Topic 2: Image Processing - Basic Operators and Filtering

Jessica Zhang
Department of Mechanical Engineering
Courtesy Appointment in Biomedical Engineering
Carnegie Mellon University
jessicaz@andrew.cmu.edu
http://www.andrew.cmu.edu/user/jessicaz

Mechanical Engineering
Carnegie Mellon

Image Processing

- Image processing is the manipulation and analysis of image-based digital data to enhance and illuminate information within the data stream.

Medical Image Processing

- The medical mission differs from the other forms of image processing
  - **Satellite surveillance analysis:** the purpose is largely a screening and cartographic task, aligning multiple types of data and corresponding them to a known map and highlighting possible points of interest.
  - **Computer vision:** camera views must be analyzed accounting for perspective geometry and photogrammetric distortions associated with the optical systems, which are the basis for robotic sensors.
  - In many systems, the primary tasks are autonomous navigation, target identification and acquisition, and threat avoidance.

Medical Imaging Processing

- The input data stream are in 3D.
- Medical image processing needs both human and computers. Computers are assistants.
- Medical tasks:
  - Data operations of filtering, noise removal, and contrast and feature enhancement.
  - Detection of medical conditions or events.
  - Quantitative analysis of the lesion or detected events, such as tumor volume or the length of a bone fracture.
- The main goal in medicine is to provide the clinician with powerful tools for image analysis and measurement.

Medical Image Processing

- Three basic categories:
  - Filtering: filtering and preprocessing the data before detection and analysis are performed.
  - Segmentation: partitioning an image into contiguous regions with cohesive properties.
  - Registration: aligning multiple data streams or images, permitting the fusion of different information creating a more powerful diagnostic tool than any single image alone.

Images

- A volume image ϕ is a mapping from R^3 to R^n where n=1 for a scalar field.

$$\phi : \vec{x} \to R, \qquad \vec{x} \subset R^3$$

where x is the domain of the volume. The image is often written as a function $\phi(x, y, z)$. Suppose F is a discrete sampling of a 2D image $\phi(x, y)$, then

$$F_{i,j} = \phi(x_i, y_j)$$

The partial derivative: $\phi_x(x_i, y_j) = \left.\frac{\partial \phi}{\partial x}\right|_{x_i, y_j} \approx \delta_x F_{i,j}$

Central Difference: $\delta_x F_{i,j} \equiv \frac{F_{i+1,j} - F_{i-1,j}}{x_{i+1} - x_{i-1}} = \frac{F_{i+1,j} - F_{i-1,j}}{2h}$

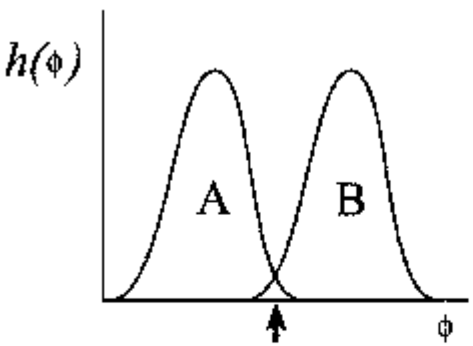
where h – grid spacing

Mechanical Engineering
Carnegie Mellon

Thresholding

- Point operators ignore information about pixel location, relying only on pixel intensity.
  - Thresholding: a binary decision made on each pixel independent of its neighbors.
  - How to choose the optimum threshold value – histogram of pixel intensity

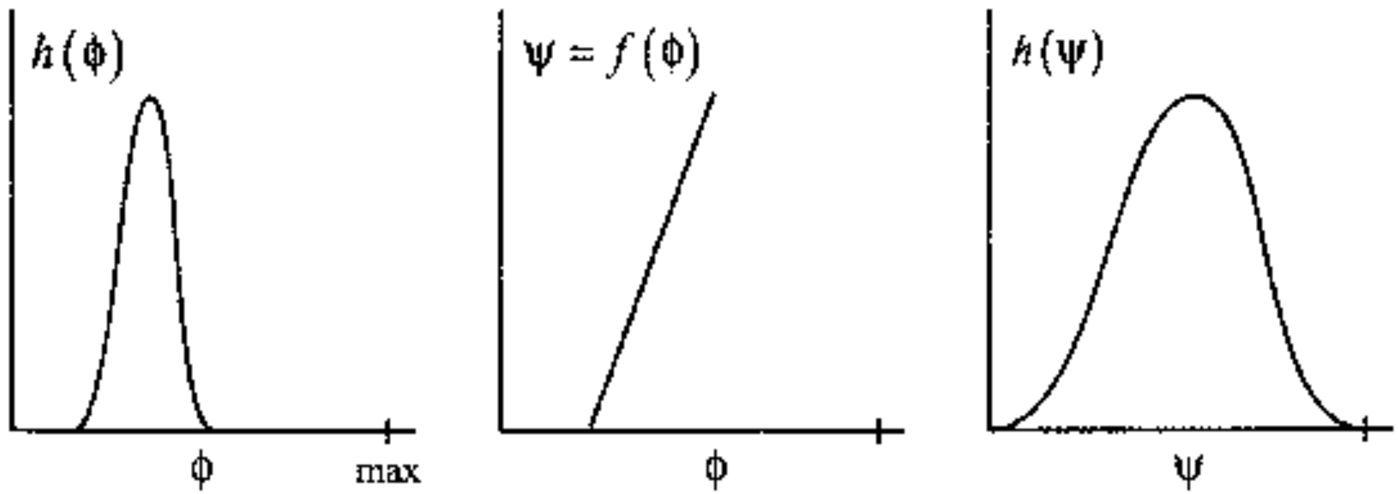
A fractured tibia bone is segmented from other tissues by thresholding



Histogram of pixel intensity for tissue type A and B.

Contrast Enhancement

- To adjust the brightness and enhance the contrast by designing a stretching function or a windowing function.



Histogram of pixel intensity for original image (left) and new image (right).
The middle picture shows the stretching function.

Contrast Enhancement

Original Image

After Contrast Enhancement

Contrast Enhancement

• A fast approach based on localized contrast manipulation. It is adaptive, multi-scale, weighted localization, anisotropic.

• Adaptive contrast enhancement – a new intensity is assigned to each pixel according to an adaptive transfer function based on the local statistics (local min, max, avg).

*A propagation scheme is used to remove block-like artifacts.

$$\mathrm{lavg}_{m,n} = (1 - C) \times \mathrm{lavg}_{m,n} + C \times \mathrm{lavg}_{m-1,n}$$

C – conductivity factor, [0, 1].

*A conditional propagation scheme:

$$\begin{cases} if(\mathrm{lmin}_{m-1,n} < \mathrm{lmin}_{m,n}) \\ \quad \mathrm{lmin}_{m,n} = (1 - C) \times \mathrm{lmin}_{m,n} + C \times \mathrm{lmin}_{m-1,n} \\ \\ if(\mathrm{lmax}_{m-1,n} > \mathrm{lmax}_{m,n}) \\ \quad \mathrm{lmax}_{m,n} = (1 - C) \times \mathrm{lmax}_{m,n} + C \times \mathrm{lmax}_{m-1,n} \end{cases}$$

Reference: Z. Yu, C. Bajaj. A Fast and Adaptive Algorithm for Image Contrast Enhancement. *ICIP'04, Vol. 2, 1001-1004.*

Contrast Enhancement

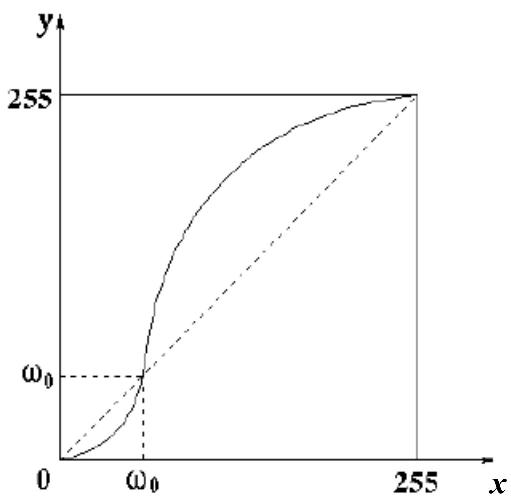
- Transfer function

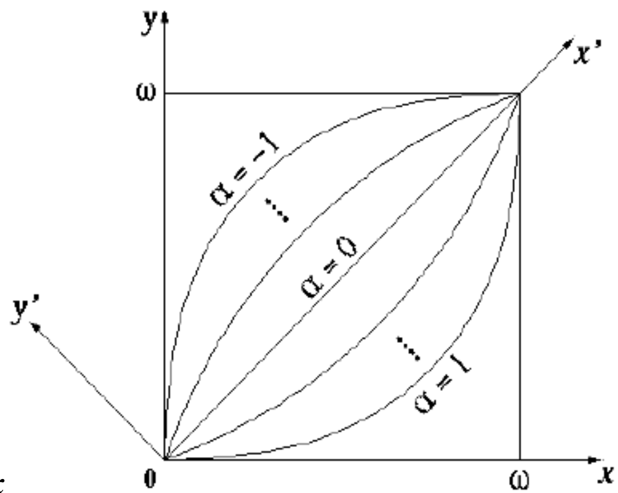
$$y = \begin{cases} w_0 - \sqrt{w_0^2 - x^2}, & \text{if } x \leq w_0 \\ w_0 + \sqrt{(255 - w_0)^2 - (255 - x)^2}, & \text{else} \end{cases}$$

x – input range

y – output range

if $|lmax - lmin| < w_0$ (a fixed value), the contrast is thought of as noise and reduced.





