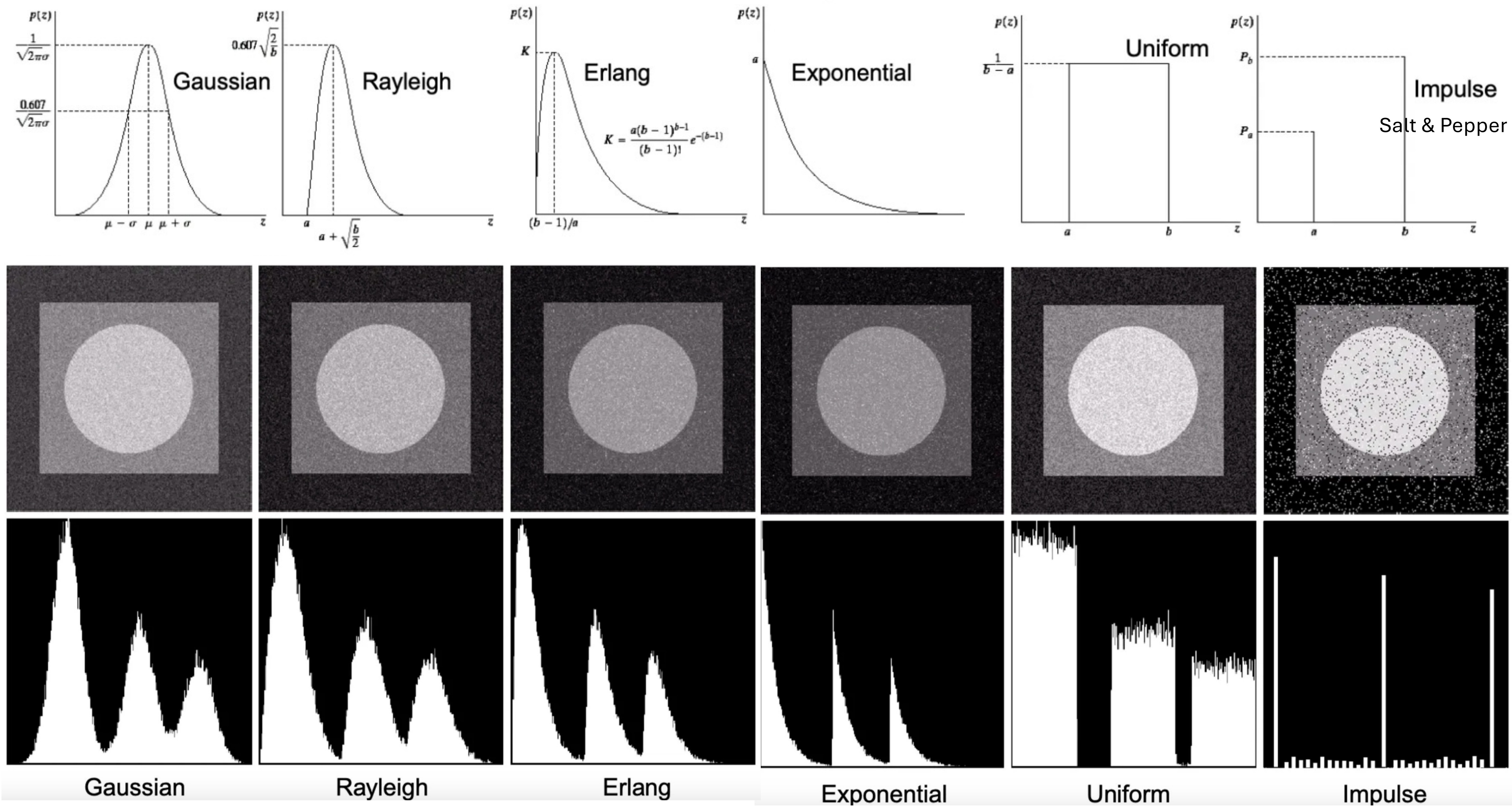
Lecture 6
Image Restoration/Inverse filters

ECE 1390/2390

Learning Objectives:

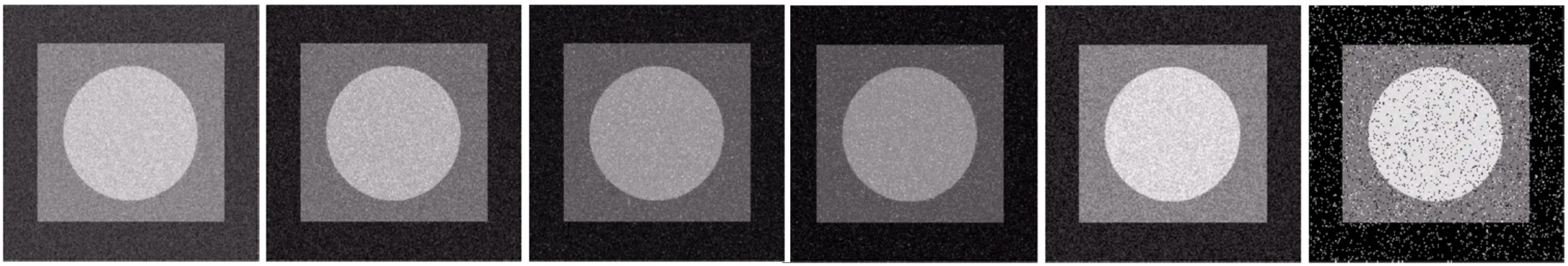
- Types of noise/artifacts
- Inverse filter
- Wiener filter



Gaussian
Rayleigh
Erlang
$K = \frac{a(b-1)^{b-1}}{(b-1)!} e^{-(b-1)}$
Exponential
Uniform
Impulse
Salt & Pepper
Gaussian
Rayleigh
Erlang
Exponential
Uniform
Impulse

Digital Noise

- Image sensor is broken
- Affected by external factors;
- Low levels of light
- High levels of light (saturation)
- Heating of sensor
- Interference from sensor to DAC.



<u>Gaussian Noise</u>

- Additive (positive & negative)
- Transmission noise
- Thermal effects
- Electronic noise

To fix:

- Mean filter
- Smoothing

<u>Rayleigh Noise</u>

- Additive (positive only)
- Transmission noise
- Thermal effects
- Electronic noise

To fix:

- Mean filter (positive bias)
- Smoothing (positive bias)

