Image Processing and Sampling

Greg Humphreys

CS445: Intro Graphics

University of Virginia, Fall 2005

Overview

- Image representation
  - What is an image?
- Halftoning and dithering
  - Trade spatial resolution for intensity resolution
  - Reduce visual artifacts due to quantization
- Sampling and reconstruction
  - Key steps in image processing
  - Avoid visual artifacts due to aliasing

What is an Image?

- An image is a 2D rectilinear array of pixels

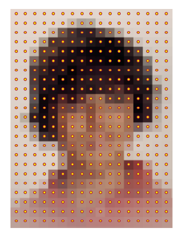
Continuous image

Digital image

What is an Image?

- An image is a 2D rectilinear array of pixels

Continuous image



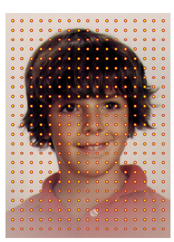
Digital image

A pixel is a sample, not a little square!

What is an Image?

- An image is a 2D rectilinear array of pixels

Continuous image



Digital image

A pixel is a sample, not a little square!

Image Acquisition

- Pixels are samples from continuous function
  - Photoreceptors in eye
  - CCD cells in digital camera
  - Rays in virtual camera

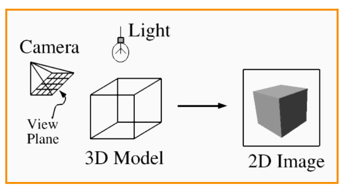
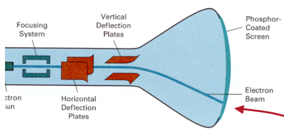


Image Display

- Re-create continuous function from samples
  - Example: cathode ray tube



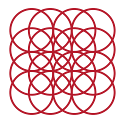


Image is reconstructed by displaying pixels with finite area (Gaussian)

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

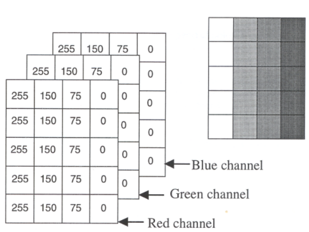
$$E^2 = \sum_{(x,y)} \left(I(x,y) - P(x,y)\right)^2$$

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

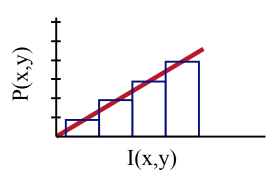
Quantization

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

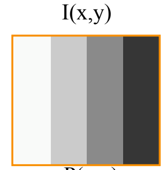


Uniform Quantization

$P(x, y) = \mathrm{trunc}(I(x, y) + 0.5)$



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

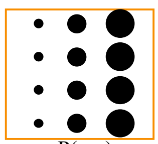
Reducing Effects of Quantization

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

Classical Halftoning

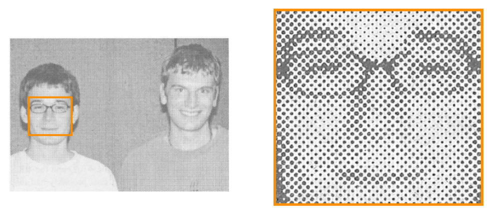
- Use dots of varying size to represent intensities
  - Area of dots proportional to intensity in image

I(x,y)



P(x,y)

Classical Halftoning



Newspaper image from *North American Bridge Championships Bulletin*, Summer 2003

Halftone patterns

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

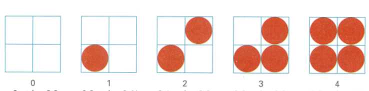
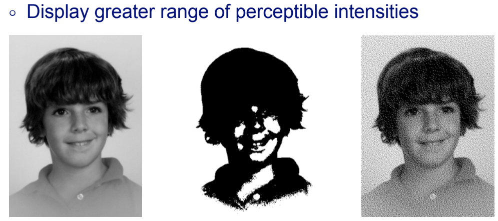


