Image Processing

Adam Finkelstein

Princeton University

COS 426, Spring 2005

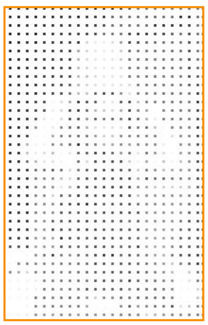
Image Processing

- Quantization
  - Uniform Quantization
  - Random dither
  - Ordered dither
  - Floyd-Steinberg dither
- Pixel operations
  - Add random noise
  - Add luminance
  - Add contrast
  - Add saturation
- Filtering
  - Blur
  - Detect edges
- Warping
  - Scale
  - Rotate
  - Warp
- Combining
  - Composite
  - Morph

What is an Image?

- An image is a 2D rectilinear array of samples

Continuous image



Digital image

Image Resolution

- Intensity resolution
  - Each pixel has only "Depth" bits for colors/intensities
- Spatial resolution
  - Image has only "Width" x "Height" pixels
- Temporal resolution
  - Monitor refreshes images at only "Rate" Hz

Typical Resolutions

	Width x Height	Depth	Rate
NTSC	640 x 480	8	30
Workstation	1280 x 1024	24	75
Film	3000 x 2000	12	24
Laser Printer	6600 x 5100	1	-

Sources of Error

- Intensity quantization
  - Not enough intensity resolution
- Spatial aliasing
  - Not enough spatial resolution
- Temporal aliasing
  - Not enough temporal resolution

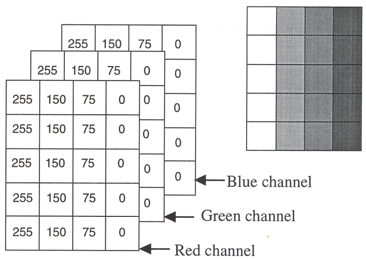
$$E^2 = \sum_{(x,y)} \left(I(x,y) - P(x,y)\right)^2$$

Overview

- Image representation
  - What is an image?
- Halftoning and dithering
  - Reduce visual artifacts due to quantization
- Sampling and reconstruction
  - Reduce visual artifacts due to aliasing

Quantization

- Artifacts due to limited intensity resolution
  - o Frame buffers have limited number of bits per pixel
  - o Physical devices have limited dynamic range

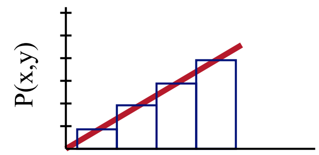


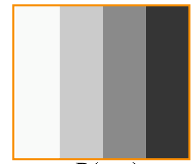
Uniform Quantization

$P(x, y) = \text{round}(\ I(x, y)\ )$

where round() chooses nearest value that can be represented.

I(x,y)

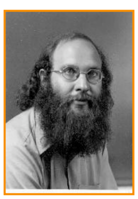




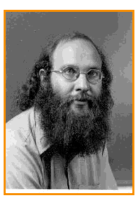
P(x,y)
(2 bits per pixel)

Uniform Quantization

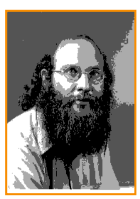
- Images with decreasing bits per pixel:



8 bits



4 bits



2 bits

1 bit

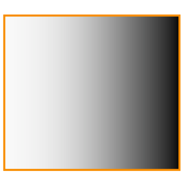
Notice contouring.

Reducing Effects of Quantization

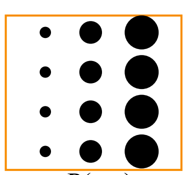
- Halftoning
  - o Classical halftoning
- Dithering
  - o Random dither
  - o Ordered dither
  - o Error diffusion dither

Classical Halftoning

- Use dots of varying size to represent intensities
  - o Area of dots proportional to intensity in image

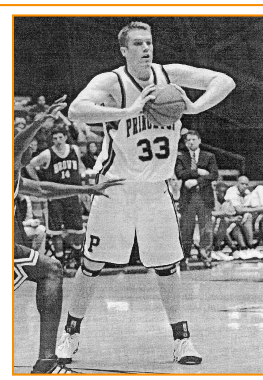


I(x,y)

