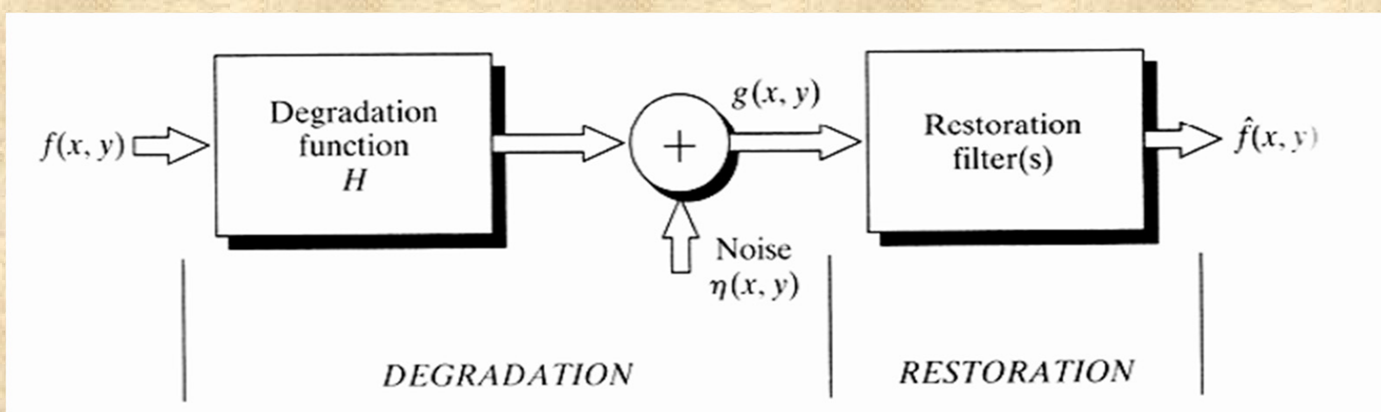
MATHEMATICAL MODEL OF IMAGE DEGRADATION

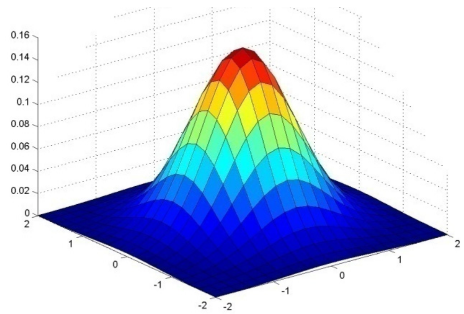
$$g(x,y) = H\{ f(x,y)\} + n(x,y)$$

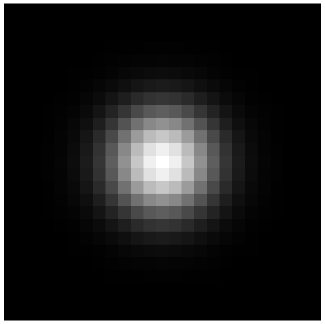
$$H(u,v).F(u,v) = G(u,v)$$

$$H_s(u,v) = G_s(u,v) \big/ F_s(u,v) \qquad \hat{F}(u,v) = G(u,v) \big/ H_s(u,v)$$