Figure 14.37 from H&B

Dithering

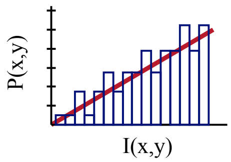
- Distribute errors among pixels
  - Exploit spatial integration in our eye
  - Display greater range of perceptible intensities

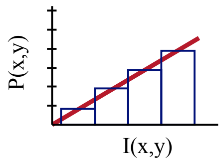


Original (8 bits) | Uniform Quantization (1 bit) | Floyd-Steinberg Dither (1 bit)

Random Dither

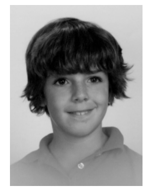
- Randomize quantization errors
  - Errors appear as noise





P(x, y) = trunc(I(x, y) + noise(x,y) + 0.5)

Random Dither



Original
(8 bits)

Uniform
Quantization
(1 bit)

Random
Dither
(1 bit)

Ordered Dither

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

```
i = x mod n
j = y mod n
e = I(x,y) - trunc(I(x,y))
if (e > D(i,j))
      P(x,y) = ceil(I(x, y))
else
      P(x,y) = floor(I(x,y))
```

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

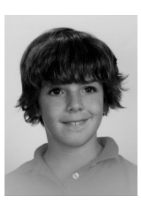
Ordered Dither

- Bayer's ordered dither matrices

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

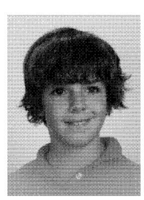
$$D_2 = \begin{bmatrix} 3 & 1 \\ 0 & 2 \end{bmatrix} \qquad D_4 = \begin{bmatrix} 15 & 7 & 13 & 5 \\ 3 & 11 & 1 & 9 \\ 12 & 4 & 14 & 6 \\ 0 & 8 & 2 & 10 \end{bmatrix}$$

Ordered Dither



Original
(8 bits)

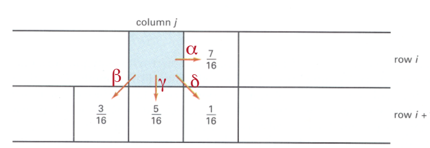
Uniform
Quantization
(1 bit)



4x4 Ordered
Dither
(1 bit)

Error Diffusion Dither

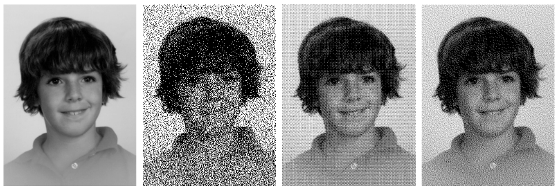
- Spread quantization error over neighbor pixels
  - Error dispersed to pixels right and below



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

Figure 14.42 from H&B

Dither Comparison

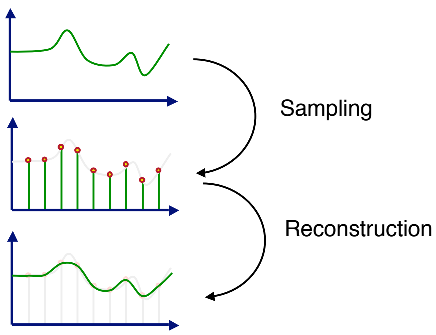


Original (8 bits) | Random Dither (1 bit) | Ordered Dither (1 bit) | Floyd-Steinberg Dither (1 bit)

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

Sampling and Reconstruction



Sampling and Reconstruction

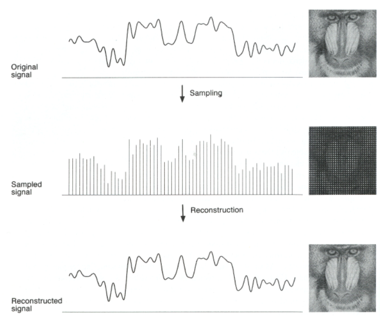
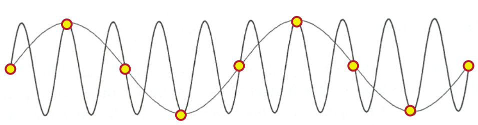