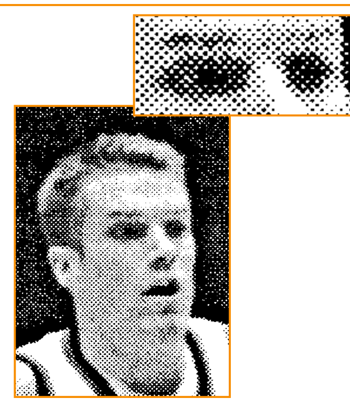


P(x,y)

Classical Halftoning

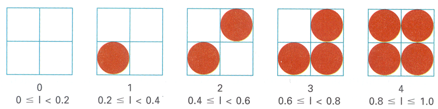


From *Town Topics*, 2/2/05



Halftone patterns

- Use cluster of pixels to represent intensity
  - o Trade spatial resolution for intensity resolution

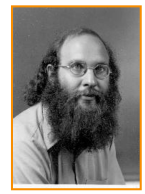


Q: In this case, would we use four "halftoned" pixels in place of one original pixel?

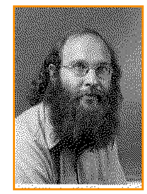
Figure 14.37 from H&B

Dithering

- Distribute errors among pixels
  - o Exploit spatial integration in our eye
  - o Display greater range of perceptible intensities

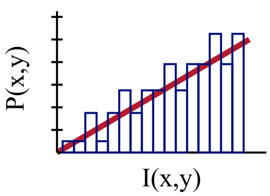

Original (8 bits)

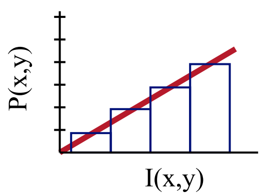

Uniform Quantization (1 bit)


Floyd-Steinberg Dither (1 bit)

Random Dither

- Randomize quantization errors
  - o Errors appear as noise





P(x, y) = round(I(x, y) + noise(x,y))

Random Dither


Original (8 bits)


Uniform Quantization (1 bit)


Random Dither (1 bit)

Ordered Dither

- Pseudo-random quantization errors
  - o Matrix stores pattern of threshholds

```
i = x mod n
j = y mod n
error = I(x,y) - floor( I(x,y) )
thresh = ( Dn(i,j) + 0.5 ) / (n+1)
if ( error > thresh )
      P(x,y) = ceil( I(x, y) )
else
      P(x,y) = floor( I(x,y) )
```

$$D_2 = \begin{bmatrix} 3 & 1 \\ 0 & 2 \end{bmatrix}$$

Ordered Dither

- Bayer's ordered dither matrices

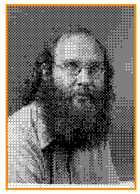
$$D_n = \begin{bmatrix} 4D_{n/2} + D_2(1,1)U_{n/2} & 4D_{n/2} + D_2(1,2)U_{n/2} \\ 4D_{n/2} + D_2(2,1)U_{n/2} & 4D_{n/2} + D_2(2,2)U_{n/2} \end{bmatrix}$$

$$D_2 = \begin{bmatrix} 3 & 1 \\ 0 & 2 \end{bmatrix} \qquad D_4 = \left[\begin{array}{cc|cc} 15 & 7 & 13 & 5 \\ 3 & 11 & 1 & 9 \\ \hline 12 & 4 & 14 & 6 \\ 0 & 8 & 2 & 10 \end{array}\right]$$

Ordered Dither

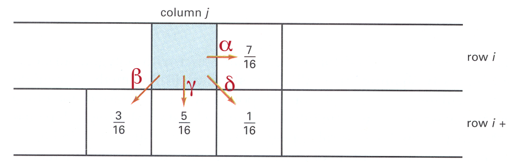

Original
(8 bits)


Random
Dither
(1 bit)


Ordered
Dither
(1 bit)

