Image processing

How computer images work?

Continuous real image

↓ *Digitization (e.g. scanning)*

square array of numbers (abstract pixels)

↓ *each physical pixel covers an area*

display (physical pixels)

↓ *The eye blurs pixels into continuous image*

perceived image

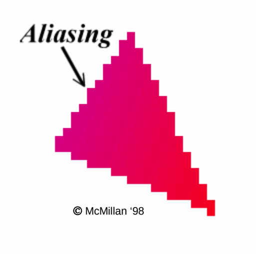
What can go wrong?

Pipeline: sample - process - reconstruct

All kinds of artifacts can appear

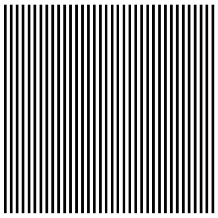
- jaggies
- alias patterns
- moire patterns
- temporal aliasing (wheels going wrong way)

Question: how do we avoid all this?

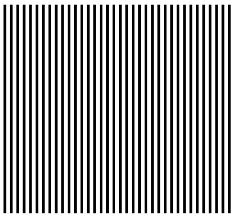


© McMillan '98

Aliasing

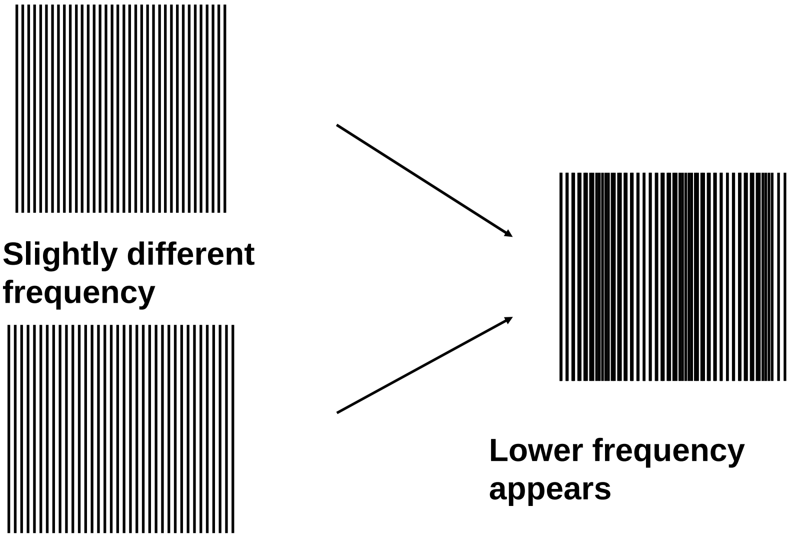


Slightly different frequency



What can go wrong?

Aliasing

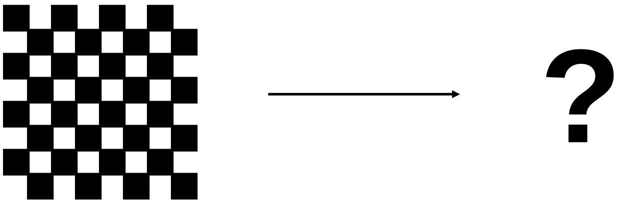


Slightly different frequency

Lower frequency appears

Shrinking

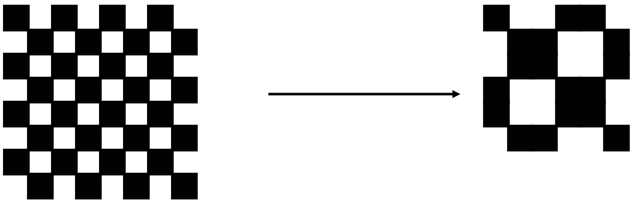
Naïve 1.5x shrinking : drop 2 out of 3



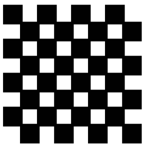
What do we get?