<u>Salt & Pepper Noise</u>

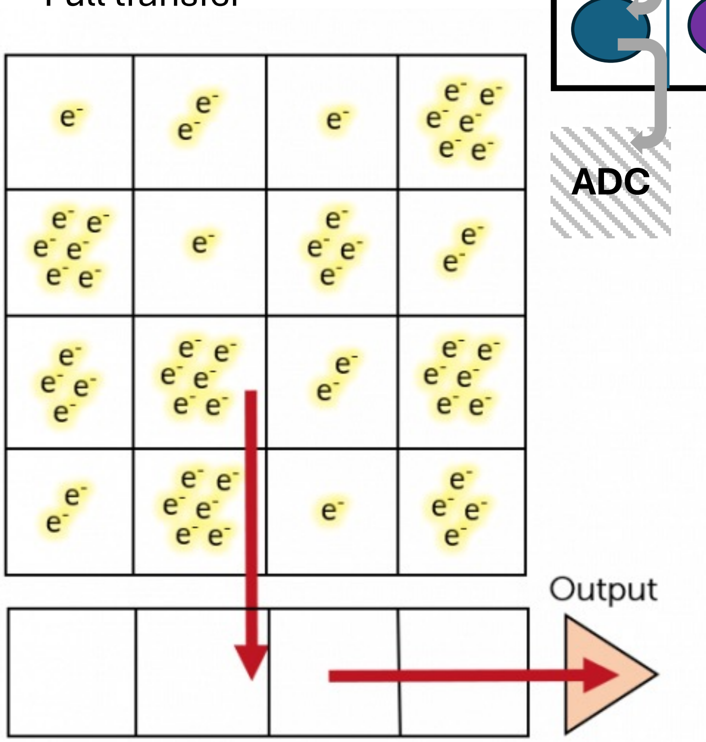
- Defective pixels

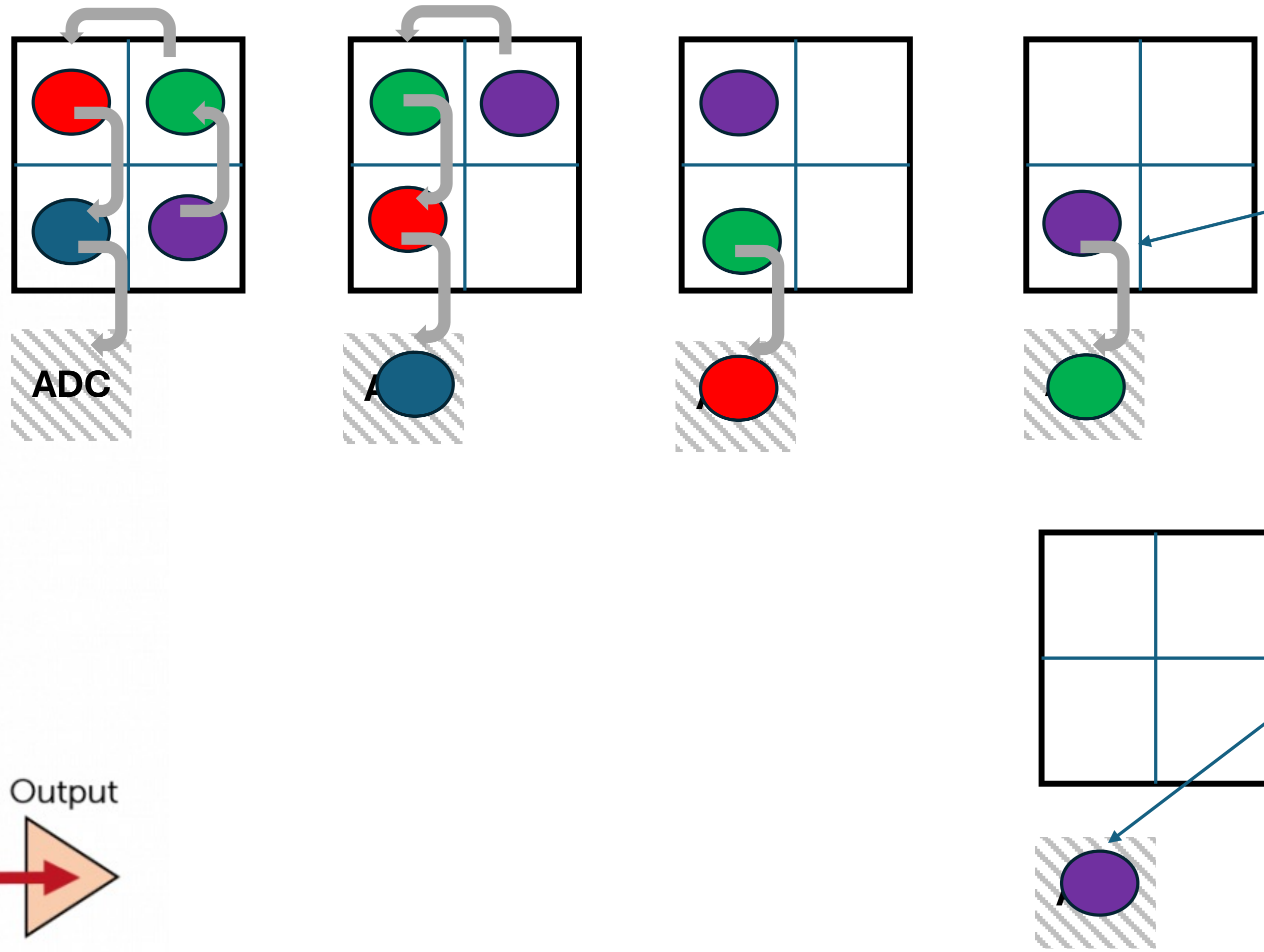
To fix:

- Median filter
- Center ignore filter
- Contra-harmonic filter

CCD Cameras

Full transfer





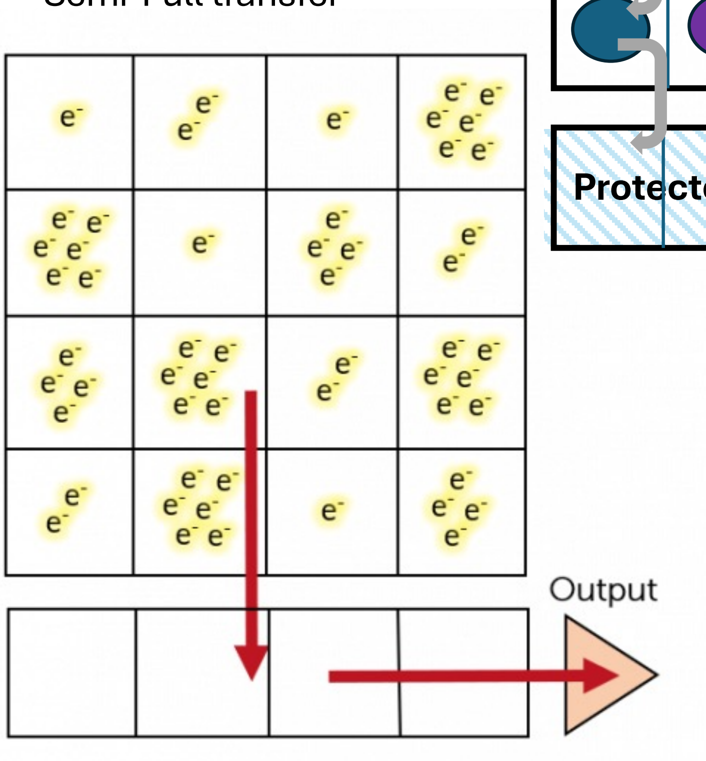
Everything passed through this pixel

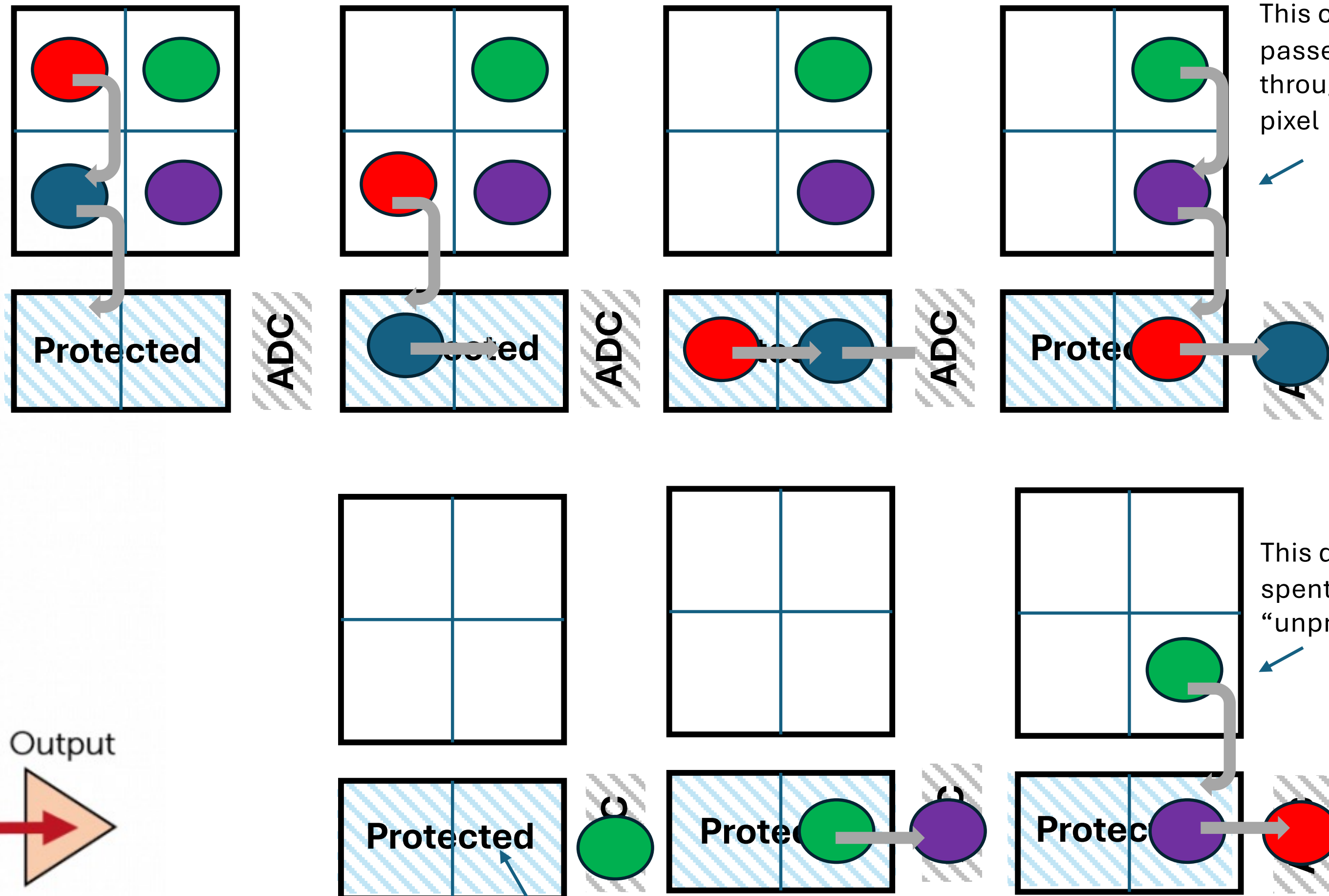
This data point spent longest “unprotected”

https://www.princetoninstruments.com/learn/camera-fundamentals/ccd-the-basics

CCD Cameras

Semi-Full transfer





This column all passed through this pixel

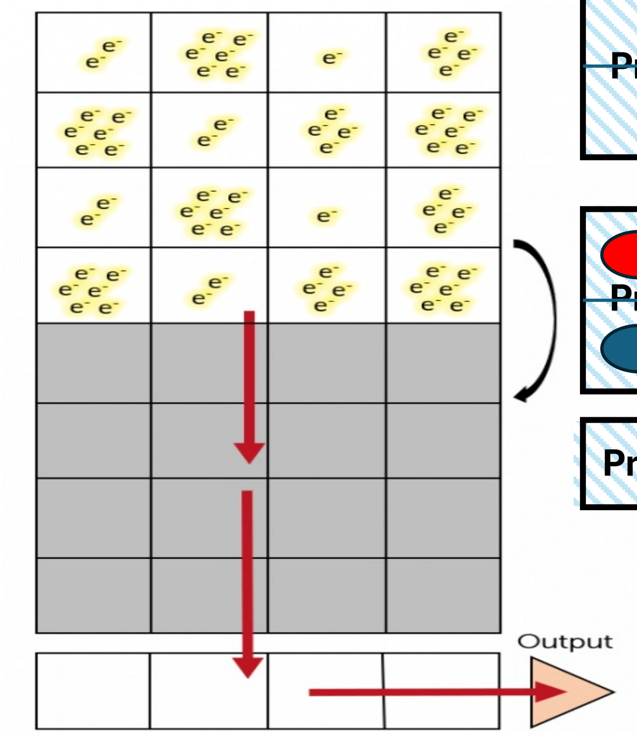
This data point spent longest “unprotected”