$$I_{new} = \omega \times \frac{I_{old} - lmin}{lmax - lmin}$$

$$A_{new} = \omega \times \frac{A_{old} - lmin}{lmax - lmin}$$

$$\alpha = \frac{(A_{new} - I_{new})}{128}$$

A – average intensity

$$I_{enh} = lmin + f(I_{new})$$

f is defined as $f = \dfrac{-b - \sqrt{b^2 - 4ac}}{2a}$

$$\begin{cases} a = \frac{\alpha}{2\omega} \\ b = \frac{\alpha}{\omega} x - \alpha - 1 \\ c = \frac{\alpha}{2\omega} x^2 - \alpha x + x \end{cases}$$

Mechanical Engineering
Carnegie Mellon

Contrast Enhancement

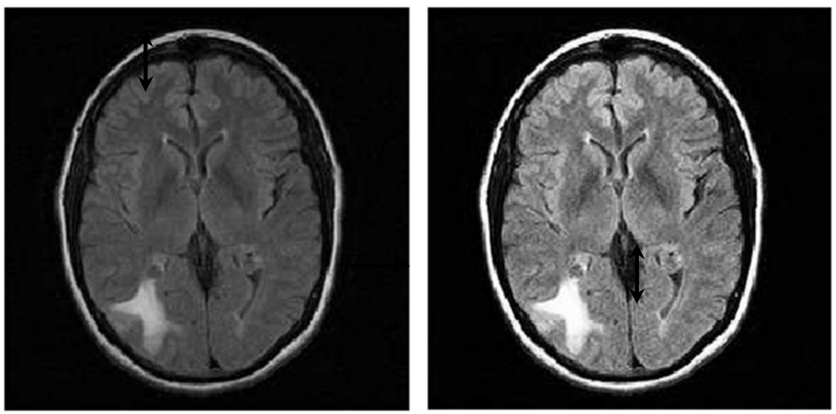
Anisotropic propagation:

$$\begin{cases} if(\mathrm{Imin}_{m-1,n} < \mathrm{Imin}_{m,n}) \\ \quad \mathrm{Imin}_{m,n} \mathrel{+}= (\mathrm{Imin}_{m-1,n} - \mathrm{Imin}_{m,n}) \\ \qquad *exp(-R * |\mathrm{Imin}_{m-1,n} - \mathrm{Imin}_{m,n}|) \\ \\ if(\mathrm{Imax}_{m-1,n} > \mathrm{Imax}_{m,n}) \\ \quad \mathrm{Imax}_{m,n} \mathrel{+}= (\mathrm{Imax}_{m-1,n} - \mathrm{Imax}_{m,n}) \\ \qquad *exp(-R * |\mathrm{Imax}_{m-1,n} - \mathrm{Imax}_{m,n}|) \\ \\ \mathrm{Iavg}_{m,n} \mathrel{+}= (\mathrm{Iavg}_{m-1,n} - \mathrm{Iavg}_{m,n}) \\ \qquad *exp(-R * |\mathrm{Iavg}_{m-1,n} - \mathrm{Iavg}_{m,n}|) \end{cases}$$

R is called *resistance factor* $[0.01, 0.1]$

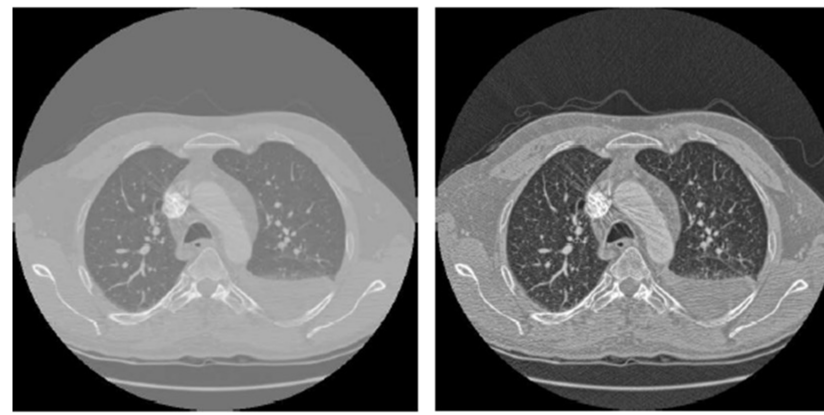
Contrast Enhancement

- Localized contrast manipulation.
- Design a stretching function to locally change the intensity to reflect better contrast.



Original | After contrast enhancement

Brain MR image



Original | After contrast enhancement

Chest CT image

Z. Yu, C. Bajaj
A Fast and Adaptive Algorithm for Image Contrast Enhancement
Proceedings of 2004 IEEE International Conference on Image Processing (ICIP'04), Volume 2, Oct. 24-27, 2004, Pages 1001-1004,

Histogram Equalization

- Histogram equalization is to optimize the match between the image and the display system, which ensures that each level of pixel intensity is equally represented in the image.

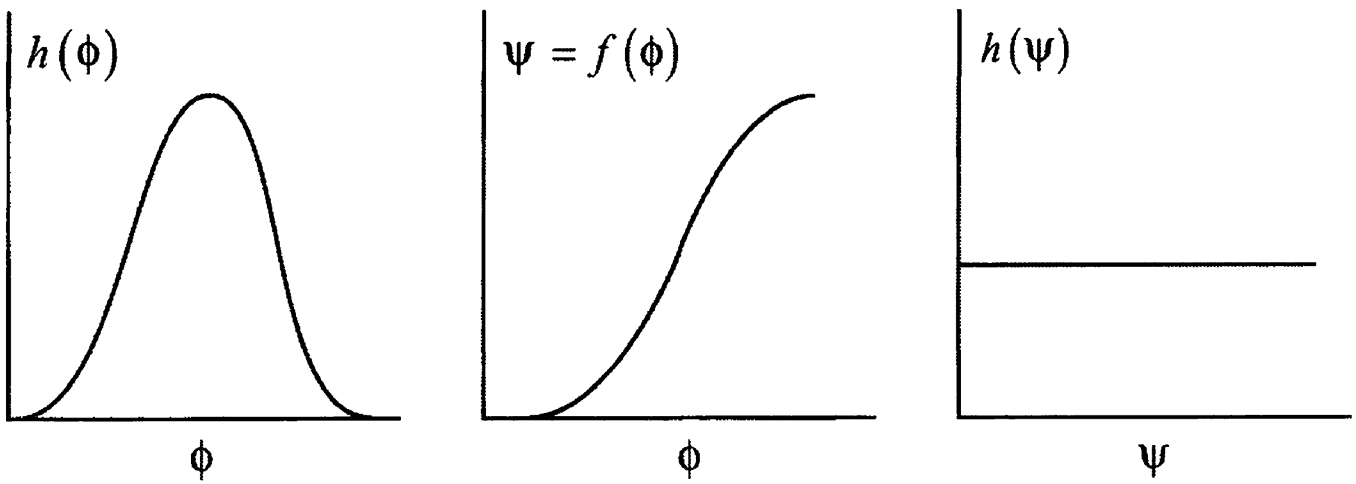
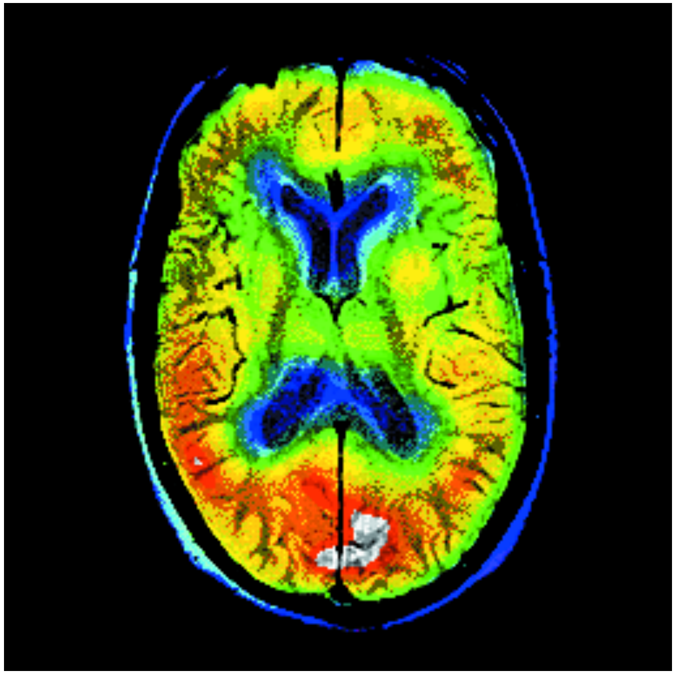