Gaussian Kernel

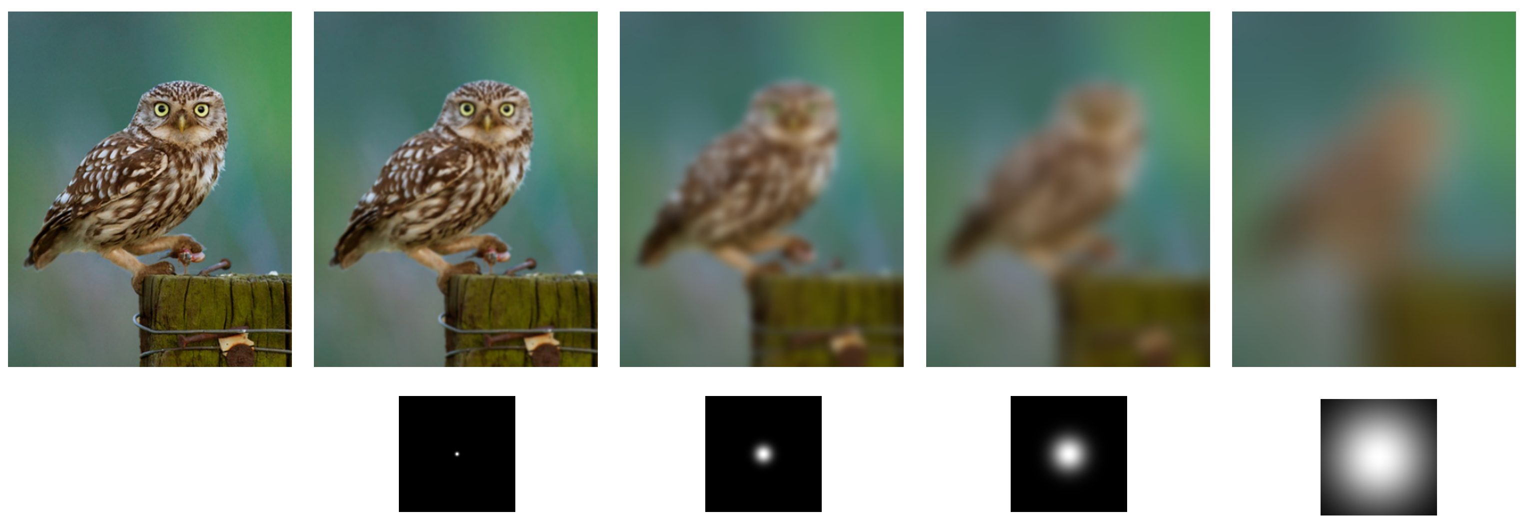




$$G_\sigma = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

Source: C. Rasmussen

Gaussian filters



σ = 1 pixel σ = 5 pixels σ = 10 pixels σ = 30 pixels

Gaussian filter

- Removes "high-frequency" components from the image (low-pass filter)
- Convolution with self is another Gaussian

  – Convolving two times with Gaussian kernel of width σ = convolving once with kernel of width $\sigma\sqrt{2}$

Source: K. Grauman

Sharpening revisited

- What does blurring take away?

–

=

Let’s add it back:

$+ \alpha$

=

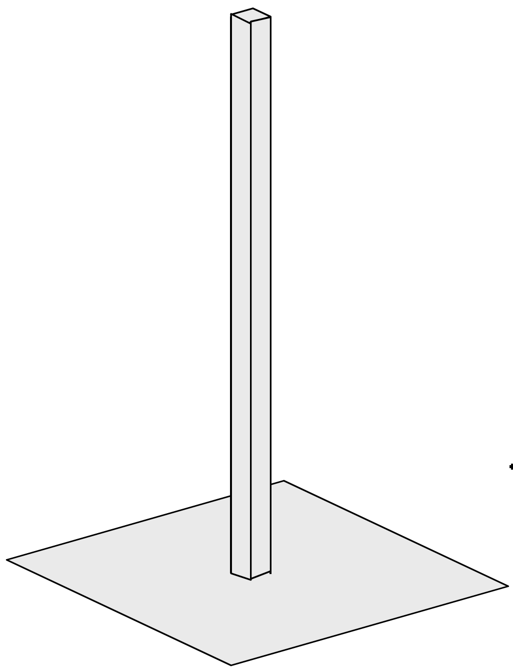
Source: S. Lazebnik

Sharpen filter

$$F + \alpha\,(F - \underbrace{F * H}_{\text{blurred image}})$$

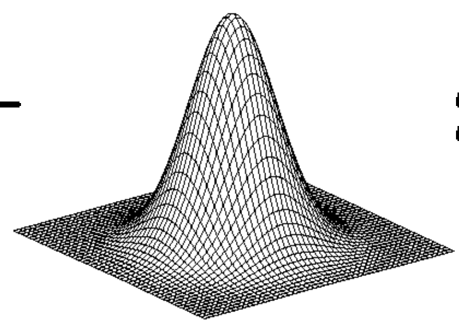
image

unit impulse
(identity)