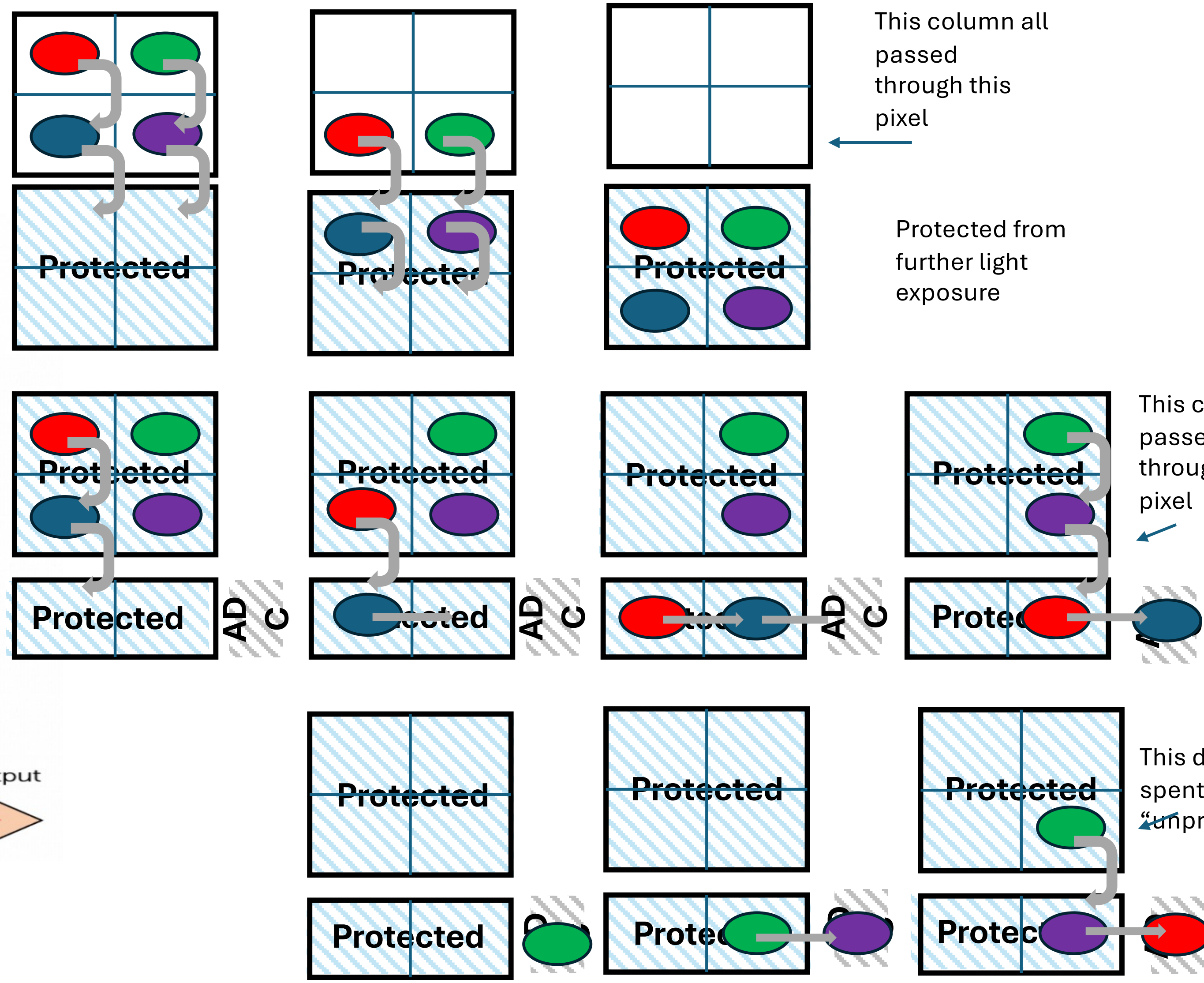
Every data point went through here

https://www.princetoninstruments.com/learn/camera-fundamentals/ccd-the-basics

CCD Cameras

Frame transfer





This column all passed through this pixel

Protected from further light exposure

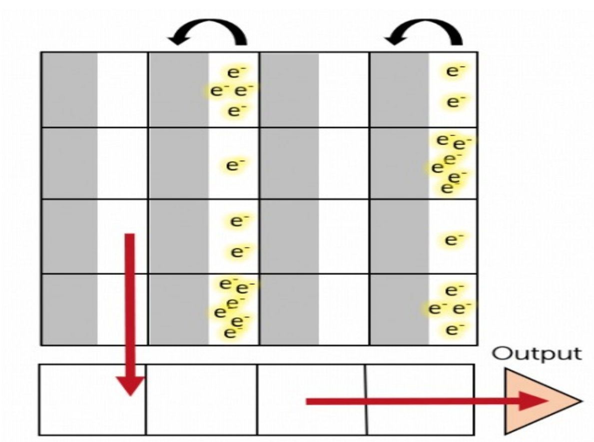
This column al passed through this pixel

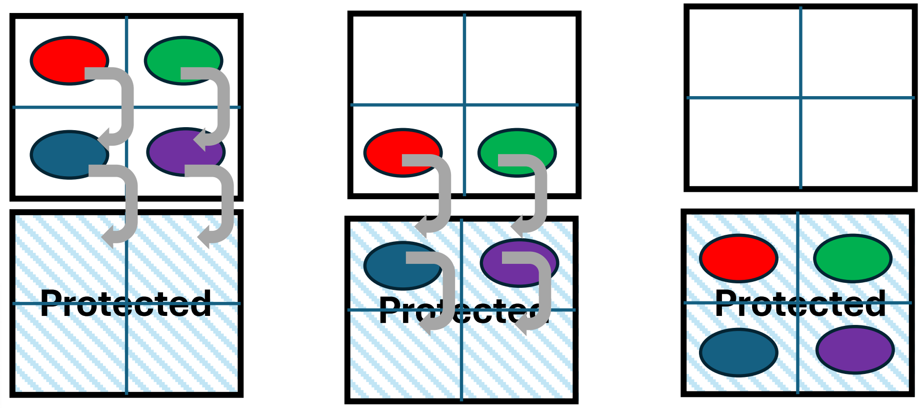
This data point spent longest "unprotected"

https://www.princetoninstruments.com/learn/camera-fundamentals/ccd-the-basics

CCD Cameras

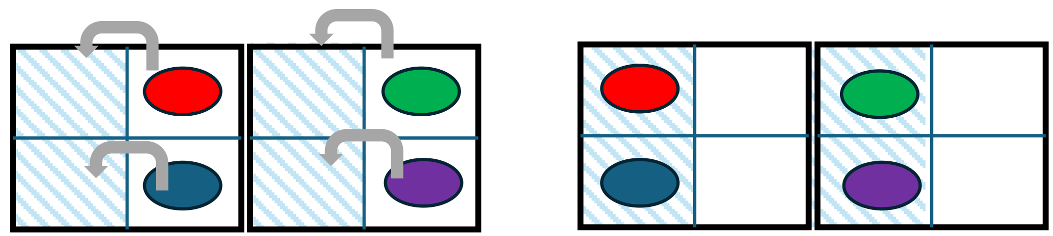
Inter-line transfer





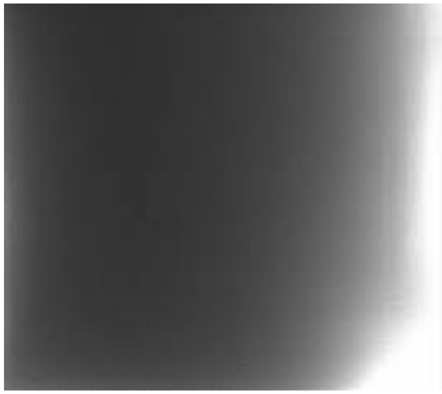
This column all passed through this pixel

Protected from further light exposure



Protected from further light exposure in one step

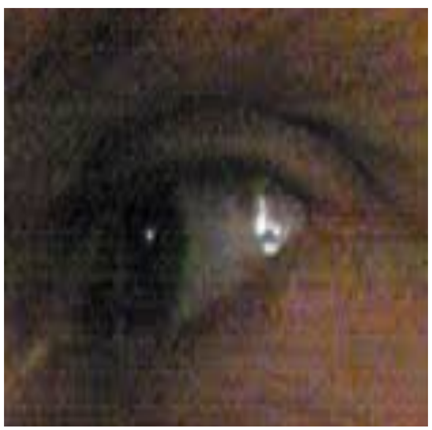
https://www.princetoninstruments.com/learn/camera-fundamentals/ccd-the-basics