Shrinking

Naïve 1.5x shrinking: drop 1 out of 3



But we want



(impossible, not enough samples)

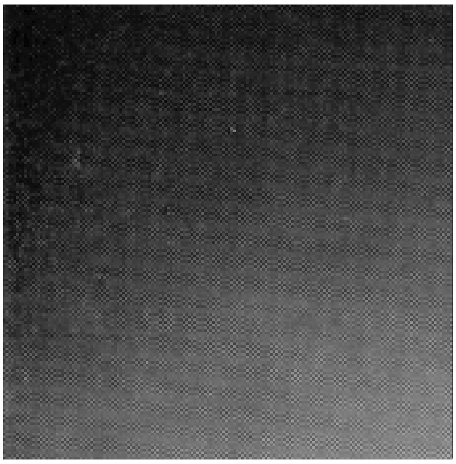
Shrinking

original

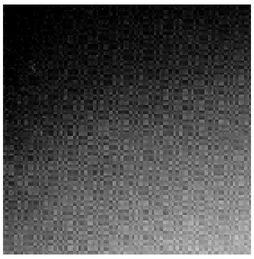
resized, nearest neighbor

resized, 11-point filter

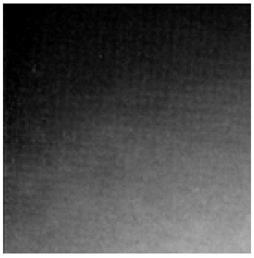
Shrinking



original



resized, nearest neighbor



resized, 11-point filter

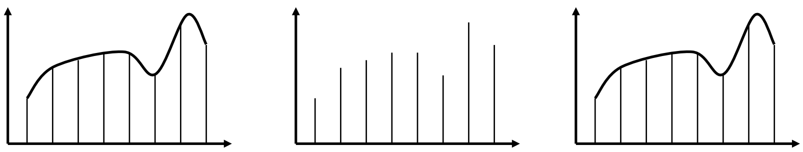
Frequency analysis

The key to fighting aliasing is to

avoid frequencies we cannot represent

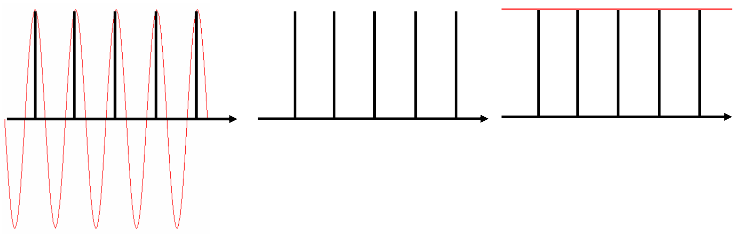
Big question:

When can we reconstruct a continuous signal from samples?



original samples reconstructed

Frequency analysis



original samples reconstructed

Need enough samples, to be more precise, sampling frequency should be twice the frequency of the wave

---

Frequency analysis

- decompose everything into waves
- mathematically convenient to use complex waves

$$A\cos(\omega t + \varphi)$$

amplitude

frequency

phase

$$A e^{i(\omega t+\varphi)} = A(\cos(\omega t + \varphi) + i\sin(\omega t + \varphi))$$

Fourier transform

Decomposes into freq. components

$$\mathrm{FT}[x](\omega) = \int_{-\infty}^{\infty} x(t)e^{-i\omega t}\,dt$$

Inverse transform reconstructs the function

$$\mathrm{IFT}[X](t) = \frac{1}{2\pi}\int_{-\infty}^{\infty} X(\omega)e^{i\omega t}\,d\omega$$

Examples of FT pairs

FT(delta function) = constant or wave

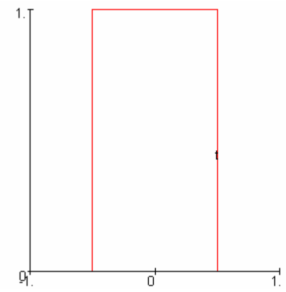
FT(constant or wave) = delta function

delta function (not quite a function):

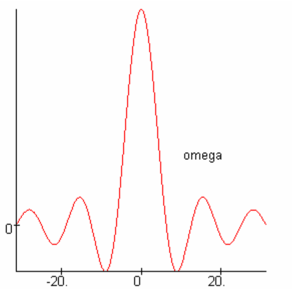
$$\text{for all } x \neq 0,\ \delta(x) = 0$$

$$\text{for all } \varepsilon > 0,\ \int_{-\varepsilon}^{+\varepsilon} \delta(x)dx = 1$$