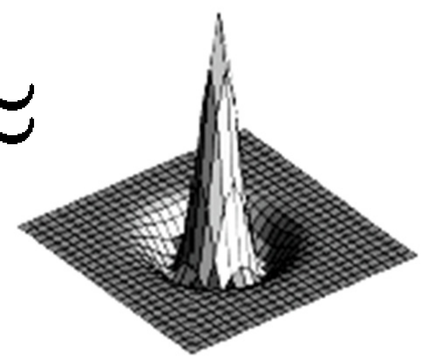
scaled impulse

$-$



Gaussian

$\approx$

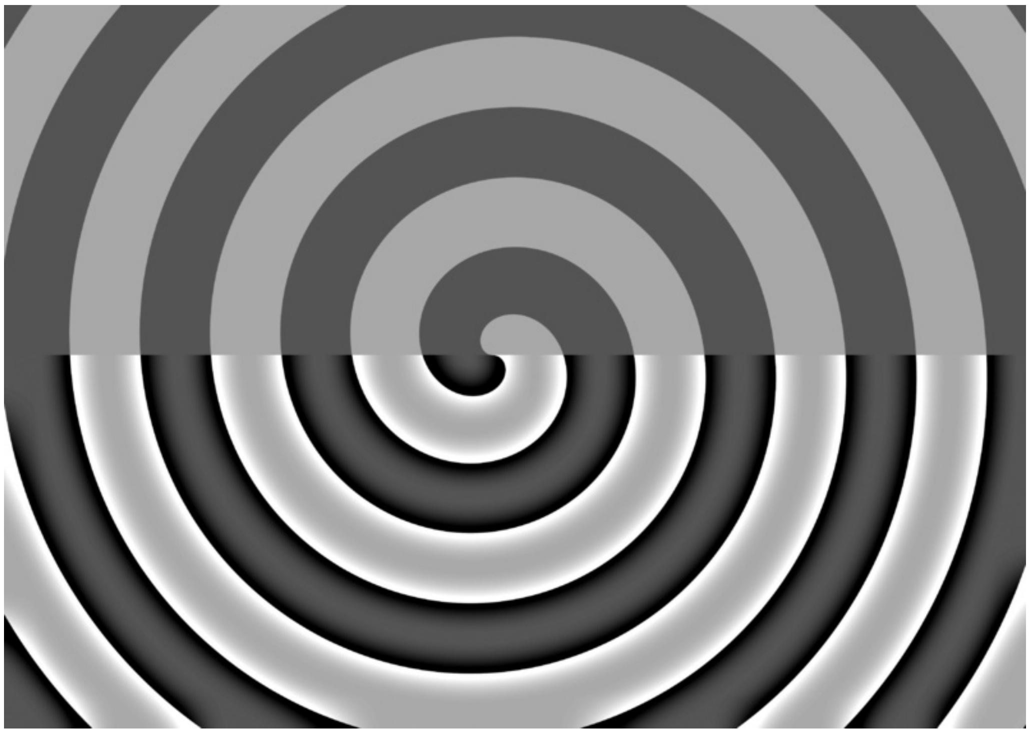


Laplacian of Gaussian

Sharpen filter

unfiltered

filtered

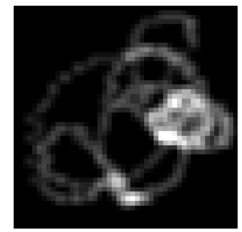


Convolution in the real world

Camera shake

=

*



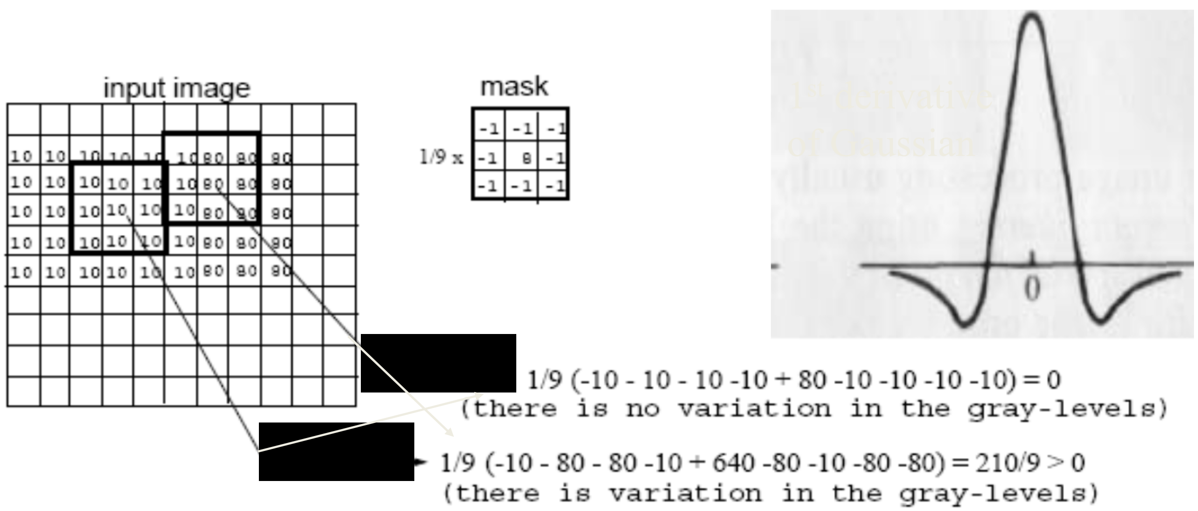
Source: Fergus, *et al.* "Removing Camera Shake from a Single Photograph", SIGGRAPH 2006

Bokeh: Blur in out-of-focus regions of an image.

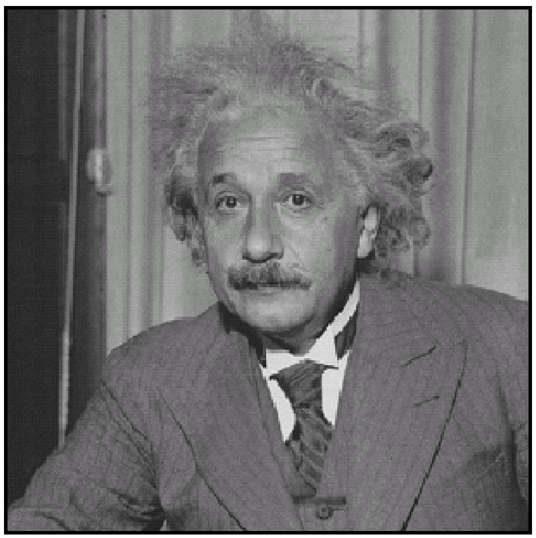
Source: http://lullaby.homepage.dk/diy-camera/bokeh.html