Figure 2.4. Histogram equalization. Left: Histogram $h(\phi)$ of pixel intensity ϕ of original image; Center: equalizing function $\psi = f(\phi)$ to yield a new intensity ψ; Right: new histogram $h(\psi)$ in which each intensity is equally represented.

Color Maps

- Color maps (red, green and blue) are often used to overlay additional information in anatomical images. (rainbow color map)



Overlay physiological activity (the biological activity) from SPECT in anatomical image of the brain (MRI).

Linear Filtering

- Linear filtering considers multiple pixels in a linear and space-invariant manner.
- Two standard techniques:
  – Convolution
  – Fourier Transform

Convolution

- Convolution is used to make measurements, detect features, smooth noisy signals, and deconvolve the effects from image acquisition (e.g., deblurring the known optical artifacts from a telescope).

- Filter kernel $h(x)$

$$\phi(x) \otimes h(x) = \int_{-\infty}^{\infty} \phi(x-\tau) h(\tau) d\tau$$

Convolution

- The expression $\phi(x) \otimes h(x)$ itself describes an image mapping, therefore:

$$\phi(x) \otimes h(x) = (\phi \otimes h)(x)$$

- 2D/3D convolution

$$\phi(x, y) \otimes h(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi(x-\tau, y-v) h(\tau, v) dv d\tau$$

$$\phi(x, y, z) \otimes h(x, y, z) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi(x-\tau, y-v, z-\omega) h(\tau, v, \omega) d\omega dv d\tau$$

Convolution - Denoising

- Consider the Gaussian function as a smoothing filter kernel
- High-frequency noise is removed, but edges are also blurred.

$$g(x,\sigma)=\frac{1}{\sqrt{2\pi\sigma}}e^{-\frac{x^2}{2\sigma^2}}$$

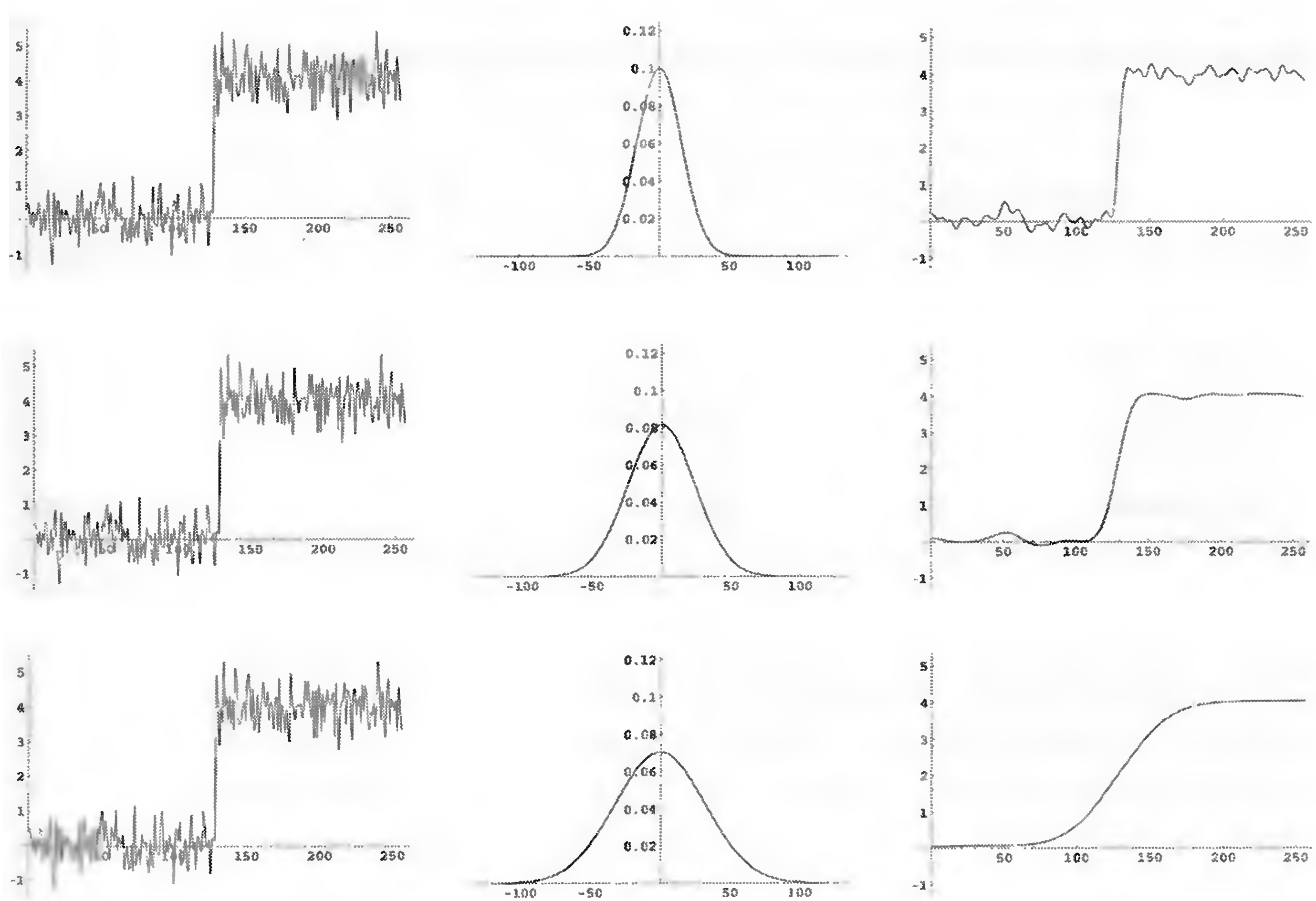
with different width σ

Input image | Gaussian kernel | Resulting image

$\sigma = 16$

$\sigma = 24$

$\sigma = 32$



Mechanical Engineering
Carnegie Mellon

Properties of Convolution Operation

- Convolution is linear, commutative, associative, and distributive over addition.

$$(\alpha p + \beta q) \otimes r = \alpha (p \otimes r) + \beta (q \otimes r)$$

$$p \otimes q = q \otimes p$$

$$(p \otimes q) \otimes r = p \otimes (q \otimes r)$$

$$p \otimes (q + r) = p \otimes q + p \otimes r$$

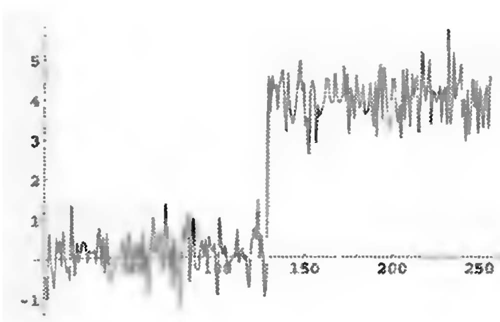
Differentiation by Convolution

- In a continuous domain, differentiation is denoted as convolution

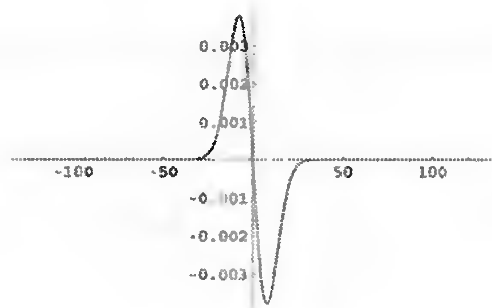
$$\frac{\partial}{\partial x}\phi = \frac{\partial}{\partial x} \otimes \phi$$