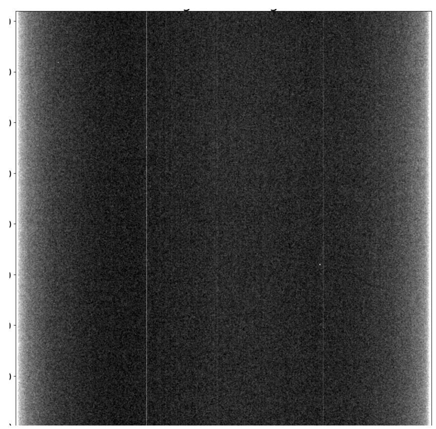
Dark noise pile-up

Dragging data through
a bright intensity

More subtle dragging
through bright intensity



Horizontal banding



Single column
vertical stripes

Morphological filters

Arithmetic Mean

- Smooths image. Removes noise and features of interest

$$m_{arith} = \frac{1}{n}\sum_{i=0}^{n} x_i$$

Geometric Mean

- Removes noise but preserves features better than arithmetic mean

$$m_{geom} = \sqrt[n]{\prod_{i=0}^{n} x_i}$$

Harmonic Mean

Good for salt noise (but not pepper)
Works well for other types of noise

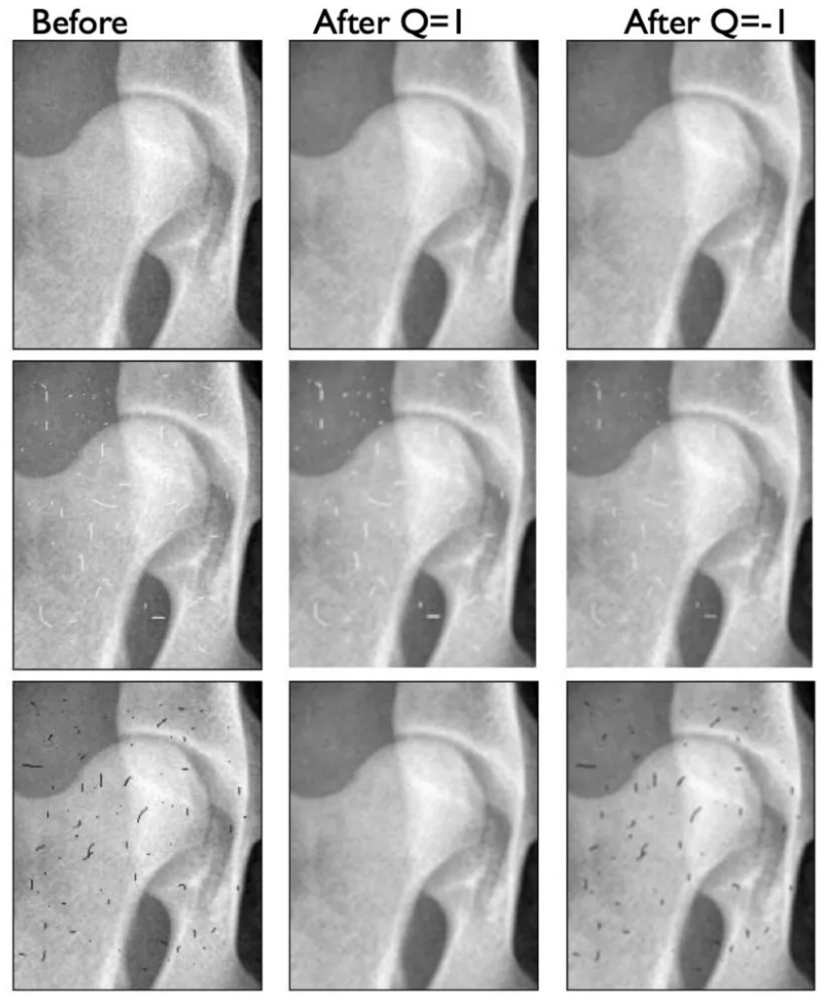
$$m_{harm} = n\frac{1}{\sum_{i=0}^{n}\frac{1}{x_i}}$$

Morphological filters

Contra-harmonic mean

$$m_{contraharm} = \frac{\sum_{i=0}^{n} (X_i)^{Q+1}}{\sum_{i=0}^{n} (X_i)^{Q}}$$

If Q > 0, Removes pepper noise
If Q < 0, Removes salt noise



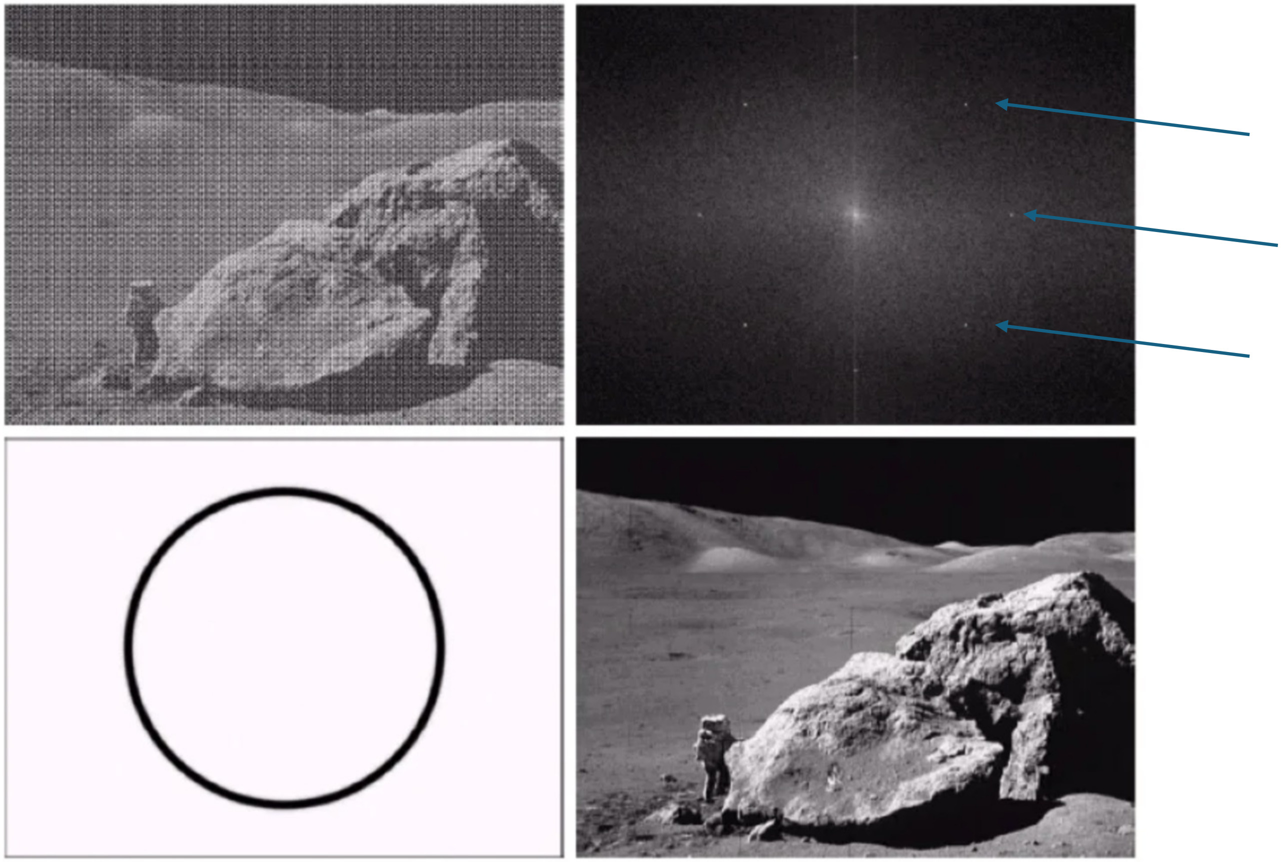
Morphological filters

- **Median filter:** Excellent at noise removal, without the smoothing effects that can occur with other smoothing filters. It is particularly good when salt and pepper noise is present.
- **Max and min filter:** Max filter is good for pepper noise and min filter is good for salt noise.
- **Midpoint filter:** This filter calculates (min+max)/2. It is good for Gaussian and uniform noise.
- **Alpha trimmed mean filter:** The *d/2* lowest and *d/2* highest grey levels can be deleted. So *gr(s, t)* represents the remaining $mn - d$ pixels