Image Sharpening

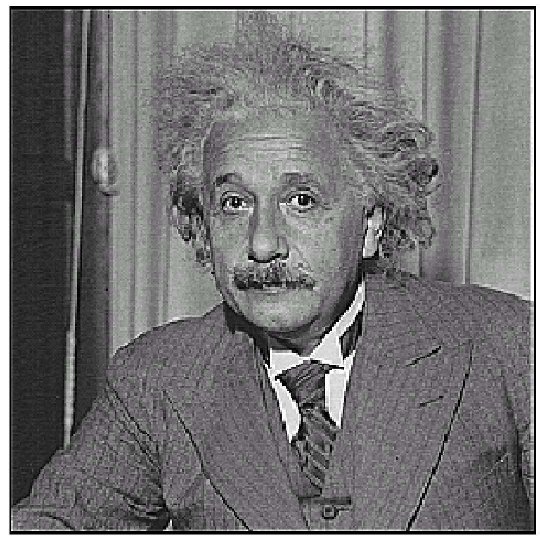
- Idea: compute intensity differences in local image regions.
- Useful for emphasizing transitions in intensity (e.g., in edge detection).



Sharpening



before



after

Source: D. Lowe

Filtering as matrix multiplication

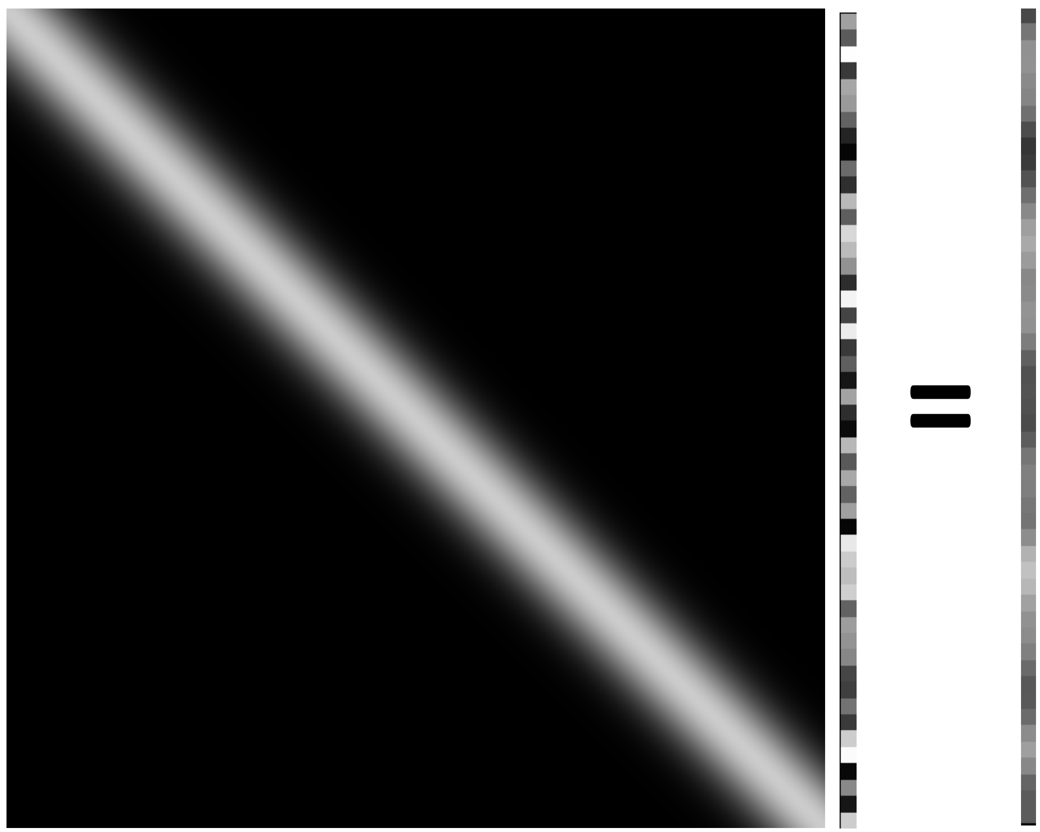
$$
\begin{bmatrix}
0.2 & 0.2 & 0.2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2 & 0.2 \\
0.2 & 0.2 & 0.2 & 0.2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2 \\
0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 \\
0.2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2 & 0.2 & 0.2 & 0.2 \\
0.2 & 0.2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.2 & 0.2 & 0.2
\end{bmatrix}
\begin{bmatrix}
0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1
\end{bmatrix}
$$

What kind of filter is this?