Examples of FT pairs

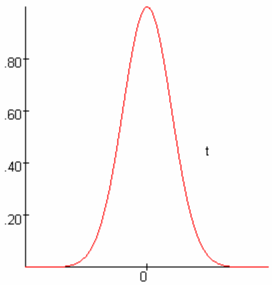


box

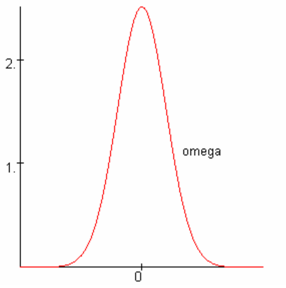


sinc

Examples of FT pairs



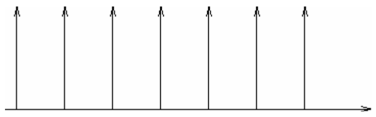
Gaussian



Gaussian

Comb function

$$\text{comb}(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT)$$



FT of a comb is also a comb

Shannon theorem

(Rough idea, not mathematically precise formulation)

Almost any function or signal can be decomposed into a sum of sine waves with different frequencies:

$$f(t) = \sum_j a_j \exp i(\omega_j t + \varphi_j)$$

amplitude

frequency

phase

A function can be reconstructed from samples if the sampling frequency is at least twice the max. frequency in the FT of the function.

General idea

Keep the part of the signal (image) we can represent (low frequencies).

Eliminate the part of the signal (image) we cannot represent (high frequencies).

Smooth slowly changing signals have lower high frequency components, smoothing eliminates high frequencies,

---

Image transformation

continuous original x(t)

sampling

often no control
(done by scanner, camera, etc.)

discrete x[n]

Can choose

discrete transformed y[n]

reconstruction

no control (display + eye)

Continuous reconstructed y(t)

Convolution

Fourier transform of a product of two functions x(t) and y(t) is a convolution of X(ω)=FT[x](ω) and Y(ω)=FT[y](ω), denoted X(ω)* Y(ω). Conversely, Inverse Fourier transform of a product of X(ω) Y(ω) is the convolution x(t)*y(t).

Convolution is defined as follows:

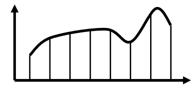
$$x(t) * y(t) = \int_{-\infty}^{\infty} x(s)y(t-s)ds$$

Convolution can be thought of as local averging

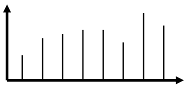
of x using y as a filter.

Sampling and reconstruction

Sampling = multiplication by comb



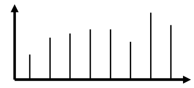
original



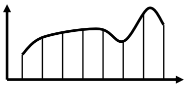
samples

$$\hat{x}(t) = \text{sinc}(t)x(t)$$

Reconstruction (ideal) = convolution with sinc



samples



reconstructed

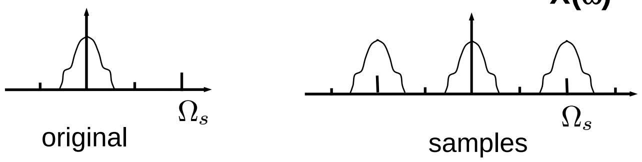
$$x(t) = \text{sinc}(t) * \hat{x}(t)$$

Samping and reconstruction

Frequency domain

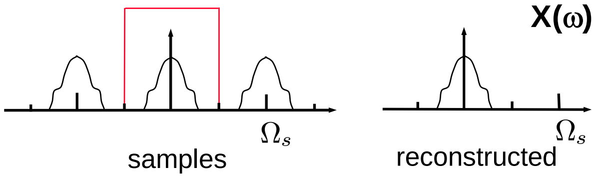
Sampling = convolution with comb = replication and shift

$$\hat{X}(\omega) = \sum_n X(\omega - n\Omega_s)$$



Reconstruction = multiplication by box (get rid of extra copies)