Figure 19.9 FvDFH

Aliasing

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

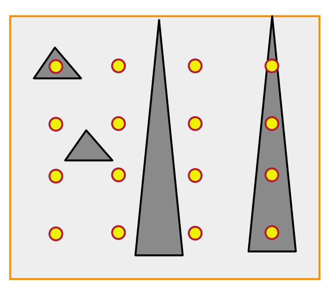


Under-sampling

Figure 14.17 FvDFH

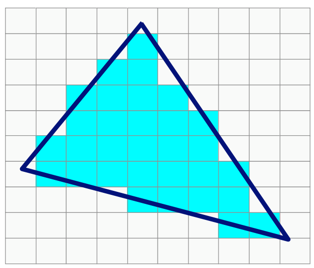
Spatial Aliasing

- Artifacts due to limited spatial resolution



Spatial Aliasing

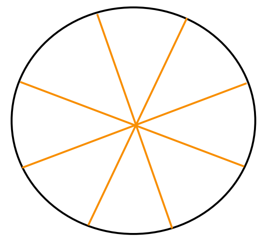
- Artifacts due to limited spatial resolution



"Jaggies"

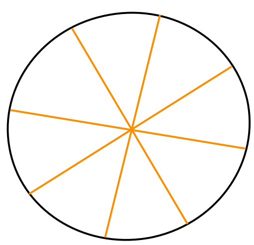
Temporal Aliasing

- Artifacts due to limited temporal resolution
  - Strobing
  - Flickering



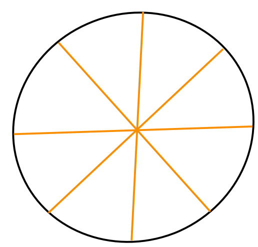
Temporal Aliasing

- Artifacts due to limited temporal resolution
  - Strobing
  - Flickering



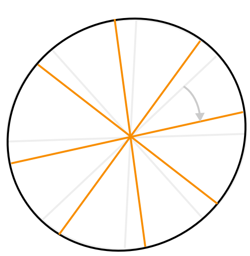
Temporal Aliasing

- Artifacts due to limited temporal resolution
  - Strobing
  - Flickering



Temporal Aliasing

- Artifacts due to limited temporal resolution
  - Strobing
  - Flickering



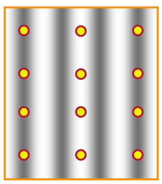
Antialiasing

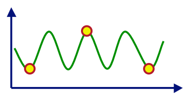
- Sample at higher rate
  - Not always possible
  - Doesn't always solve problem
- Pre-filter to form bandlimited signal
  - Form bandlimited function (low-pass filter)
  - Trades aliasing for blurring

Must consider sampling theory!

Sampling Theory

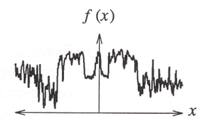
- How many samples are required to represent a given signal without loss of information?
- What signals can be reconstructed without loss for a given sampling rate?

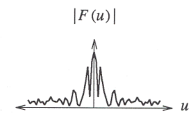




Spectral Analysis

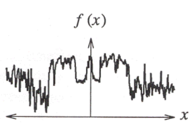
- Spatial domain:
  - Function: f(x)
  - Filtering: convolution
- Frequency domain:
  - Function: F(u)
  - Filtering: multiplication

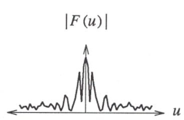




Any signal can be written as a sum of periodic functions.