Error Diffusion Dither

- Spread quantization error over neighbor pixels
  - Error dispersed to pixels right and below



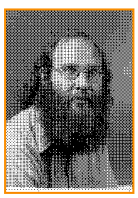
$$\alpha + \beta + \gamma + \delta = 1.0$$

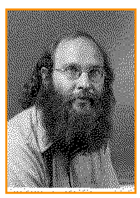
Figure 14.42 from H&B

Error Diffusion Dither


Original
(8 bits)


Random
Dither
(1 bit)


Ordered
Dither
(1 bit)


Floyd-Steinberg
Dither
(1 bit)

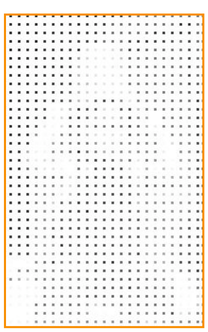
Overview

- Image representation
  - What is an image?
- Halftoning and dithering
  - Reduce visual artifacts due to quantization
- Sampling and reconstruction
  - Reduce visual artifacts due to aliasing

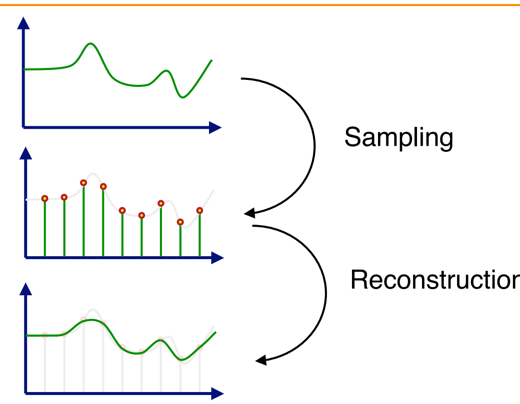
What is an Image?

- An image is a 2D rectilinear array of samples


Continuous image


Digital image

Sampling and Reconstruction



Sampling and Reconstruction

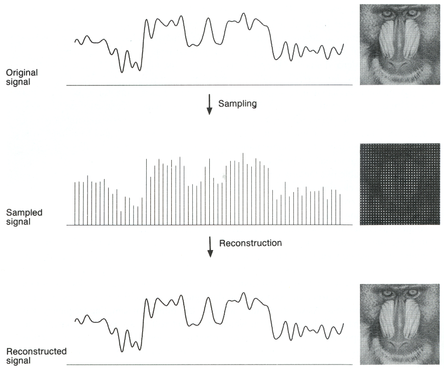


Figure 19.9 FvDFH

Image Processing

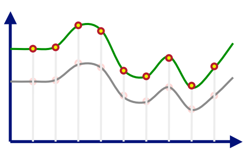
- Quantization
  - o Uniform Quantization
  - o Random dither
  - o Ordered dither
  - o Floyd-Steinberg dither
- Pixel operations
  - o Add random noise
  - o Add luminance
  - o Add contrast
  - o Add saturation
- Filtering
  - o Blur
  - o Detect edges
- Warping
  - o Scale
  - o Rotate
  - o Warps
- Combining
  - o Composite
  - o Morph

Adjusting Brightness

- Simply scale pixel components
  - o Must clamp to range (e.g., 0 to 255)

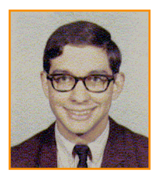
Original

Brighter

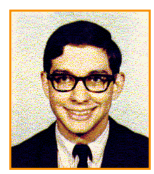


Adjusting Contrast

- Compute mean luminance L for all pixels
  - o luminance = 0.30*r + 0.59*g + 0.11*b
- Scale deviation from L for each pixel component
  - o Must clamp to range (e.g., 0 to 255)



Original



More Contrast

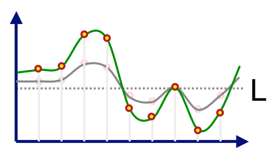
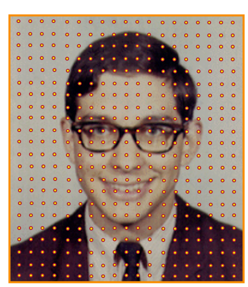