- Differentiation enhances high-frequency noise, so it is necessary to smooth the image before computing its derivative

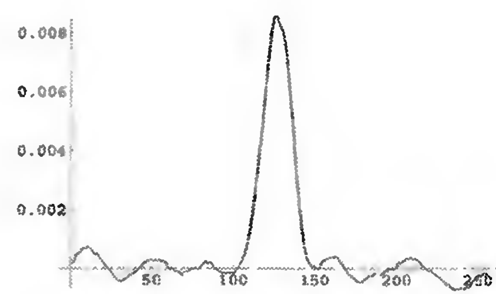
$$h(x) \otimes \frac{\partial}{\partial x}\phi(x) = \frac{\partial}{\partial x}h(x) \otimes \phi(x)$$



Input image



The derivative of a Gaussian kernel



The derivative of the smoothed input image

Mechanical Engineering
Carnegie Mellon

Convolution of Discrete Data

- Compared to the continuous form, convolution of discrete data substitutes integration with discrete summation.
- A discrete convolution of image P with kernel Q

$$P_{x,y} \otimes Q_{x,y} = \sum_{i}^{domain_x[Q]} \sum_{j}^{domain_y[Q]} P_{x-i,y-j} Q_{i,j}$$

- Convolution of discrete data is linear, commutative, associative, and distributive over addition.
- Can be used to smooth data, detect boundaries.

Central difference operation

$$\delta_x F_{i,j} \equiv \frac{F_{i+1,j} - F_{i-1,j}}{x_{i+1} - x_{i-1}} = \frac{F_{i+1,j} - F_{i-1,j}}{2h}$$

Kernel

$$\begin{bmatrix} 0 & 0 & 0 \\ -0.5 & 0 & 0.5 \\ 0 & 0 & 0 \end{bmatrix}$$

Difference of Offset Gaussians

- Boundaries in an image represent areas of high gradient magnitude, i.e., the image intensity increases or decreases rapidly across the edge.
- The gradient vector

$$\left[\frac{\partial\phi}{\partial x} \quad \frac{\partial\phi}{\partial y}\right]$$

  is oriented in the direction of the steepest change in image intensity, normal to the implied boundary.
- The gradient magnitude represents an orientation-independent measure of boundary strength.

$$\left(\frac{\partial\phi}{\partial x}\right)^2+\left(\frac{\partial\phi}{\partial y}\right)^2$$

Higher-Order Differentiation

- The 2nd derivative is a matrix, known as the Jacobian

$$\begin{bmatrix} \frac{\partial^2 \phi}{\partial x^2} & \frac{\partial^2 \phi}{\partial x \partial y} \\ \frac{\partial^2 \phi}{\partial y \partial x} & \frac{\partial^2 \phi}{\partial y^2} \end{bmatrix}$$

- The Laplacian $\frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2}$: A boundary exists where there is no net change in gradient (i.e., the gradient is at a maximum on the boundary), making the Laplacian zero.

- $\frac{\partial^2 \phi}{\partial y \partial x}$ indicates a decrease in the y-component of the gradient as one moves in the positive x-direction.

- $\frac{\partial^2 \phi}{\partial x \partial y}$ indicates a decrease in the x-component of the gradient as one moves in the positive y-direction.

Difference of Gaussians (DOG)

- As with the Laplacian, the DOG kernel can detect edges independent of orientation. The process is also called "unsharp masking", and the result is an edge-enhanced image.
- Given a 2D input image $\phi(x, y)$ and two Gaussian filter kernels of differing aperture, $g(x, y, \sigma_1)$ and $g(x, y, \sigma_2)$ where $\sigma_1 < \sigma_2$, and edge enhanced image $\phi'(x, y)$ can be formed by a linear combination of the two filters.

$$\phi'(x, y) = \phi(x, y) \otimes (g(x, y, \sigma_1) - g(x, y, \sigma_2))$$

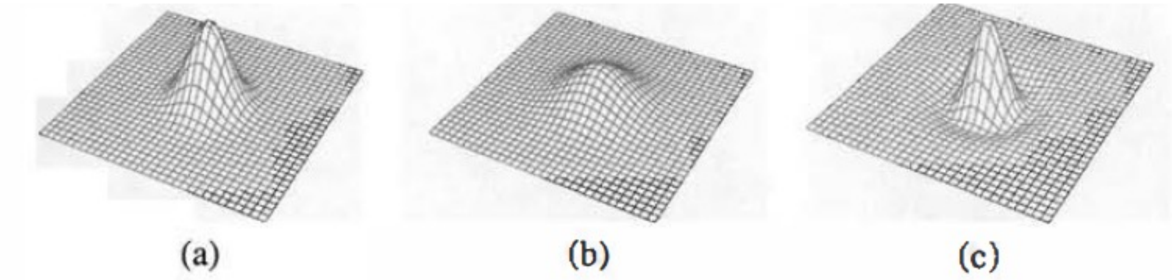
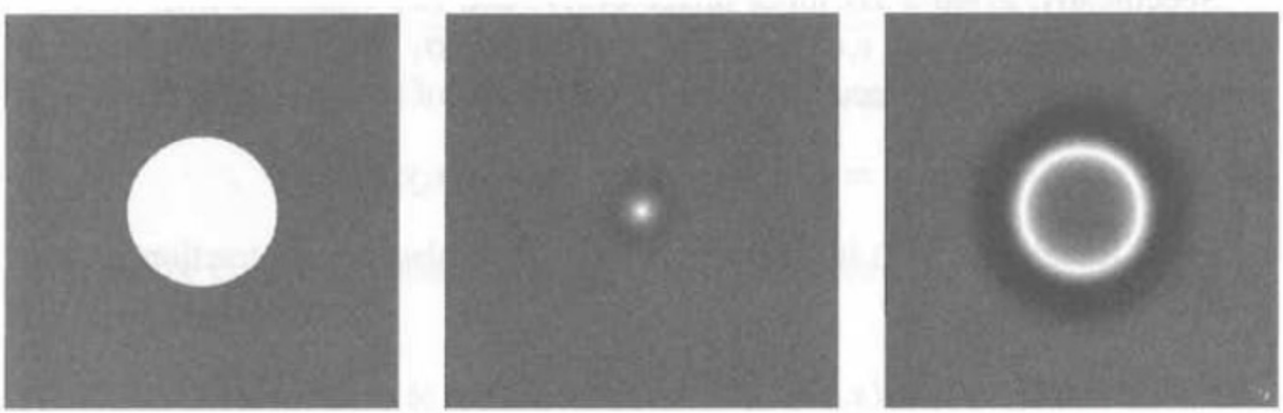


Figure 2.13. A difference of Gaussians filter kernel (depicted in 3D as a height field): (a) 2D Gaussian kernel $g(x,y,\sigma_1 = 16)$; (b) 2D Gaussian kernel $g(x,y,\sigma_2 = 32)$; (c) $g(x,y,\sigma_1) - g(x,y,\sigma_2)$.

Difference of Gaussians (DOG)

(a) (b) (c)

Figure 2.14. 2D image filtered by a difference of Gaussians (depicted in 3D as a height field): (a) 2D input $\phi(x,y)$; (b) $g(x,y,\sigma_1 = 16) - g(x,y,\sigma_2 = 32)$; (c) $\phi(x,y) \otimes (g(x,y,\sigma_1) - g(x,y,\sigma_2))$.



(a) (b) (c)

Figure 2.15. 2D image filtered by a difference of Gaussians (depicted as a 2D grayscale density field). (a) 2D input $\phi(x,y)$; (b) $g(x,y,\sigma_1 = 16) - g(x,y,\sigma_2 = 32)$; (c) $\phi(x,y) \otimes (g(x,y,\sigma_1) - g(x,y,\sigma_2))$.

Difference of Gaussians (DOG)

- Modify the behavior from edge detection to contrast enhancement

$$2g(x, y, \sigma_1 = 16) - g(x, y, \sigma_2 = 32)$$

- This DOG kernel returns unchanged the non-zero pixel values inside the circle, while exaggerating the discontinuity in a band around the boundary.

(a) (b) (c)

Figure 2.16. A difference of Gausssians filter used for contrast enhancement (depicted as a 2D greyscale density field). This method is also known as *unsharp masking*: (a) 2D input $\phi(x,y)$; (b) $2g(x,y,\sigma_1 = 16) - g(x,y,\sigma_2 = 32)$; (c) $\phi(x,y) \otimes (2g(x,y,\sigma_1) - g(x,y,\sigma_2))$.