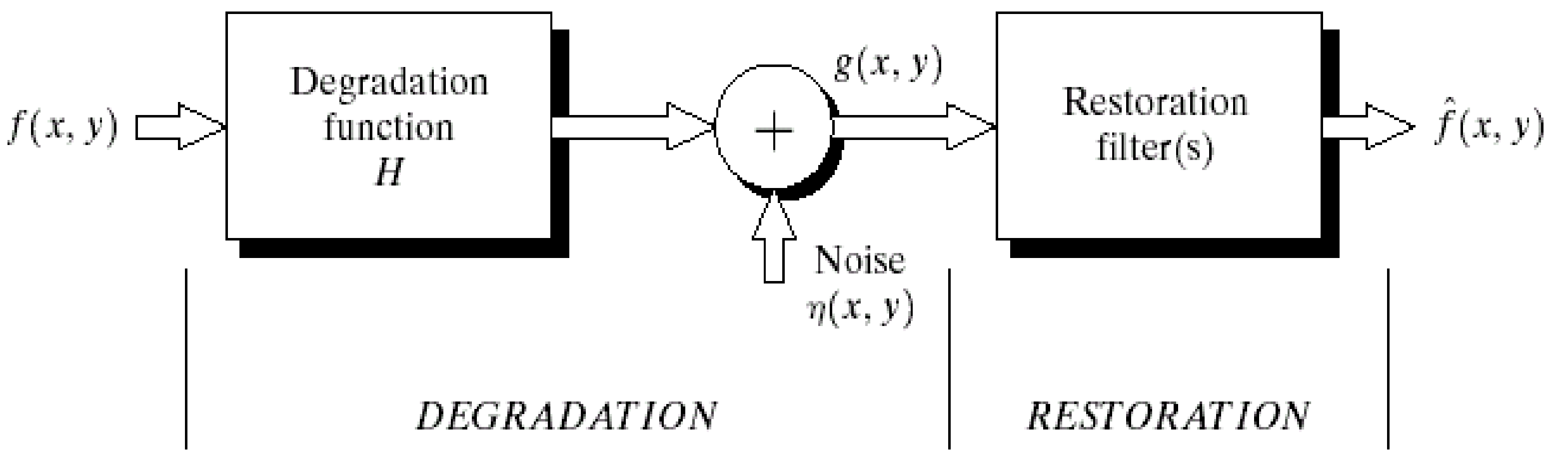
Multiplying row and column vectors

$$\begin{bmatrix} 0 & 0 & 0.2 & 0.2 & 0.2 & 0.2 & 0.2 & 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} = ?$$

Filtering as matrix multiplication



A model of the image degradation/restoration process



$$\begin{cases} g(x,y)=f(x,y)*h(x,y)+\eta(x,y) \\ G(u,v)=F(u,v)H(u,v)+N(u,v) \end{cases}$$

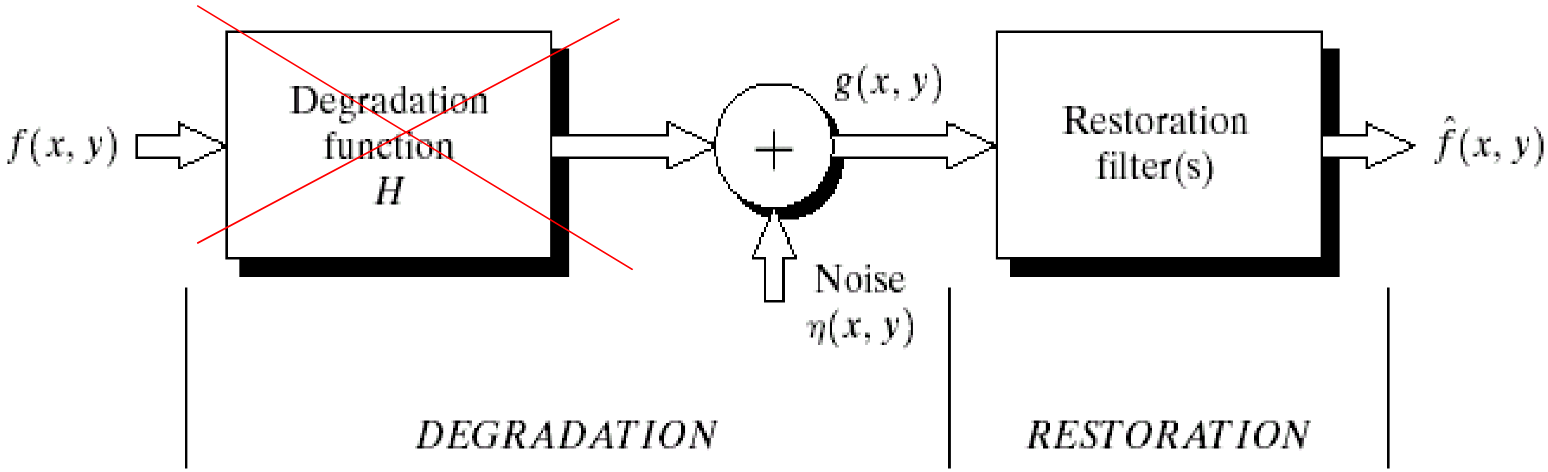
Measure the mean and variance

- Histogram is an estimate of PDF

$$\begin{cases} \mu = \sum_{z_i \in S} z_i p(z_i) \\ \sigma^2 = \sum_{z_i \in S} (z_i - \mu)^2 p(z_i) \end{cases}$$

$\Leftrightarrow$ $\begin{cases} \text{Gaussian: } \mu, \sigma \\ \text{Uniform: } a, b \end{cases}$

Additive noise only



$$\begin{cases} g(x,y)=f(x,y)+\eta(x,y) \\ G(u,v)=F(u,v)+N(u,v) \end{cases}$$

Estimation by image observation

- Take a window in the image
  - Simple structure
  - Strong signal content
- Estimate the original image in the window

known

$$H_s(u,v) = \frac{G_s(u,v)}{\hat{F}_s(u,v)}$$

estimate

Inverse filtering

- With the estimated degradation function H(u,v)