$$\hat{f}(x,y) = \frac{1}{mn-d} \sum_{(s,t)\in S_{xy}} g_r(s,t)$$

Image corrupted by sinusoidal noise

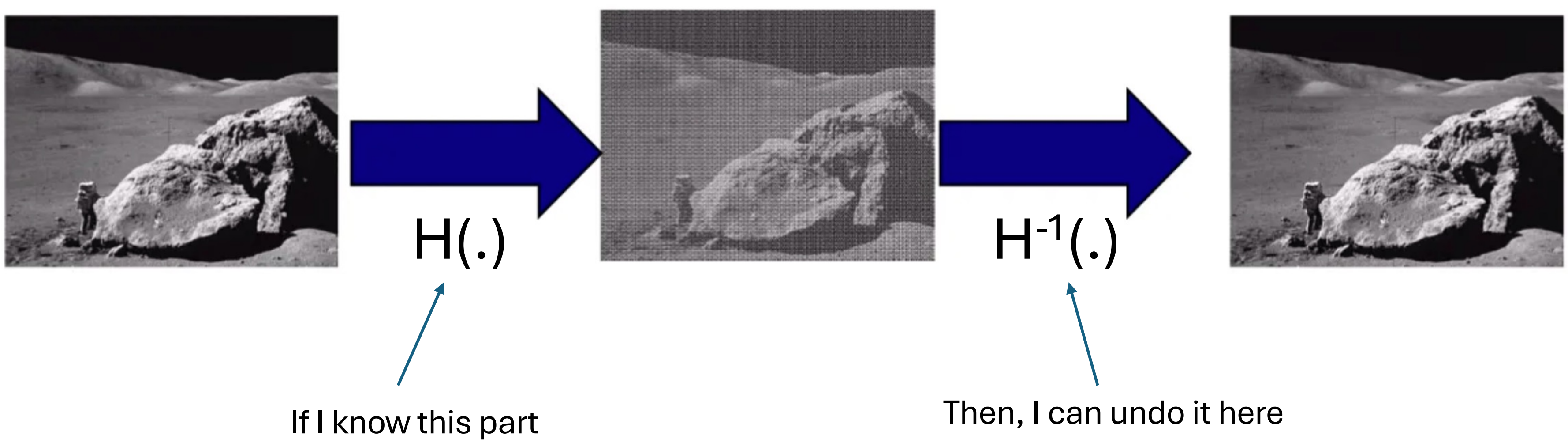
Fourier spectrum of corrupted image



Butterworth band reject filter

Filtered image

Inverse filters



Inverse filters

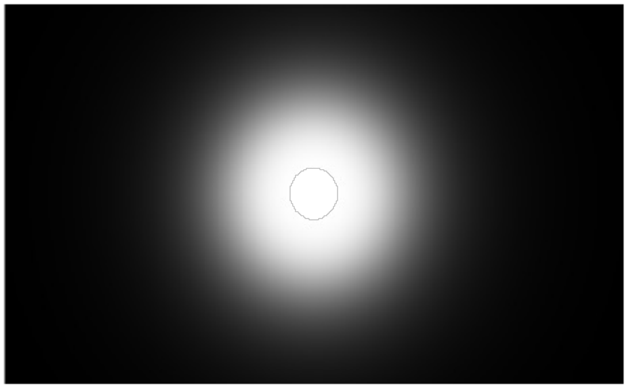
$y * h$

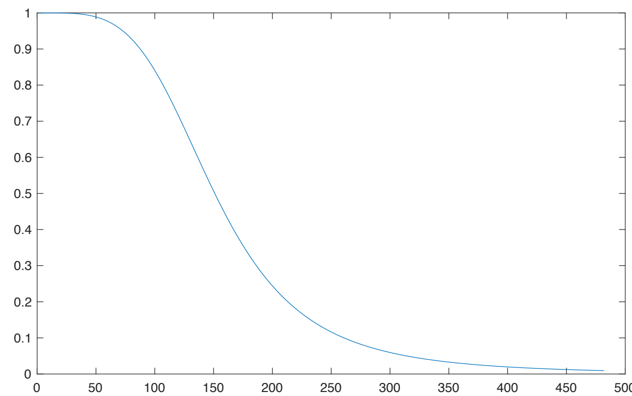
$Yf \cdot Hf$

$(Yf \cdot Hf) \cdot Hf^{-1}$

$$Yf = DFT(y)$$

$$\mathrm{H}f = DFT(h)$$

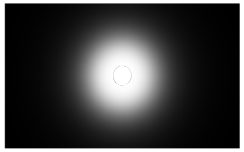




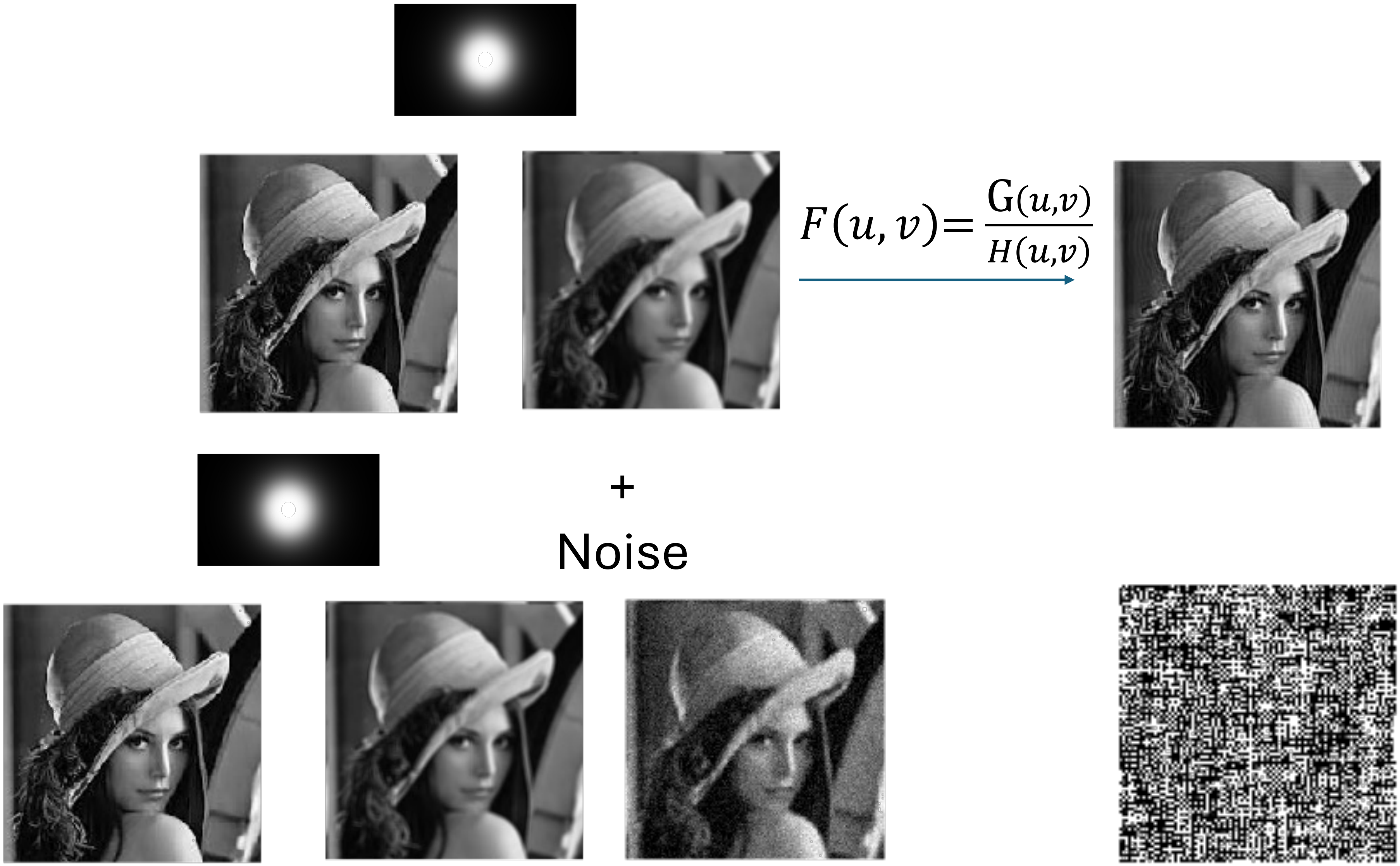
$$g(x, y) = h(x, y) * f(x, y)$$

$$\mathrm{G}(u, v) = H(u, v) \cdot F(u, v))$$

$$F(u, v) = \frac{\mathrm{G}(u,v)}{H(u,v)}$$



$$F(u,v)=\frac{G(u,v)}{H(u,v)}$$

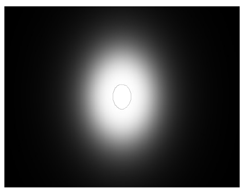

$F(u,v) = \frac{G(u,v)}{H(u,v)}$
+
Noise

Wiener filter

Minimum Mean Least-Squares Error Filter

$$g(x,y) = h(x,y) * f(x,y) + n(x,y)$$

$$G(u,v) = H(u,v) \cdot F(u,v) + N(u,v)$$



+

Noise

$$F(u,v) = \frac{G(u,v)}{H(u,v)} - \frac{N(u,v)}{H(u,v)}$$