Fourier Transform

- The Fourier transform decomposes an image into component sinusoidal spatial functions, and it maps the spatial/image domain to the frequency domain.
- Frequency: periodic variations of intensity in space

Fourier Transform:

$$F(\phi(x)) = \int_{-\infty}^{\infty} \phi(x) e^{-i2\pi\omega x} dx = \Phi(\omega)$$

Inverse Fourier Transform:

$$F^{-1}(\Phi(\omega)) = \int_{-\infty}^{\infty} \Phi(\omega) e^{i2\pi\omega x} d\omega = \phi(x)$$

Key Properties of Fourier Transforms

- Scaling in image space ↔ Inverse scaling in frequency space

$$F(\phi(ax)) = \frac{1}{a}\Phi(\frac{\omega}{a})$$

- Convolution in image space ↔ Multiplication in frequency space

Given a kernel $h(x)$, and its Fourier transform $H(\omega)$:

Convolution theory: $F(\phi(x) \otimes h(x)) = \Phi(\omega) H(\omega)$

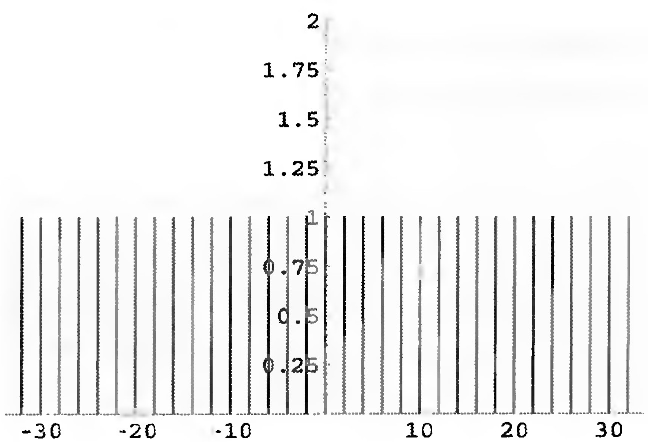
$$\phi(x) \otimes h(x) = F^{-1}(F(\phi(x))F(h(x)))$$

Four Important Transform Pairs

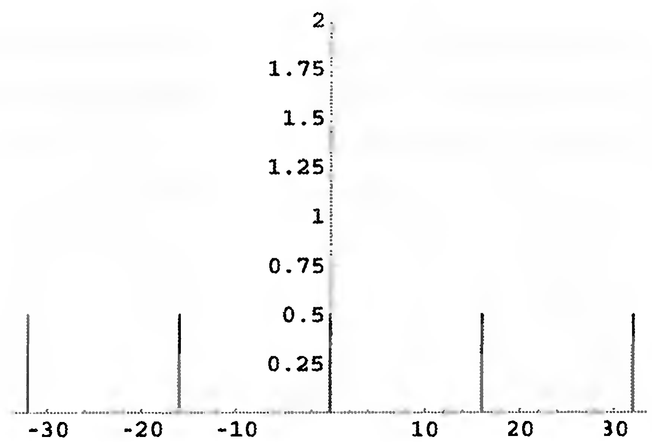
1. Comb function – used to sample data discretely in the image domain
2. Box filter – a square pulse, used to resample the discrete image into another lattice.
3. Pyramid filter – a triangular shaped filter used for linear interpolation in reconstruction
4. Gaussian filter – a good all purpose filter, except for its infinite extent

Comb Function

- A comb function is a series of pulse, or delta-function.
- A pulse is an infinitely narrow, infinitely high spike with finite area.
- Multiplying $f(x)$ by the comb function captures evenly-spaced samples to produce a digital image.



Comb filter



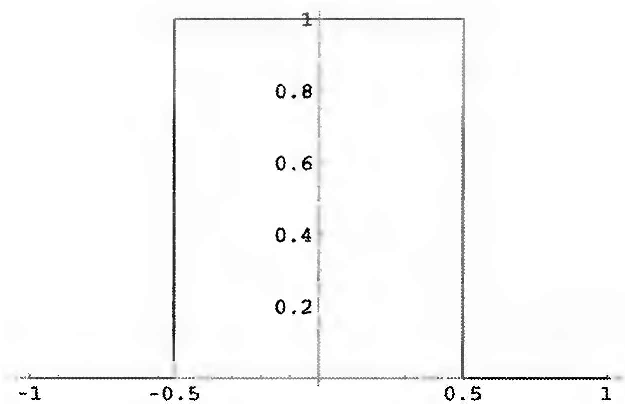
Fourier transform of
comb filter

Aliasing – Nyquist Criterion

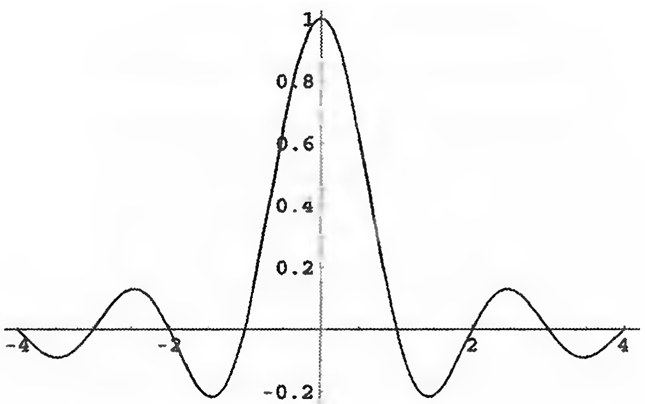
- Multiplying $f(x)$ with the comb function in image space leads to convolving their Fourier transform.
- The Fourier transform $F(f(x))$ is repeated by the comb function in frequency space at each impulse.
- If $F(f(x))$ is broader than the impulse spacing in the frequency space, aliasing will occur.
- Nyquist frequency: the maximum allowable frequency to avoid aliasing, half of the sampling frequency.

Box Filter

- A reconstruction function in nearest-neighbor interpolation – to re-sample a discrete image to a different lattice.



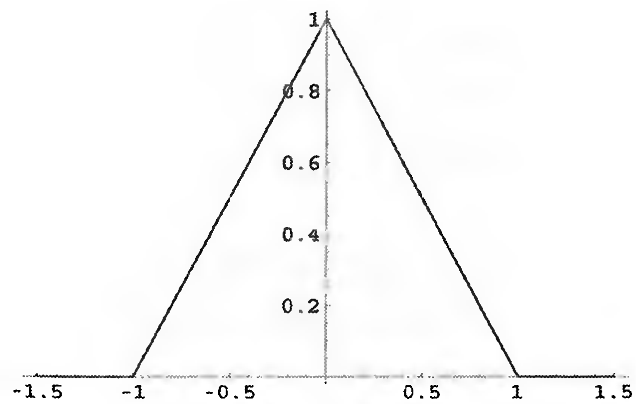
Box function



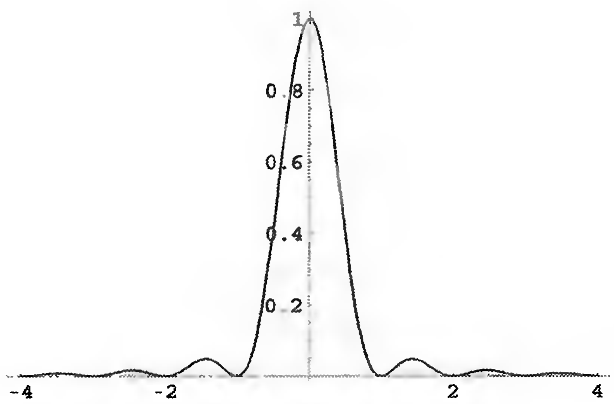
Fourier transform of box function

Pyramid Filter

- Used for linear interpolation
- More accurate than the box filter, but more expensive