Fourier Transform





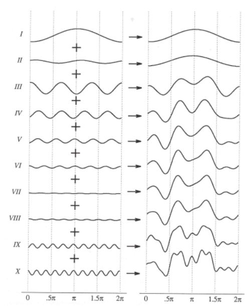


Figure 2.6 Wolberg

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
- Nyquist rate (or Nyquist limit)

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

Convolution

- Convolution of two functions (= filtering):

$$g(x) = f(x) \otimes h(x) = \int_{-\infty}^{\infty} f(\lambda) h(x - \lambda) d\lambda$$

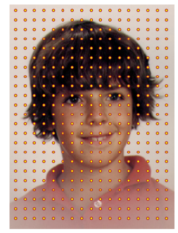
- Convolution theorem
  - Convolution in frequency domain is same as multiplication in spatial domain, and vice-versa

Image Processing

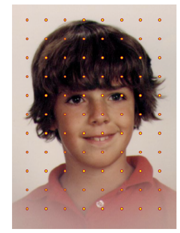
- Quantization
  - Uniform Quantization
  - Random dither
  - Ordered dither
  - Floyd-Steinberg dither
- Pixel operations
  - Add random noise
  - Add luminance
  - Add contrast
  - Add saturation
- Filtering
  - Blur
  - Detect edges
- Warping
  - Scale
  - Rotate
  - Warps
- Combining
  - Morphs
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

Image Processing

- Consider reducing the image resolution



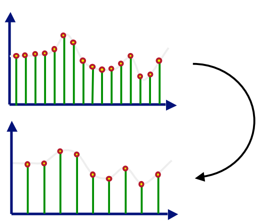
Original image



1/4 resolution

Image Processing

- Image processing is a resampling problem



Resampling

Thou shalt avoid aliasing!

Antialiasing in Image Processing

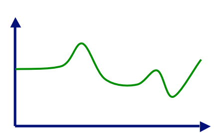
- General Strategy
  - Pre-filter transformed image via convolution with low-pass filter to form bandlimited signal
- Rationale
  - Prefer blurring over aliasing

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

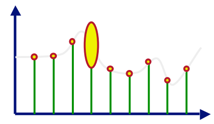
↓ **Real world**
Sample
↓ Discrete samples (pixels)
Reconstruct
↓ Reconstructed function
Transform
↓ Transformed function
Filter
↓ Bandlimited function
Sample
↓ Discrete samples (pixels)
Reconstruct
↓ Display



Continuous Function

Image Processing

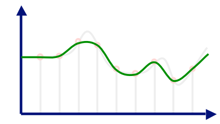
Real world
Sample
Discrete samples (pixels)
Reconstruct
Reconstructed function
Transform
Transformed function
Filter
Bandlimited function
Sample
Discrete samples (pixels)
Reconstruct
Display



Discrete Samples

Image Processing

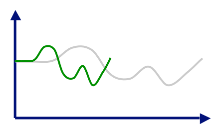
Real world
Sample
Discrete samples (pixels)
Reconstruct
Reconstructed function
Transform
Transformed function
Filter
Bandlimited function
Sample
Discrete samples (pixels)
Reconstruct
Display



Reconstructed Function

Image Processing

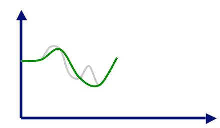
Real world
Sample
Discrete samples (pixels)
Reconstruct
Reconstructed function
Transform
Transformed function
Filter
Bandlimited function
Sample
Discrete samples (pixels)
Reconstruct
Display



Transformed Function

Image Processing

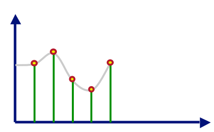
Real world
Sample
Discrete samples (pixels)
Reconstruct
Reconstructed function
Transform
Transformed function
Filter
Bandlimited function
Sample
Discrete samples (pixels)
Reconstruct
Display