$$F(u,v) = \frac{G(u,v)}{H(u,v)} - \frac{N(u,v)}{H(u,v)}$$

Wiener filter

Minimum Mean Least-Squares Error Filter

$$\arg min \sum \left(f(x,y) - \hat{f}(x,y)\right)^2$$

Assumes $n \in N(0, \sigma^2)$

$$\hat{F}(u,v) = \left[\frac{H^*(u,v) S_f(u,v)}{S_f(u,v)|H(u,v)|^2 + S_n(u,v)}\right] G(u,v)$$

$S_n(u,v)$ Power Spectrum of noise

$S_f(u,v)$ Power Spectrum of original image

Wiener filter

Minimum Mean Least-Squares Error Filter

$$\arg min \sum \left(f(x,y) - \hat{f}(x,y)\right)^2$$

Assumes $n\ is\ mean\ zero$

$$\hat{F}(u,v) = \left[\frac{H^*(u,v)S_f(u,v)}{S_f(u,v)|H(u,v)|^2 + S_n(u,v)}\right] G(u,v)$$

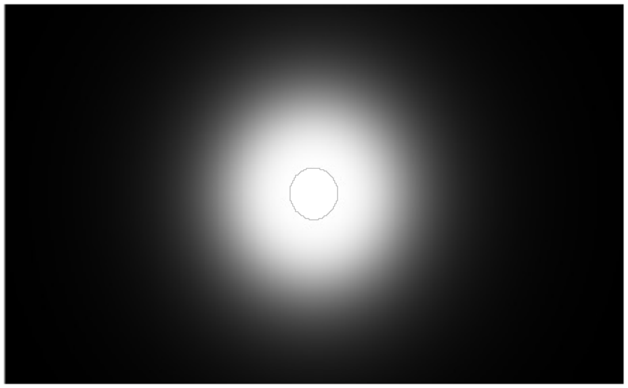
$$\hat{F}(u,v) = \left[\frac{H^*(u,v)}{|H(u,v)|^2 + \frac{S_n(u,v)}{S_f(u,v)}}\right] G(u,v)$$

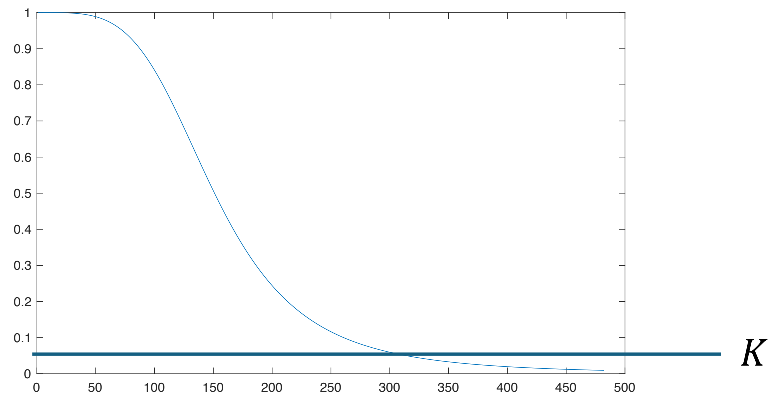
$S_n(u,v)$ Power Spectrum of noise

$S_f(u,v)$ Power Spectrum of original image

$$\hat{F}(u,v) = \left[\frac{1}{H(u,v)} \frac{|H(u,v)|^2}{|H(u,v)|^2 + \frac{S_n(u,v)}{S_f(u,v)}}\right] G(u,v)$$