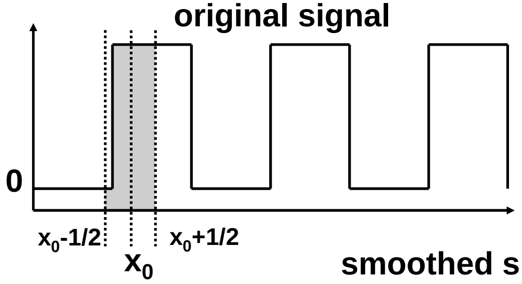
$$X(\omega) = \sum_n box(\omega)\hat{X}(\omega)$$



Continuous smoothing

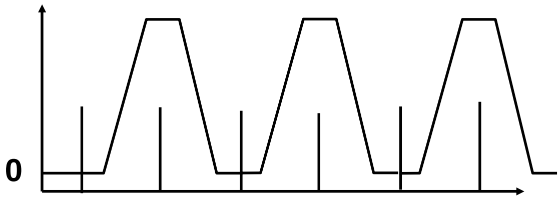
Smoothing can be done by local integration:

original signal



slide the integration interval along the x-axis

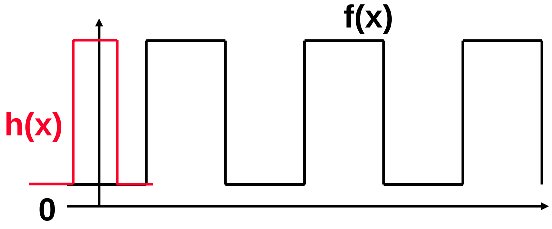
smoothed signal



$$f^{smoothed}(x_0) = \int_{x_0-1/2}^{x_0+1/2} f(x)dx$$

Continuous smoothing

Another way to look at the same integral:



Box function:

$$h(x) = \begin{cases} 1, & \text{if } -1/2 \le x \le 1/2 \\ 0, & \text{otherwise} \end{cases}$$

$$f^{\text{smoothed}}(x_0) = \int_{x_0-1/2}^{x_0+1/2} f(x)dx = \int_{-\infty}^{\infty} h(x_0 - x)f(x)dx$$

This is convolution with h(x)!
Can replace box with some other bump-like function h(x), resulting in better smoothing.

Filter quality

Why some filters h(t) are better than others?

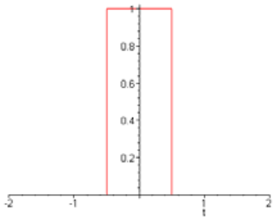
Recall that we want to eliminate high frequency components, and keep low-frequency components.

We can evaluate the filter by looking what it does to sine waves of different frequencies (frequency response). For an ideal filter,

$$\int_{-\infty}^{\infty} h(x_0 - x)\sin(\omega x)dx$$

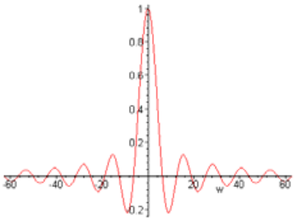
should be zero for frequencies > sampling freq./2
and 1 for frequencies < sampling freq./2

Box filter response



box(t)

$$\int_{-\infty}^{\infty} \mathrm{box}(x_0 - x)\sin \omega x dx = \frac{\sin \omega/2}{\omega/2} \sin \omega x_0$$



We have got the same wave back, but with different amplitude. The higher the frequency, the lower the amplitude. But the amplitude decreases as 1/ω, which is not very fast.

Ideal filter

A function called sinc has the ideal response, that is, for low frequencies 1 for high frequencies zero:

$$\mathrm{sinc}(x) = \frac{\sin \pi x}{\pi x}$$

$$\int_{-\infty}^{\infty} \mathrm{sinc}(x_0 - x)\sin \omega x dx = \sin \omega x_0 \quad \text{if } \omega < \pi$$

$$\int_{-\infty}^{\infty} \mathrm{sinc}(x_0 - x)\sin \omega x dx = 0 \qquad \text{if } \omega > \pi$$

Why not use it? It is not zero anywhere, except a few points -- we would have to do infinite summations!