Unknown noise

G(u,v)=F(u,v)H(u,v)+N(u,v)

$$=> \hat{F}(u,v) = \frac{G(u,v)}{H(u,v)} = F(u,v) + \frac{N(u,v)}{H(u,v)}$$

Estimate of original image

Problem: 0 or small values

Sol: limit the frequency around the origin

Atmospheric Turbulence Blur

$$H(u,v) = e^{-k(u^2+v^2)^{\frac{5}{6}}}$$

Obtain restoration as:

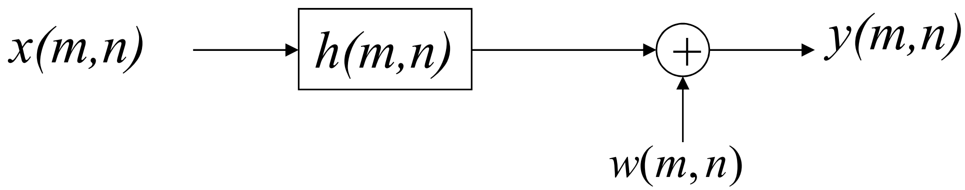
$$F(u,v) = H^{-1}(u,v)G(u,v)$$

Minimize:

$$\sum [g(x,y) - h(x,y) * f(x,y)]^2$$

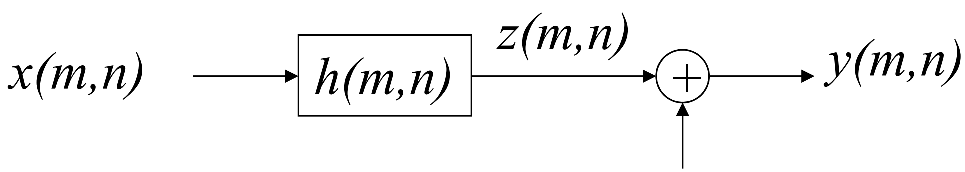
Modeling Blurring Process

- Linear degradation model



$h(m,n)$	blurring filter
$w(m,n) \sim N(0,\sigma_w^2)$	additive white Gaussian noise

The Curse of Noise



$$w(m,n) \sim N(0, \sigma_w^2)$$

Blurring SNR

$$BSNR = 10\log_{10}\frac{\sigma_z^2}{\sigma_w^2}$$

Blind vs. Nonblind Deblurring

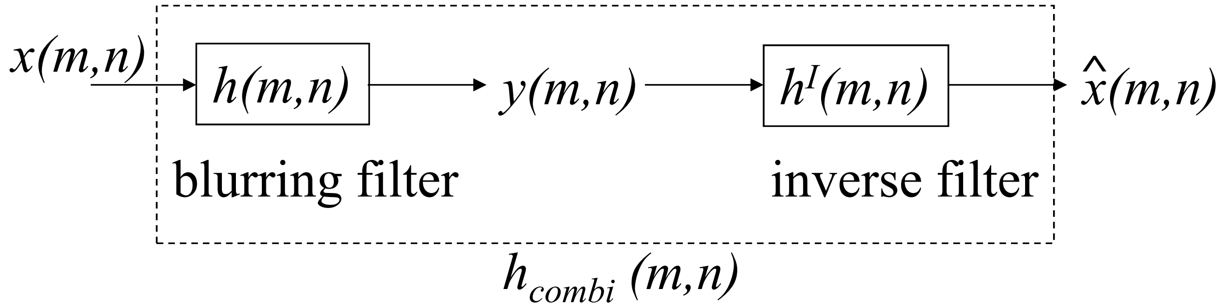
- Blind deblurring (deconvolution): blurring kernel h(m,n) is unknown
- Nonblind deconvolution:

  blurring kernel h(m,n) is known
- In this course, we only cover the nonblind case (the easier case)

Image Deblurring

- Introduction
- Inverse filtering
  - Suffer from noise amplification
- Wiener filtering
  - Tradeoff between image recovery and noise suppression
- Iterative deblurring*
  - Landweber algorithm

Inverse Filter



To compensate the blurring, we require

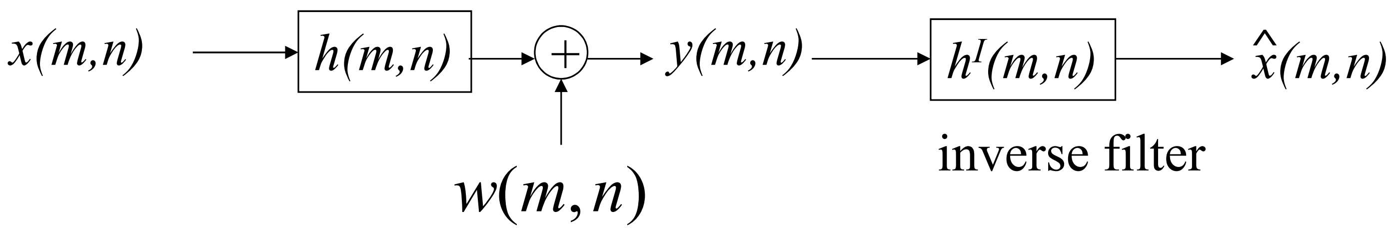
$$h_{combi}(m,n) = h(m,n) \otimes h^I(m,n) =$$

$$\sum_{k=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} h^I(m-k, n-l) h(k,l) = \delta(m,n), \forall (m,n)$$

$$\Downarrow$$

$$\boxed{H^I(w_1, w_2) = \frac{1}{H(w_1, w_2)}}$$

Inverse Filtering (Con't)