Image Processing

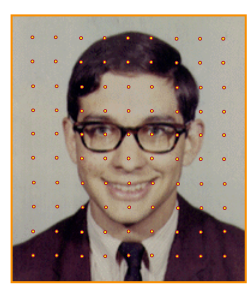
- Quantization
  - o Uniform Quantization
  - o Random dither
  - o Ordered dither
  - o Floyd-Steinberg dither
- Pixel operations
  - o Add random noise
  - o Add luminance
  - o Add contrast
  - o Add saturation
- Filtering
  - o Blur
  - o Detect edges
- Warping
  - o Scale
  - o Rotate
  - o Warps
- Combining
  - o Composite
  - o Morph

Image Processing

- Consider reducing the image resolution



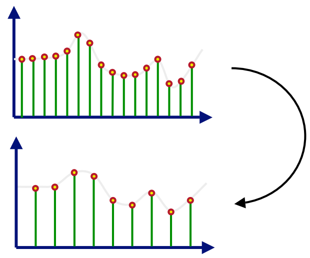
Original image



1/4 resolution

Image Processing

- Image processing is a resampling problem

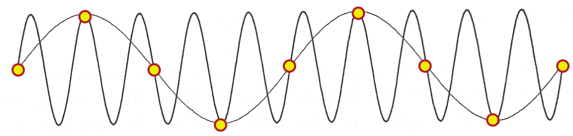


Resampling

Thou shalt avoid aliasing!

Aliasing

- In general:
  - Artifacts due to under-sampling or poor reconstruction
- Specifically, in graphics:
  - Spatial aliasing
  - Temporal aliasing

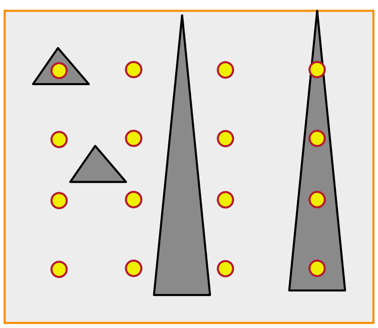


Under-sampling

Figure 14.17 FvDFH

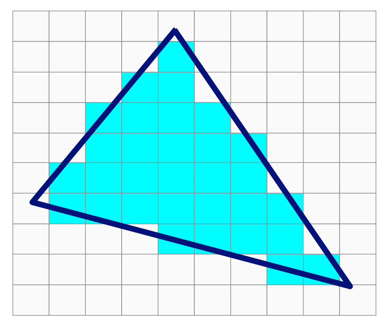
Spatial Aliasing

- Artifacts due to limited spatial resolution



Spatial Aliasing

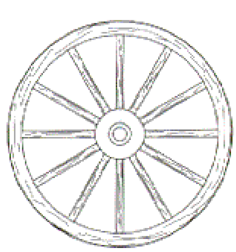
- Artifacts due to limited spatial resolution



"Jaggies"