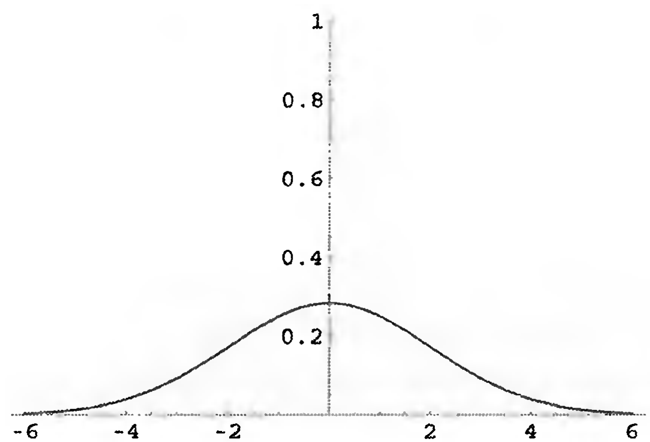
Pyramid function



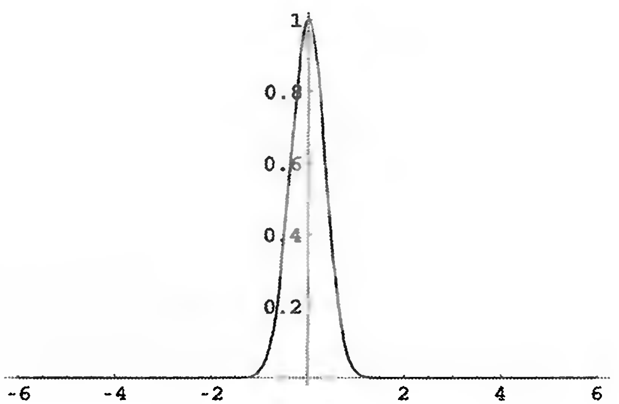
Fourier transform of pyramid function

Gaussian Filter

- The Fourier transform of a Gaussian is itself a Gaussian
- No side lobes
- Used to blur images



Gaussian function



Fourier transform of
Gaussian function

2D/3D Fourier Transform

- Fourier transform can be extended to 2D and 3D

Fourier Transform:

$$F(\phi(x,y)) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \phi(x,y) e^{-i2\pi(\omega x + vy)} dx dy = \Phi(\omega, v)$$

Inverse Fourier Transform:

$$F^{-1}(\Phi(\omega,v)) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \Phi(\omega,v) e^{i2\pi(\omega x + vy)} d\omega dv = \phi(x,y)$$

Mechanical Engineering
Carnegie Mellon

Separability

- The multi-dimensional Fourier transform is separable in each of the orthonormal basis dimensions.

$$F(\phi(x,y)) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\phi(x,y)e^{-i2\pi(\omega x+vy)}dxdy = \Phi(\omega,v)$$

can be rewritten as

$$F(\phi(x,y)) = \int_{-\infty}^{\infty}\left[\int_{-\infty}^{\infty}\phi(x,y)e^{-i2\pi\omega x}dx\right]e^{-i2\pi vy}dy = \Phi(\omega,v)$$

If the function $\phi(x,y)$ can be separated into component functions

$$\phi(x,y) = \phi_1(x)\phi_2(y),$$

then the Fourier transform can also be separated

$$\Phi(\omega,v) = \Phi_1(\omega)\Phi_2(v)$$

where

$$\Phi_1(\omega) = F(\phi_1(x)) \qquad \Phi_2(v) = F(\phi_2(y))$$

Mechanical Engineering
Carnegie Mellon

Separability

- Separability is applicable to the 2D Gaussian function

$$g(x, y) = g_1(x)g_2(y)$$

The Fourier transform of $g(x, y)$ is

$$F(g(x, y)) = G(\omega, v) = G_1(\omega)G_2(v)$$

where

$$G_1(\omega) = F(g_1(x)) \qquad G_2(v) = F(g_2(y))$$

Rotational Invariance

- Rotation of a 2D image about the origin of the (x, y) plane

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

- The 2D Fourier transform will likewise be rotated about the origin of the (ω, v) plane

$$\begin{bmatrix} \omega' \\ v' \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} \omega \\ v \end{bmatrix}$$

Projection

- Consider the projection P_x of $\phi(x, y)$ onto the x-axis

$$P_x(\phi(x, y)) = \int_{-\infty}^{\infty} \phi(x, y) dy$$

- We have separability:

$$F(\phi(x, y)) = \int_{-\infty}^{\infty} \left[\int_{-\infty}^{\infty} \phi(x, y) e^{-i2\pi\omega x} dx \right] e^{-i2\pi v y} dy = \Phi(\omega, v)$$

rewrite it by setting $v = 0$:

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi(x, y) e^{-i2\pi\omega x} dx dy = \Phi(\omega, 0) = F(P_x(\phi(x, y)))$$

In other words, the Fourier transform of the projection of $\phi(x, y)$ onto the x-axis is nothing but the 2D Fourier transform $\Phi(\omega, v)$ evaluated along the v-axis.

Mechanical Engineering
Carnegie Mellon

Combining Projection with Rotational Invariance

- The combination leads to the somewhat surprising result: projecting a 2D image onto any line passing through the origin in image space transforms into a 1D spectrum taken from the same line along the Fourier transform in frequency space.

- This result leads to a transform called Radon transform, central to the process of filtered back projection, by which images are formed in computed tomography (CT) and other useful applications.

Nonlinear Image Filtering with PDEs

- Most nonlinear filtering methods are built upon one of three strategies:
  - **Heuristics**: set of rules on pixels that are designed specifically to achieve a particular result.
  - **Statistics**: filters designed on robust statistics (e.g., median) or using properties from stochastic processes.
  - **Partial Differential Equations (PDEs)**: the input is some initial value in a PDE and the output is from the solution of that PDE.

Gaussian Blurring

- Gaussian blurring is a kind of low-pass transform, which can be understood in the frequency (Fourier) domain as filters that reduce high frequencies more than low frequencies. This is useful for denoising.

- The noise in an image has more energy in high frequencies than does the signal.

Gaussian Blurring

- An image $f(x, y)$ is a mapping from an image domain, usually a small rectangle, to a scalar field, the intensity.
- The Gaussian is a low-pass transform

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

$$G(u,v) = F(g) = e^{-\frac{(\mathrm{u}^2+\mathrm{v}^2)\sigma^2}{2}}$$

where F denotes the Fourier transform, and σ is the standard deviation, which is the effective width of the kernel.

Gaussian Blurring

- The Gaussian kernel has several nice properties: smooth, self-similar, which means that repeated applications of Gaussian filtering can be achieved with a single Gaussian filter.
- The diffusion equation (initial value problem):

$$\frac{\partial f}{\partial t} = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = \nabla \bullet c \nabla f$$

where $f(x, y, t)$ can be viewed as a sequence of images that starts with $f(x, y, 0) = I(x, y)$ and evolves over time to become more smooth. Here we choose $c = 1$ (isotropic diffusion).

Gaussian Blurring

- In the frequency domain ($F\{\partial f / \partial x\} = juF, \quad j^2 = -1$)

$$\frac{\partial F}{\partial t} = -(u^2 + v^2)F$$

- The solution is an exponential of the form:

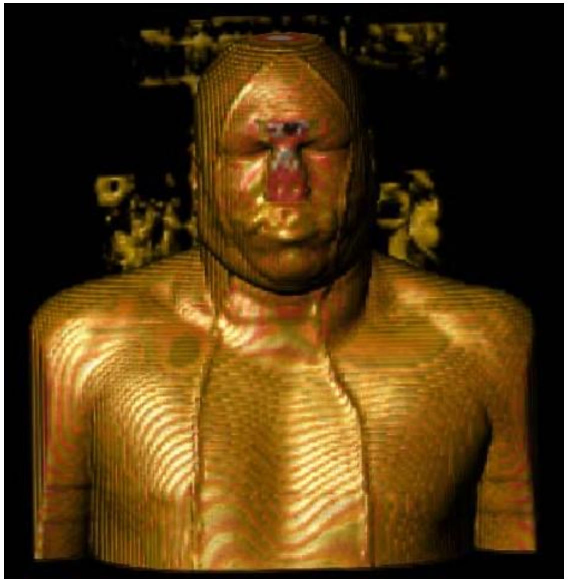
$$F(u,v,t) = e^{-(u^2+v^2)t} F(u,v,0)$$

$$f(x,y,t) = \frac{1}{\sqrt{4\pi t}} e^{-(x^2+y^2)/4t} \otimes f(x,y,0)$$

- The solution to the diffusion equation at a particular time is the same as convolving the initial condition with a Gaussian of standard deviation $\sigma = \sqrt{2t}$.

Gaussian Blurring

- Gaussian blurring can remove noise, but it also reduces high-frequency components of the image (or features such as sharp edges) at the same time.