$$\frac{S_n(u,v)}{S_f(u,v)} = K$$





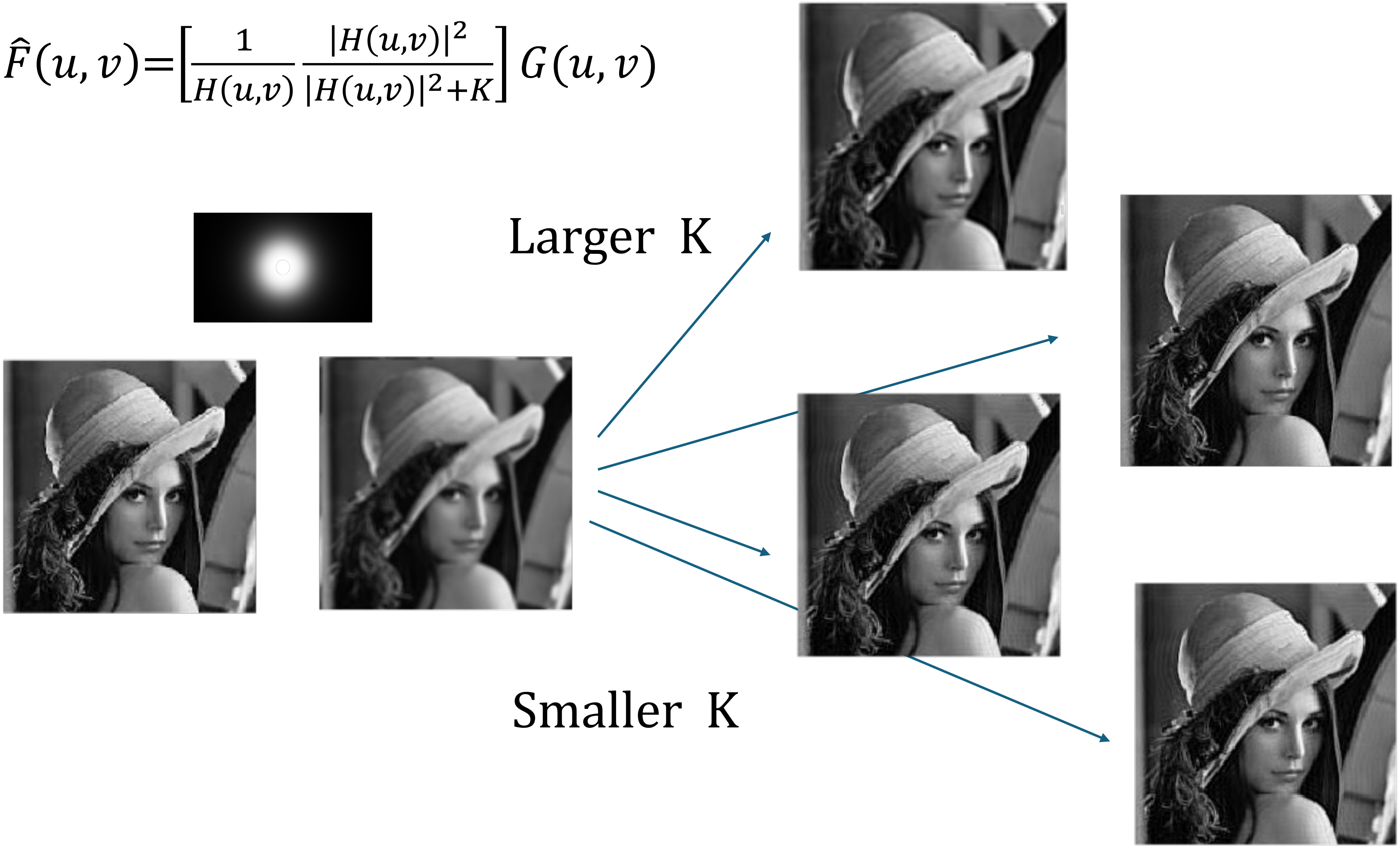
$$g(x,y) = h(x,y) * f(x,y)$$

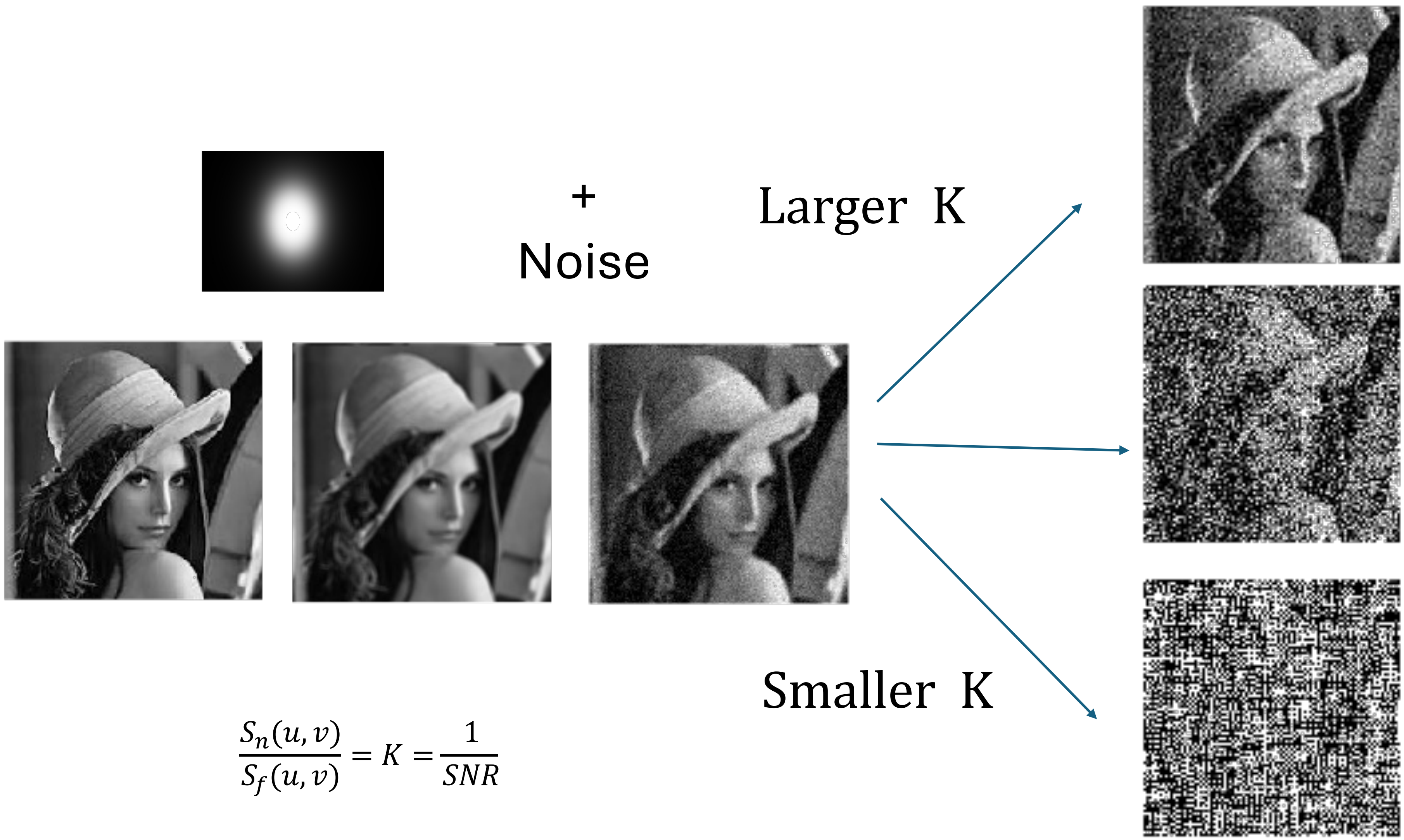
$$\hat{F}(u,v) = \left[\frac{1}{H(u,v)} \frac{|H(u,v)|^2}{|H(u,v)|^2 + K}\right] G(u,v)$$

$$\mathrm{G}(u,v) = H(u,v) \cdot F(u,v))$$

$$y_{recovered} = iDFT(iHF * Yf)$$

$$\hat{F}(u,v)=\left[\frac{1}{H(u,v)}\frac{|H(u,v)|^2}{|H(u,v)|^2+K}\right]G(u,v)$$





$$\frac{S_n(u,v)}{S_f(u,v)} = K = \frac{1}{SNR}$$

Constrained least-squares filtering

$$\hat{F}(u,v)=\left[\frac{1}{H(u,v)}\frac{|H(u,v)|^2}{|H(u,v)|^2+\frac{S_n(u,v)}{S_f(u,v)}}\right]G(u,v)$$

$$\arg min \left|f(x,y)-\hat{f}(x,y)\right|^2+\lambda\cdot\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left[\nabla^2\hat{f}(x,y)\right]^2$$

Constrained least-squares filtering

$$\hat{F}(u,v)=\left[\frac{1}{H(u,v)}\frac{|H(u,v)|^2}{|H(u,v)|^2+\frac{S_n(u,v)}{S_f(u,v)}}\right]G(u,v)$$

$$\arg min\left|f(x,y)-\hat{f}(x,y)\right|^2+\lambda\cdot\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left[\nabla^2\hat{f}(x,y)\right]^2$$

$$\arg min\left|f(x,y)-\hat{f}(x,y)\right|^2+\lambda\cdot\left|p(x,y)\hat{f}(x,y)\right|^2$$

$$p(x,y)=\begin{bmatrix}0 & -1 & 0\\ -1 & 4 & -1\\ 0 & -1 & 0\end{bmatrix}$$

$$\hat{F}(u,v)=\left[\frac{1}{H(u,v)}\frac{|H(u,v)|^2}{|H(u,v)|^2+\lambda\cdot|P(u,v)|^2}\right]G(u,v)$$

https://www.youtube.com/watch?v=MdWDZx6E13U

Inpainting

Inpainting

- Navier-Stokes (cv.INPAINT_NS)

$$\rho \frac{d\vec{v}}{dt} = -\nabla p + \rho \vec{g} + \mu \nabla^2 \vec{v}$$

- Fast Marching method (cv.INPAINT_TELEA)