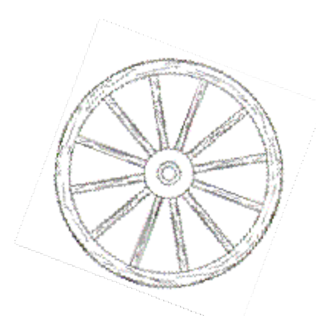
Temporal Aliasing

- Artifacts due to limited temporal resolution
  - Strobing
  - Flickering



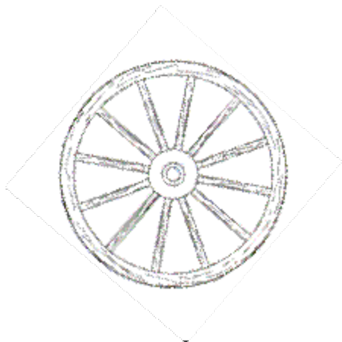
Temporal Aliasing

- Artifacts due to limited temporal resolution
  - Strobing
  - Flickering



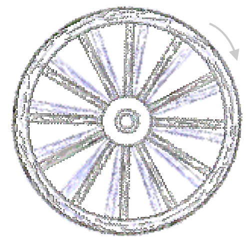
Temporal Aliasing

- Artifacts due to limited temporal resolution
  - Strobing
  - Flickering



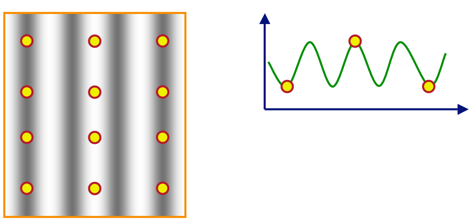
Temporal Aliasing

- Artifacts due to limited temporal resolution
  - Strobing
  - Flickering



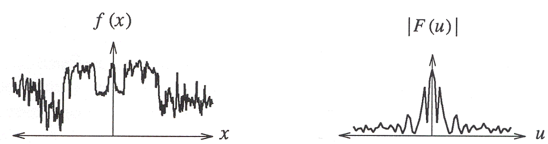
Sampling Theory

- When does aliasing happen?
  - How many samples are required to represent a given signal without loss of information?
  - What signals can be reconstructed without loss for a given sampling rate?



Spectral Analysis

- Spatial domain:
  - Function: f(x)
  - Filtering: convolution
- Frequency domain:
  - Function: F(u)
  - Filtering: multiplication



Any signal can be written as a sum of periodic functions.

Fourier Transform

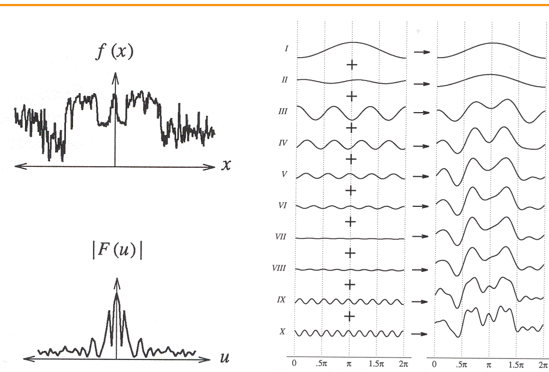


Figure 2.6 Wolberg

Fourier Transform

- Fourier transform:

$$F(u) = \int_{-\infty}^{\infty} f(x) e^{-i2\pi xu} dx$$

- Inverse Fourier transform:

$$f(x) = \int_{-\infty}^{\infty} F(u) e^{+i2\pi ux} du$$

Sampling Theorem

- A signal can be reconstructed from its samples, if the original signal has no frequencies above 1/2 the sampling frequency - Shannon
- The minimum sampling rate for bandlimited function is called “Nyquist rate”

A signal is bandlimited if its highest frequency is bounded. The frequency is called the bandwidth.

Antialiasing

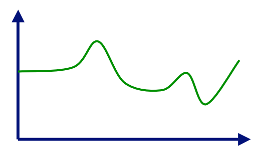
- Sample at higher rate
  - Not always possible
  - Doesn’t always solve problem
- Pre-filter to form bandlimited signal
  - Form bandlimited function using low-pass filter
  - Trades aliasing for blurring

Image Processing

↓ Real world
Sample
↓ Discrete samples (pixels)
Reconstruct
↓ Reconstructed function
Transform
↓ Transformed function
Filter
↓ Bandlimited function
Sample
↓ Discrete samples (pixels)
Reconstruct
↓ Display

Image Processing

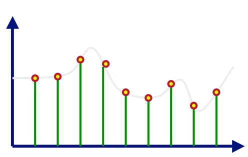
↓ **Real world**
Sample
↓ Discrete samples (pixels)
Reconstruct
↓ Reconstructed function
Transform
↓ Transformed function
Filter
↓ Bandlimited function
Sample
↓ Discrete samples (pixels)
Reconstruct
↓ Display