Spatial:

$$\hat{x}(m,n) = y(m,n) \otimes h^{I}(m,n) = (x(m,n) \otimes h(m,n) + w(m,n)) \otimes h^{I}(m,n)$$

Frequency:

$$\hat{X}(w_1,w_2) = Y(w_1,w_2)H^{I}(w_1,w_2) = \frac{X(w_1,w_2)H(w_1,w_2) + W(w_1,w_2)}{H(w_1,w_2)}$$

$$= X(w_1,w_2) + \frac{W(w_1,w_2)}{H(w_1,w_2)} \longrightarrow$$ amplified noise

Pseudo-inverse Filter

<u>Basic idea:</u>

To handle zeros in $H(w_1,w_2)$, we treat them separately when performing the inverse filtering

$$\boxed{H^{-}(w_1,w_2)=\begin{cases}\dfrac{1}{H(w_1,w_2)} & |H(w_1,w_2)|>\delta\\ 0 & |H(w_1,w_2)|\le\delta\end{cases}}$$

$$H_I=\frac{H^*}{|H|^2+\delta^2}$$

then

$$H_I\approx\frac{1}{H}\quad\text{if}\quad|\delta|<<|H|$$

and

$$H_I\approx 0\quad\text{if}\quad|\delta|>>|H|$$

Image Deblurring

- Introduction
- Inverse filtering
  - Suffer from noise amplification
- Wiener filtering
  - Tradeoff between image recovery and noise suppression
- Iterative deblurring*
  - Landweber algorithm

Norbert Wiener (1894-1964)

The renowned MIT professor Norbert Wiener was famed for his absent-mindedness. While crossing the MIT campus one day, he was stopped by a student with a mathematical problem. The perplexing question answered, Norbert followed with one of his own: "In which direction was I walking when you stopped me?" he asked, prompting an answer from the curious student. "Ah," Wiener declared, "then I've had my lunch”

Anecdote of Norbert Wiener

The Wiener filter can be understood better in the frequency domain. Suppose we want to design a frequency-domain filter G(k,l) so that the restored image is given by

$$\hat{X}(k,l) = G(k,l)Y(k,l)$$

We can choose G(k,l) so that we minimize

$$E[|X(k,l) - G(k,l)Y(k,l)|^2]$$

y(m,n) = h(m,n)*x(m,n) + u(m,n)

where * is 2-D convolution, h(m,n) is the point-spread function (PSF), f(m,n) is the original image, and u(m,n) is noise.

The minimizer of this expression is

$$G(k,l) = \frac{H(k,l)}{|H(k,l)|^2 + S_u(k,l)/S_x(k,l)}$$

where $S_x(k,l) =$ signal power spectrum

and $S_u(k,l) =$ noise power spectrum

$\sigma_u^2 =$ the variance at each pixel

then the noise power spectrum is given by

$$S_u(k,l) = MN\sigma_u^2$$

This filter gives the minimum mean-square error estimate of X(k,l). Now

Wiener Filtering

Also called Minimum Mean Square Error (MMSE) or Least-Square (LS) filtering

$$\boxed{H_{mmse}(w_1, w_2) = \frac{H^*(w_1, w_2)}{|H(w_1, w_2)|^2 + K}}$$

constant

<u>Example choice of K:</u> $K = \dfrac{\sigma_w^2}{\sigma_z^2}$

noise energy

signal energy

K=0 $\rightarrow$ inverse filtering

Constrained Least Square Filtering

Similar to Wiener but a different way of balancing the tradeoff between

$$\boxed{H_{mmse}(w_1, w_2) = \frac{H^*(w_1, w_2)}{|H(w_1, w_2)|^2 + \gamma |C(w_1, w_2)|^2}}$$

<u>Example choice of C:</u>

$$C(m,n) = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

Laplacian operator

γ=0 $\rightarrow$ inverse filtering

Image Deblurring

- Introduction
- Inverse filtering
  - Suffer from noise amplification
- Wiener filtering
  - Tradeoff between image recovery and noise suppression
- Iterative deblurring*
  - Landweber algorithm

Method of Successive Substitution

- A powerful technique for finding the roots of any function f(x)
- Basic idea
  - Rewrite $f(x)=0$ into an equivalent equation $x=g(x)$ (x is called fixed point of g(x))
  - Successive substitution: $x_{i+1}=g(x_i)$
  - Under certain condition, the iteration will converge to the desired solution