Bandlimited Function

Image Processing

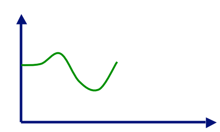
Real world
Sample
Discrete samples (pixels)
Reconstruct
Reconstructed function
Transform
Transformed function
Filter
Bandlimited function
Sample
Discrete samples (pixels)
Reconstruct
Display



Discrete samples

Image Processing

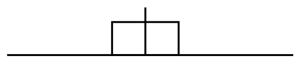
Real world
Sample
Discrete samples (pixels)
Reconstruct
Reconstructed function
Transform
Transformed function
Filter
Bandlimited function
Sample
Discrete samples (pixels)
Reconstruct
Display



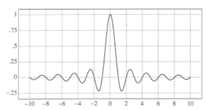
Display

Ideal Low-Pass Filter

- Frequency domain



- Spatial domain

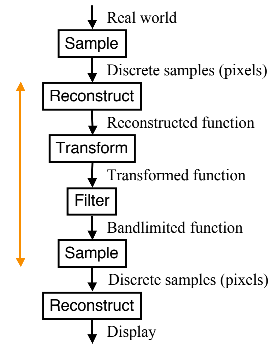


$$Sinc(x) = \frac{\sin \pi x}{\pi x}$$

Figure 4.5 Wolberg

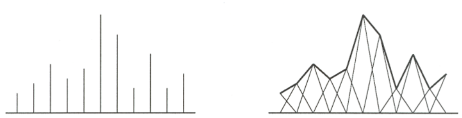
Practical Image Processing

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



Triangle Filter

- Convolution with triangle filter

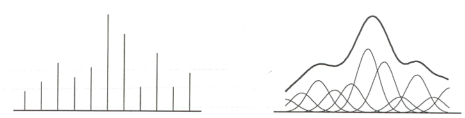


Input Output

Figure 2.4 Wolberg

Gaussian Filter

- Convolution with Gaussian filter



Input Output

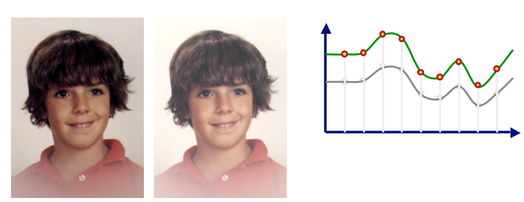
Figure 2.4 Wolberg

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
  - Morphs
  - Composite

Brightness

- Simply scale pixel components
  - Must clamp to range (e.g., 0 to 255)
- Trick: interpolate (extrapolate) from a black image



Original Brighter

Contrast

- Compute mean luminance L for all pixels
  - luminance = 0.30*r + 0.59*g + 0.11*b
- Scale deviation from L for each pixel component
- Interpolate (extrapolate) from an average gray image


Original


More Contrast

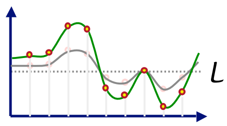


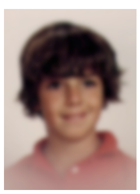
Image Processing

- 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
  - Morphs
  - Composite

Blur and Sharpen

- Convolve with a filter whose entries sum to one
  - Each pixel becomes a weighted average of its neighbors
- Trick: extrapolate from blurry image = sharpen!
  - "Unsharp mask" in Photoshop


Original


Blur


Sharpen

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

Saturation

- Interpolate (extrapolate) from grayscale version


Original


Less saturated


More saturated


Negatively Saturated

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
  - Morphs
  - Composite

Scaling

- Resample with triangle or Gaussian filter

More on this next lecture!

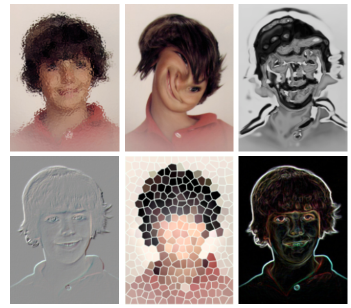
Original

½ resolution

2x 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