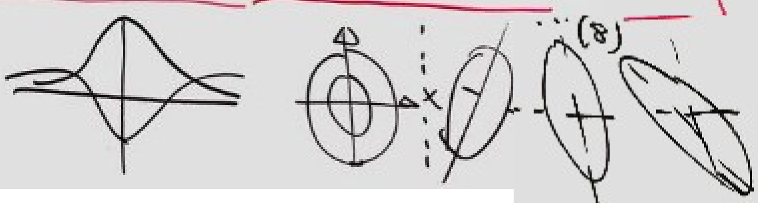
$$\boxed{\left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\right)G(x,y) * I(x,y) \cdots(3)}$$

$$\frac{\partial}{\partial x}G(x,y) = \frac{\partial}{\partial x}\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2+(y-\mu)^2}{2\sigma^2}} \cdots)$$

$\frac{\partial}{\partial y}G(x,y) \cdots(5)$ then, $\frac{\partial}{\partial y^2}G(x,y)$ $\|$

$$\frac{\partial}{\partial x^2}G(x,y) = \frac{\partial}{\partial x}\left(\frac{\partial}{\partial x}G(x,y)\right) \cdots(6), \quad \frac{\partial}{\partial y}\left(\frac{\partial}{\partial y}G(x,y)\right) \cdots(7)$$

$$\boxed{\nabla^2 G(x,y) = \left(\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}\right)G(x,y) = \frac{x^2+y^2-2\sigma^2}{\sqrt{2\pi}\,\sigma^5} e^{-\frac{x^2+y^2}{2\sigma^2}}} \cdots(8)$$

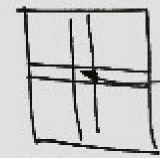


Note:

1° OpenCV. 2D filter w/ Kernel user defined.

(1.1) $\frac{1}{6}\begin{bmatrix} -1 & & 1 \\ -1 & & 1 \\ -1 & & 1 \end{bmatrix}$ (1.2) $\frac{1}{8}\begin{bmatrix} -1 & & 1 \\ -2 & & 2 \\ -1 & & 1 \end{bmatrix}$

2° OpenCV. From Eq(8) 3x3, 5x5, 7x7, ..., 33x33 etc.
LoG(x,y) $\mu_x=\mu_y=0$
$\sigma_x=\sigma_y=1.8\sim2.4$



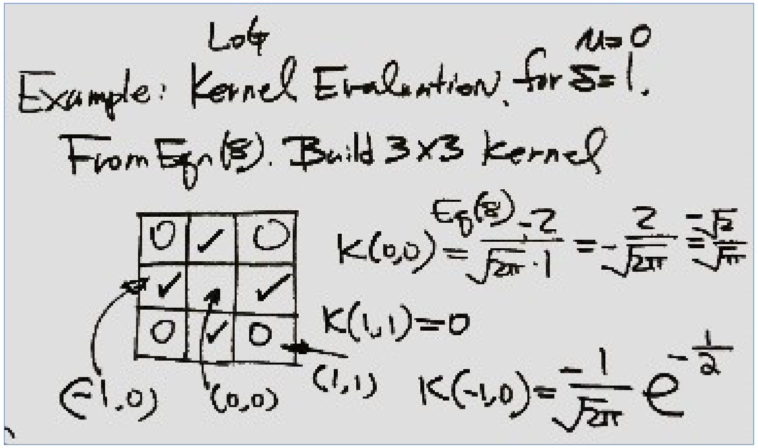
3° OpenCV. GaussianBlur Function. → Compare to Pyramid.

4° Edge Detection (Google: LoG(x,y) OpenCV for Edge Detection)

Next Lecture: Colour Segmentation + Binary Image Processing (Contours, moments).

(1) What is Laplace operator? (2) how do you combine Laplace operator with Gaussian to form LoG(x,y)? (2) Is LoG(x,y) orientation invariant edge detector? (3) How do you select meaningful sigma x and sigma y for LoG(x,y) convolution?

6-22-18 LoG(x,y) Coefficients



(1) How do you compute LoG(x,y) kernel coefficients? (2) Is LoG(x,y) symmetric? Based on its symmetric property, can you simplify the computation of LoG(x,y) coefficients? Give an example to demonstrate this symmetric property.