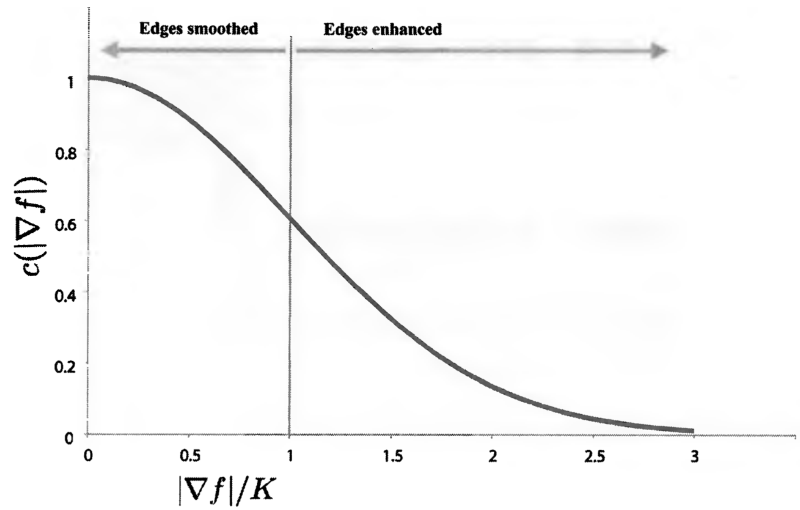
Noisy image

Gaussian filtering

Anisotropic Diffusion (Nonlinear)

- In order to preserve features, we choose c as a function of (x, y). $c(x, y)$ controls the rate of flow and allows it to be defined differently in each part of the image, i.e., lower near interesting features (edges).

$$\frac{\partial f}{\partial t} = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = \nabla \bullet c(|\nabla f|)\nabla f \qquad c(|\nabla f|) = e^{\frac{|\nabla f|^2}{2K^2}}$$



Mechanical Engineering
Carnegie Mellon

Filtering

- Nonlinear diffusion vs. linear diffusion

- Linear diffusion equation

$$\partial_t \phi - \nabla^2 \phi = 0$$

- Nonlinear diffusion equation

(Perona-Malik model, 1990)

$$\partial_t \phi - \mathrm{div}(\mathrm{g}(|\nabla \phi|)\nabla \phi) = 0$$

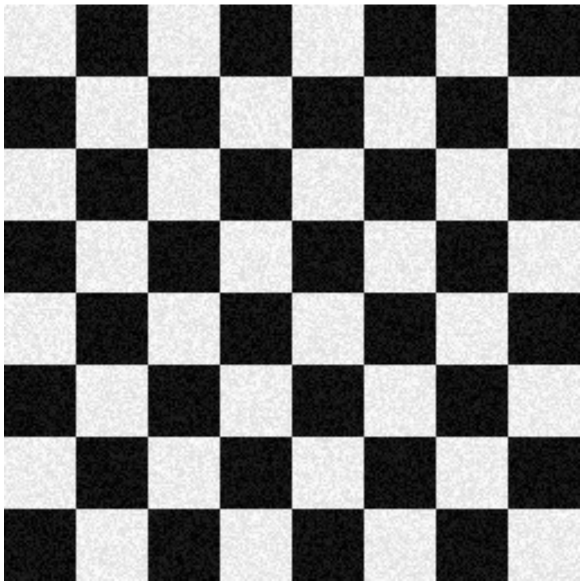
where $g(|\nabla \phi|) = \dfrac{1}{1 + |\nabla \phi|^2 / \lambda^2}$