Continuous Function

Image Processing

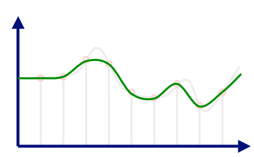
↓ Real world
Sample
↓ Discrete samples (pixels)
Reconstruct
↓ Reconstructed function
Transform
↓ Transformed function
Filter
↓ Bandlimited function
Sample
↓ Discrete samples (pixels)
Reconstruct
↓ Display



Discrete Samples

Image Processing

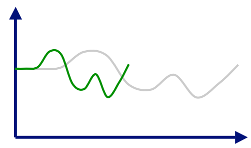
↓ Real world
Sample
↓ Discrete samples (pixels)
Reconstruct
↓ Reconstructed function
Transform
↓ Transformed function
Filter
↓ Bandlimited function
Sample
↓ Discrete samples (pixels)
Reconstruct
↓ Display



Reconstructed Function

Image Processing

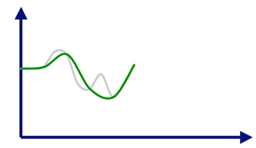
Real world
Sample
Discrete samples (pixels)
Reconstruct
Reconstructed function
Transform
Transformed function
Filter
Bandlimited function
Sample
Discrete samples (pixels)
Reconstruct
Display



Transformed Function

Image Processing

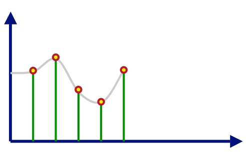
Real world
Sample
Discrete samples (pixels)
Reconstruct
Reconstructed function
Transform
Transformed function
Filter
Bandlimited function
Sample
Discrete samples (pixels)
Reconstruct
Display



Bandlimited Function

Image Processing

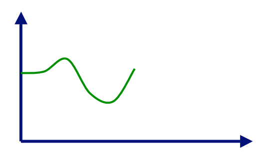
Real world
Sample
Discrete samples (pixels)
Reconstruct
Reconstructed function
Transform
Transformed function
Filter
Bandlimited function
Sample
Discrete samples (pixels)
Reconstruct
Display



Discrete samples

Image Processing

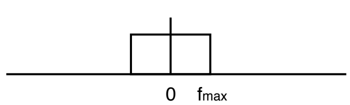
Real world
Sample
Discrete samples (pixels)
Reconstruct
Reconstructed function
Transform
Transformed function
Filter
Bandlimited function
Sample
Discrete samples (pixels)
Reconstruct
Display



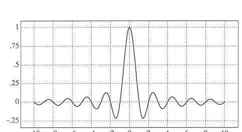
Display

Ideal Bandlimiting Filter

- Frequency domain



- Spatial domain



$$Sinc(x) = \frac{\sin \pi x}{\pi x}$$

Figure 4.5 Wolberg

Practical Image Processing

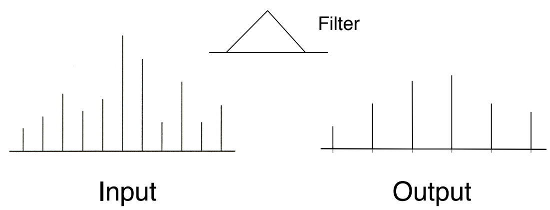
- Finite low-pass filters
  - Point sampling (bad)
  - Triangle filter
  - Gaussian filter

Real world
Sample
Discrete samples (pixels)
Reconstruct
Reconstructed function
Transform
Transformed function
Filter
Bandlimited function
Sample
Discrete samples (pixels)
Reconstruct
Display

Convolution

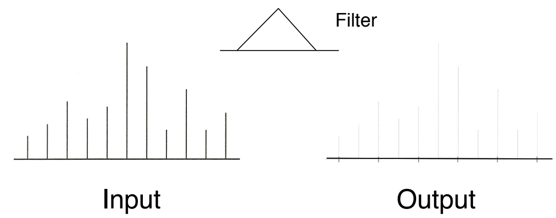
Convolution

- Spatial domain: output pixel is weighted sum of pixels in neighborhood of input image
  - Pattern of weights is the "filter"