Numerical Example

$$f(x) = x^2 - 3x + 2$$

$$\downarrow$$

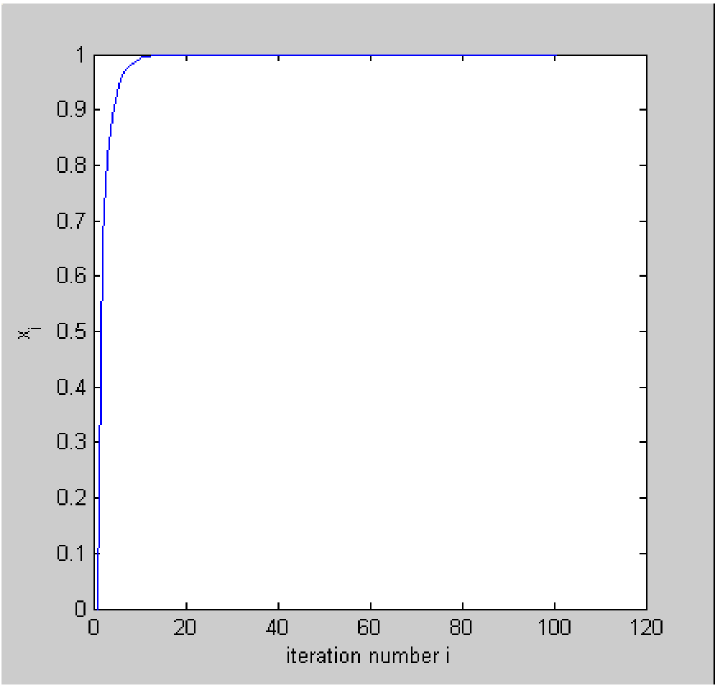
Two roots: $x_1 = 1, x_2 = 2$

$$f(x) = x^2 - 3x + 2 = 0 \Rightarrow x = \frac{x^2 + 2}{3} = g(x)$$

$$\Downarrow$$

successive substitution: $\boxed{x_{i+1} = \frac{{x_i}^2 + 2}{3}}$

Numerical Example (Con't)



Note that iteration quickly converges to x=1

Landweber Iteration

Linear blurring $Y(w_1, w_2) = H(w_1, w_2) X(w_1, w_2)$

$\downarrow$

We want to find the root of $f(X) = Y - HX$

$\downarrow$

$$f(X) = 0 \Rightarrow X = X + \beta f(X) = X + \beta(Y - HX) = g(X)$$

β relaxation parameter – controls convergence property

Successive substitution:

$$\boxed{\begin{aligned} &X_0 = 0 \\ &X_{n+1} = X_n + \beta(Y - HX_n) \end{aligned}}$$

TOTAL VARIATION DEBLURRING

Tikhonov regularization: The main objective of regularization is to incorporate more information about the desired solution in order to stabilize the problem and find a useful and stable solution. The most common and well-known form of regularization is that of Tikhonov (Stand and Rudnicki, 2007). The Tikhonov regularized minimum norm solution of Eq. 1 is the vector $F_\delta \in U^N$ that minimizes the expression:

$$\|Hf - g\|_2^2 + \lambda^2 \|Lf\|_2^2 \qquad (5)$$

where, $\lambda>0$ is called a regularization parameter.

We denote:

$$F_\delta = \arg\min_x \left\{ \|Hf - g\|_2^2 + \lambda^2 \|Lf\|_2^2 \right\} \qquad (6)$$

Regularization can be understood as a balance between two requirements:

- f should give a small residual Hf-g
- f should be small in L^2 norm

The regularization parameter $\lambda>0$ can be used to tune the balance. Note that in inverse problems there are typically infinitely many f satisfying Eq. 6.

APPLICATION TO IMAGE RESTORATION

This study will establish a new approach that can be used to solve a constrained optimization ill-posed problem in order to improve a blurred or noisy image. We will add many types of degradation functions to matrices of different sizes and then try to restore the original. Our starting image is a gray-level image contained in the mxn matrix. Each element in the matrix represents a pixel's gray intensity between black and white (0 and 255).

Assume, we know how fast the blurring function operator is known. The simplest approach is to solve the least squares problem:

$$\min(\|H \times X - G\|^2) \tag{18}$$

In practice the results obtained with this simple approach tend to be noisy, because this term expresses only the fidelity to the available data g. To compensate for this, a regularization term below is added to improve smoothness of the estimate:

$$0.004 \times \|L \times X\|^2 \tag{19}$$

where, L is the discrete Laplacian, which relates each pixel to those surrounding it. L = del2(X) is a discrete approximation of:

$$l = \frac{\nabla^2 X}{2N} = \frac{1}{2N}\left(\frac{d^2X}{dx^2} + \frac{d^2X}{dy^2}\right) \tag{20}$$

where, X is the estimated matrix. The matrix L has the same size as X with each element equal to the difference between an element of X and the average of its four neighbors.

Since, we know we are looking for a gray intensity, we also impose the constraint that the elements of X must fall between 0 and 255.

To obtain the deblurred image, we want to solve for X:

$$\min (\| H \times X - G \|^2 + 0.004 \times \| L \times X \|^2) \tag{21}$$

We can implement our objective function using this expression; the number of variables in this objective function to be minimized will be mxn which is the size of the original matrix representing the original image.