Linear diffusion

Nonlinear diffusion

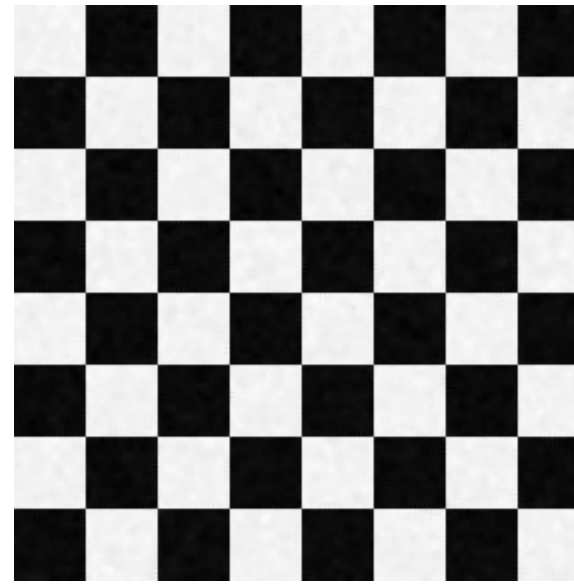
Filtering

- Examples



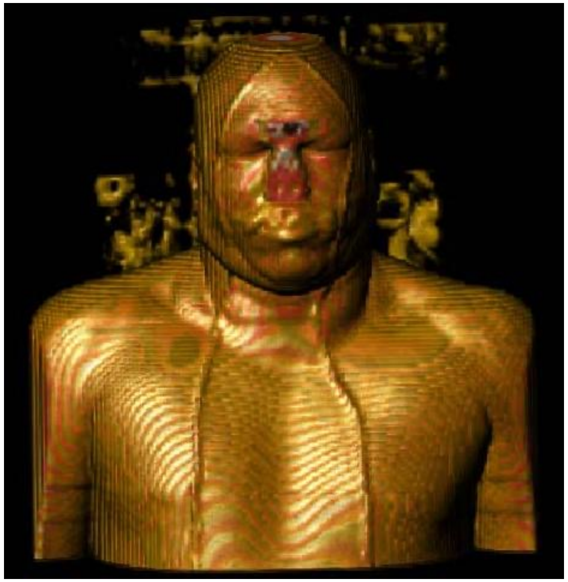
Noisy image

Linear diffusion



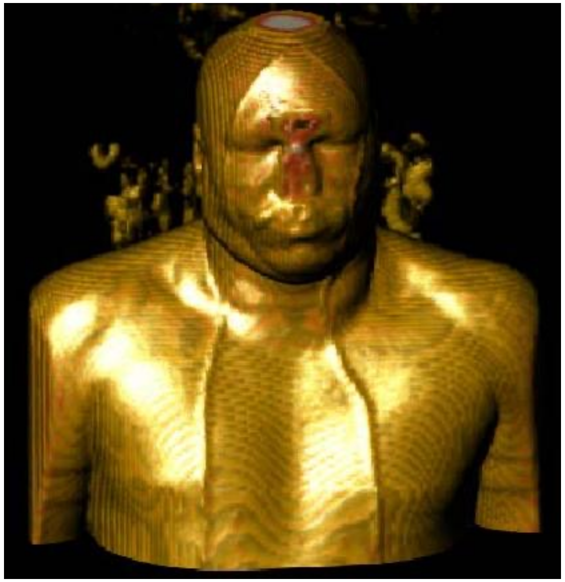
Nonlinear diffusion

Anisotropic Filtering



Noisy image

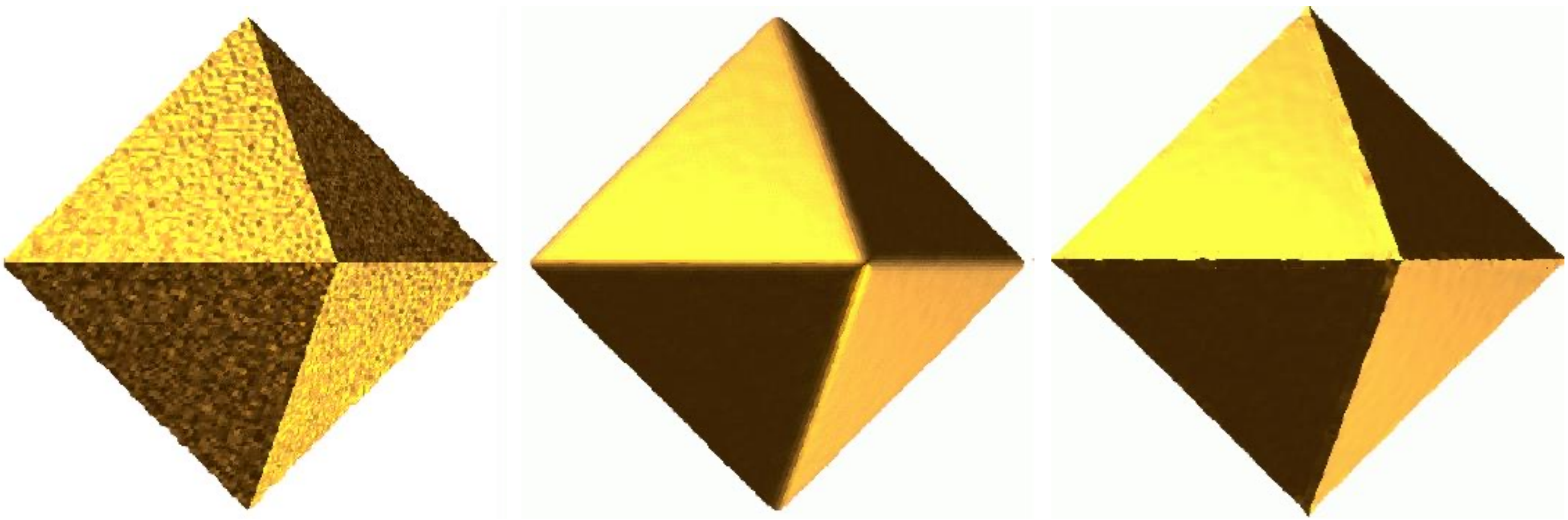
Gaussian filtering



Anisotropic diffusion

Filtering

- Extension of anisotropic diffusion to surfaces



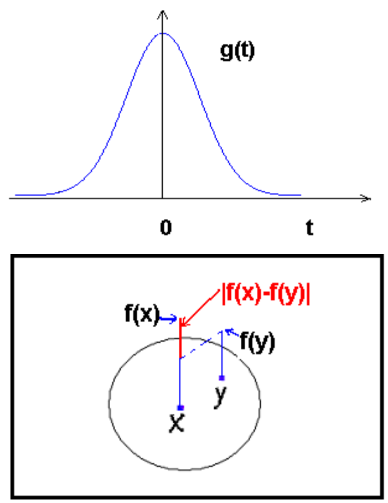
Noisy image | Linear diffusion | Nonlinear diffusion

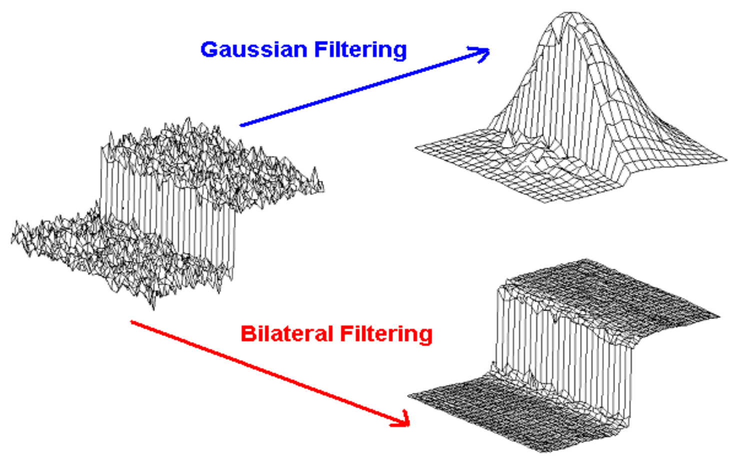
C. Bajaj, G. Xu
Anisotropic Diffusion of Subdivision Surfaces and Functions on Surfaces
ACM Transactions on Graphics, 22(1):4-32, 2003.

Mechanical Engineering
Carnegie Mellon

Filtering

- Input image is often noisy.
- Choices –
  1. Gaussian Filtering
  2. Bilateral Filtering
  3. Anisotropic Filtering





W. Jiang, M. Baker, Q. Wu, C. Bajaj, W. Chiu
Journal of Structural Biology, 144(5):114-122, 2003.

Bilateral Filtering

- Bilateral Filtering - Consider a low-pass domain filter applied to an image, the input image f_1 and output image f_2

Domain filtering:
$$f_2(\mathbf{x}) = k_d^{-1} \int_{\mathbf{y}} f_1(\mathbf{y}) c(\mathbf{y} - \mathbf{x}) d\mathbf{y}$$

$$k_d = \int_{\mathbf{y}} c(\mathbf{y} - \mathbf{x}) d\mathbf{y}$$

Range filtering:
$$f_2(\mathbf{x}) = k_r^{-1} \int_{\mathbf{y}} f_1(\mathbf{y}) s(f_1(\mathbf{y}) - f_1(\mathbf{x})) d\mathbf{y}$$

$$k_r = \int_{\mathbf{y}} s(f_1(\mathbf{y}) - f_1(\mathbf{x})) d\mathbf{y}$$

Bilateral filtering is the combination of domain and range filtering:

$$f_2(\mathbf{x}) = k_b^{-1} \int_{\mathbf{y}} f_1(\mathbf{y}) s(f_1(\mathbf{y}) - f_1(\mathbf{x})) c(\mathbf{y} - \mathbf{x}) d\mathbf{y}$$

$$k_b = \int_{\mathbf{y}} c(\mathbf{y} - \mathbf{x}) s(f_1(\mathbf{y}) - f_1(\mathbf{x})) d\mathbf{y}$$

Original

After anisotropic diffusion and bilateral filtering

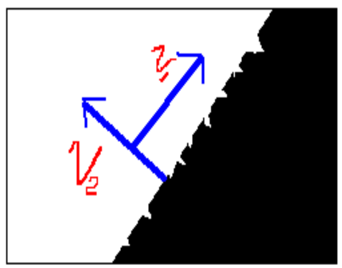
Filtering

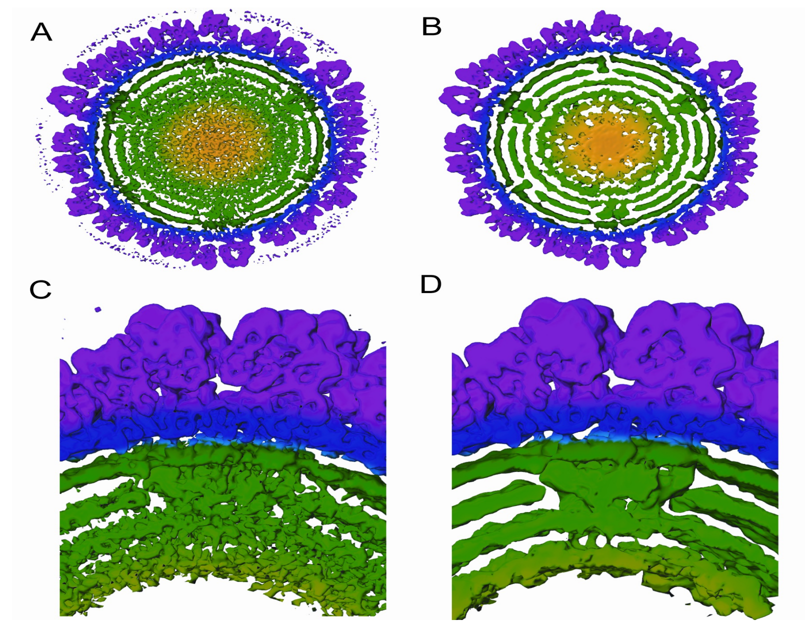
Bilateral filtering

$$h(x,\xi) = e^{-\frac{(x-\xi)^2}{2\sigma_d^2}} \cdot e^{-\frac{(f(x)-f(\xi))^2}{2\sigma_r^2}}$$

where σ_d and σ_r are parameters

and $f(.)$ is the image intensity value.





W. Jiang, M. Baker, Q. Wu, C. Bajaj, W. Chiu *Journal of Structural Biology*, 144(5):114-122, 2003.

Anisotropic diffusion filtering

$$\partial_t \phi - \mathrm{div}(a(|\nabla \phi_\sigma|)\nabla \phi) = 0$$

where *a* stands for the diffusion tensor determined by local curvature estimation.

C. Bajaj, G. Xu, *ACM Transactions on Graphics*, 22(1):4- 32, 2003.

Mechanical Engineering
Carnegie Mellon

References

- Yoo, Terry (Editor) (2004), Insight into Images, A K Peters, ISBN 1-56881-217-5. Chapter 2&4.
- Z. Yu, C. Bajaj. A Fast and Adaptive Algorithm for Image Contrast Enhancement. Proceedings of 2004 IEEE International Conference on Image Processing (ICIP'04), Volume 2, Oct. 24-27 2004, Pages 1001-1004.
- C. Bajaj, G. Xu, ACM Transactions on Graphics, (2003),22(1), pp. 4-32.
- W. Jiang, M. Baker, Q. Wu, C. Bajaj, W. Chiu, Journal of Structural Biology, 144, 5,(2003), Pages 114-122.
- Castleman, K. Digital Image Processing. Englewood Cliffs, NJ, Prentice Hall, 1996
- Oppenheim, A., and Willsky, A. Signals and Systems, (2nd Edition). Englewood Cliffs, NJ, Prentice Hall, 1996