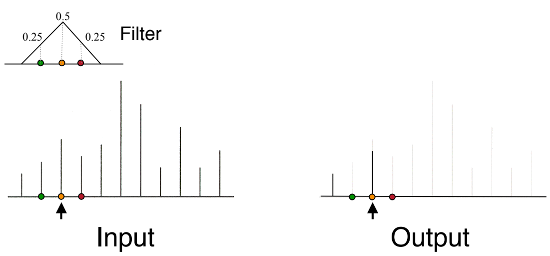
Convolution with a Triangle Filter

- Example 1:



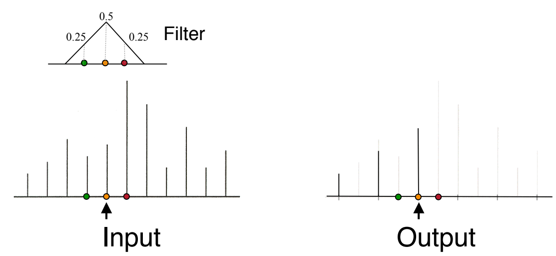
Convolution with a Triangle Filter

- Example 1:



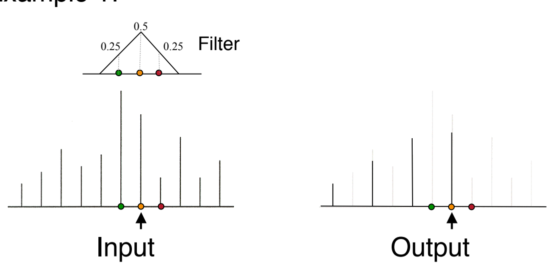
Convolution with a Triangle Filter

- Example 1:



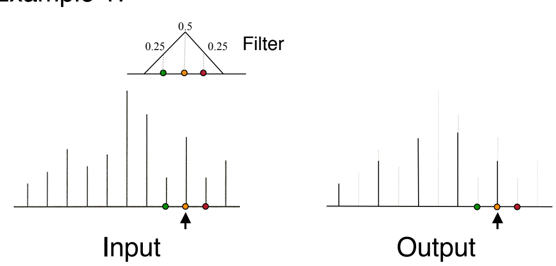
Convolution with a Triangle Filter

- Example 1:



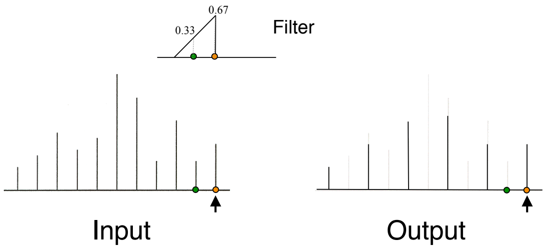
Convolution with a Triangle Filter

- Example 1:



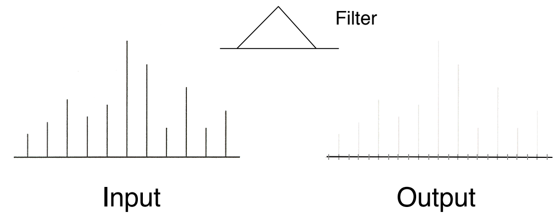
Convolution with a Triangle Filter

- Example 1:



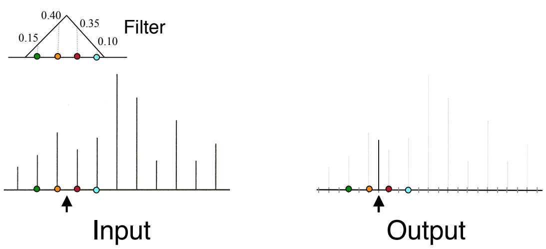
Convolution with a Triangle Filter

- Example 2:



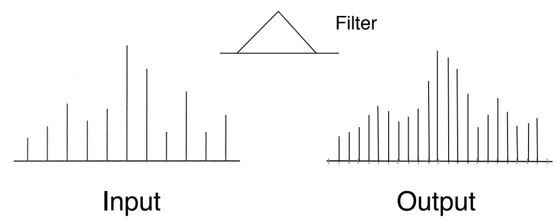
Convolution with a Triangle Filter

- Example 2:



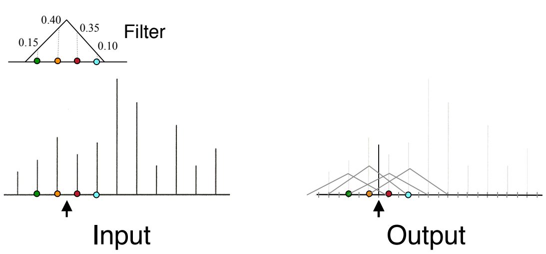
Convolution with a Triangle Filter

- Example 2:



Convolution with a Triangle Filter

- Example 2:



Convolution with a Triangle Filter

- Example 3:

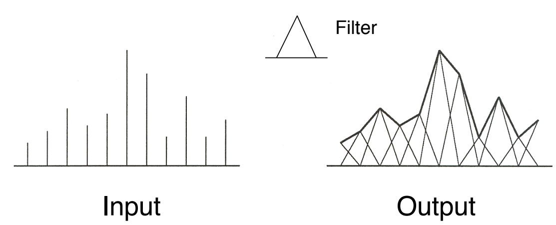
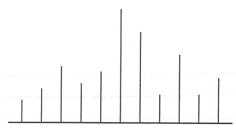


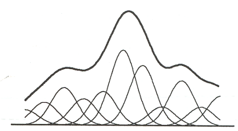
Figure 2.4 Wolberg

Convolution with a Gaussian Filter

- Example 4:



Input



Output

Figure 2.4 Wolberg

Image Processing

- Quantization
  - Uniform Quantization
  - Random dither
  - Ordered dither
  - Floyd-Steinberg dither
- Pixel operations
  - Add random noise
  - Add luminance
  - Add contrast
  - Add saturation
- Filtering
  - Blur
  - Detect edges
- Warping
  - Scale
  - Rotate
  - Warps
- Combining
  - Composite
  - Morph

Adjust Blurriness

- Convolve with a filter whose entries sum to one
  - Each pixel becomes a weighted average of its neighbors

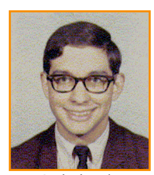
Original

Blur

$$\text{Filter} = \begin{bmatrix} \frac{1}{16} & \frac{2}{16} & \frac{1}{16} \\ \frac{2}{16} & \frac{4}{16} & \frac{2}{16} \\ \frac{1}{16} & \frac{2}{16} & \frac{1}{16} \end{bmatrix}$$

Edge Detection

- Convolve with a filter that finds differences between neighbor pixels



Original

Detect edges

$$\text{Filter} = \begin{bmatrix} -1 & -1 & -1 \\ -1 & +8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Image Processing

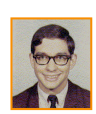
- Quantization
  - Uniform Quantization
  - Random dither
  - Ordered dither
  - Floyd-Steinberg dither
- Pixel operations
  - Add random noise
  - Add luminance
  - Add contrast
  - Add saturation
- Filtering
  - Blur
  - Detect edges
- Warping
  - Scale
  - Rotate
  - Warps
- Combining
  - Composite
  - Morph

Scaling

- Resample with triangle or Gaussian filter

More on this next lecture!

Original



1/4X resolution

4X resolution

Image Processing

- Image processing is a resampling problem
  - Avoid aliasing
  - Use filtering





Summary

- Image representation
  - A pixel is a sample, not a little square
  - Images have limited resolution
- Halftoning and dithering
  - Reduce visual artifacts due to quantization
  - Distribute errors among pixels
    - » Exploit spatial integration in our eye
- Sampling and reconstruction
  - Reduce visual artifacts due to aliasing
  - Filter to avoid undersampling
    - » Blurring is better than aliasing