Lecture 7

Motion filters
Sampling
Video Sequences
Sequences

(time)

\[ m \]

\[ n \]

\[ t \]
Sequences

Cube size = 128x128x90
Sequences

Cube size = 128x128x90
Globally constant motion

Let’s work on the continuous space-time domain…
A global motion can be written as:

\[ f(x, y, t) = f_0(x - v_x t, y - v_y t) \]

Where:

\[ f_0(x, y) = f(x, y, 0) \]
\[ f(x, y, t) = f_0(x - v_x t, y - v_y t) \]

\[ F(w_x, w_y, w_t) = F_0(w_x, w_y) \delta(w_t + v_x w_x + v_y w_y) \]
Temporal Gaussian

\[ g(x, y, t; \sigma_x, \sigma_t) = \frac{1}{(2\pi)^{3/2}\sigma_x^2\sigma_t} \exp\left(-\frac{x^2 + y^2}{2\sigma_x^2}\right) \exp\left(-\frac{t^2}{2\sigma_t^2}\right) \]
Spatio-temporal Gaussian
Spatio-temporal Gaussian

How could we create a filter that keeps sharp objects that move at some velocity \((v_x, v_y)\) while blurring the rest?

(Note: although some of the analysis is done on continuous variables, the processing is done on the discrete domain)
Spatio-temporal Gaussian

How could we create a filter that keeps sharp objects that move at some velocity \((v_x, v_y)\) while blurring the rest?

\[ g_{v_x,v_y}(x, y, t) = g(x - v_x t, y - v_y t, t) \]
derivatives of Gaussians
derivatives of Gaussians
Space-time Gaussian derivatives

\[ \frac{\partial g}{\partial t} = \frac{-t}{\sigma_t^2} g(x, y, t) \]

\[ \nabla g = (g_x(x, y, t), g_y(x, y, t), g_t(x, y, t)) = \]

\[ = \left( \frac{-x}{\sigma^2}, \frac{-y}{\sigma^2}, \frac{-t}{\sigma_t^2} \right) g(x, y, t) \]

**Note:** we can discretize time derivatives in the same way we discretized spatial derivatives. For instance:

\[ f[m, n, t] - f[m, n, t - 1] \]
Cancelling moving objects

Can we create a filter that *removes* objects that move at some velocity \((v_x, v_y)\) while keeping the rest?
Space-time Gaussian derivatives

For a global translation, we can write:

\[ f(x, y, t) = f_0(x - v_x t, y - v_y t) \]

Therefore, we can write the temporal derivative of \( f \) as a function of the spatial derivatives of \( f_0 \):

\[
\frac{\partial f}{\partial t} = \frac{\partial f_0}{\partial t} = -v_x \frac{\partial f_0}{\partial x} - v_y \frac{\partial f_0}{\partial y}
\]

And from here (using derivatives of \( f \)):

\[
\frac{\partial f}{\partial t} + v_x \frac{\partial f}{\partial x} + v_y \frac{\partial f}{\partial y} = 0
\]

This relation is known as the “Brightness change constraint equation”, introduced by Horn & Schunck in 1981.
Space-time Gaussian derivatives

Can could we create a filter that removes objects that move at some velocity \((v_x, v_y)\) while keeping the rest?

Yes, we could create a filter that implements this constraint:

\[
\frac{\partial f}{\partial t} + v_x \frac{\partial f}{\partial x} + v_y \frac{\partial f}{\partial y} = 0
\]

We can create this filter as a combination of Gaussian derivatives:

\[
h(x, y, t; v_x, v_y) = g_t + v_x g_x + v_y g_y
\]

\[
= \nabla g (1, v_x, v_y)^T
\]
Space-time Gaussian derivatives

Nulling-out $v_x = 0, v_y = 0$ motion
Nulling-out $v_x = -1, v_y = 0$ motion
Nulling-out $v_x = 1, v_y = 0$ motion
Fourier phase, and motion as phase changes
Gabor wavelets

\[ \psi_c(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \cos(2\pi u_0 x) \]

\[ \psi_s(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \sin(2\pi u_0 x) \]
Gabor filters at different scales and spatial frequencies

Top row shows anti-symmetric (or odd) filters; these are good for detecting odd-phase structures like edges. Bottom row shows the symmetric (or even) filters, good for detecting line phase contours.
Fourier transform of a Gabor wavelet

\[ \psi_c(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \cos(2\pi u_0 x) \]
Quadrature pair

\[ \psi_c(x,y) = e^{\frac{x^2 + y^2}{2\sigma^2}} \cos(2\pi u_0 x) \]

\[ \psi_s(x,y) = e^{\frac{x^2 + y^2}{2\sigma^2}} \sin(2\pi u_0 x) \]
“oriented energy” from a quadrature pair
Quadrature filter pairs

A quadrature filter is a complex filter whose real part is related to its imaginary part via a Hilbert transform along a particular axis through origin of the frequency domain.
edge

energy response to an edge
energy response to a line
Using phase changes of local Gabor filters to analyze or generate motion

\[
\psi_c(x,y) = e^{-\frac{x^2+y^2}{2\sigma^2}} \cos(2\pi u_0 x + \phi t)
\]
Space-time plot of the a slice through the patio-temporal filter of the previous slide

\[ \psi_c(x, y) = e^{-\frac{x^2 + y^2}{2\sigma^2}} \cos(2\pi u_0 x + \phi t) \]
Motion without movement

Figure 1: 1-d cross-sections of filters. (a) Even phase ($G_2$). (b) Odd phase ($H_2$). (c) Filters modulated in phase according to Eq. (1). Note the apparent rightward motion of the filter ripples.

Figure 2: (a) and (b): $G_2$ and $H_2$ filters were applied to an image of Einstein. (c) Images modulated as in Eq. (1). When viewed as a temporal sequence, this generates the perception of rightward motion, yet image remains stationary.
Motion without movement
original
Sampling
Sampling

Continuous world

Pixels
Sampling
Sampling
Sampling

Continuous image $f(x, y)$

We can sample it using a rectangular grid as

$$f[n, m] = f(nT_x, mT_y)$$
Let’s start with this continuous image (it is not really continuous...)
Aliasing
Modeling the sampling process

Continuous image $f(x, y)$

We can sample it using a rectangular grid as

$$f[n, m] = f(nT_x, mT_y)$$

Or a more general sampling pattern

$$f[n, m] = f(an + bm, cn + dm)$$

If $a = T, b = 0, c = 0, d = T$ then we will have a rectangular sampling
Modeling the sampling process

\[ f[n] = f(nT_s) \]

A convenient writing:

\[ \hat{f}(t) = f(t) \sum_{n=-\infty}^{\infty} \delta(t - nT_s) \]
Modeling the sampling process

The Fourier transform is a convolution…

Interesting property of the delta train: the Fourier transform of a delta train of period $T$ is another delta train with period $2\pi/T$
Interesting property of the delta train: the Fourier transform of a delta train of period $T$ is another delta train with period $2\pi/T$. Demo in the class notes.
Modeling the sampling process

What happens when the repetitions overlap?
Both waves fit the same samples. Aliasing consists in “perceiving” the red wave when the actual input was the blue wave.
The sampling theorem (also known as Nyquist theorem) states that for a signal to be perfectly reconstructed from its samples, the sampling period $T_s$ has to be $T_s > T_{\text{min}}/2$ where $T_{\text{min}}$ is the period of the highest frequency present in the input signal.
Antialising filtering

Before sampling, apply a low pass-filter to remove all the frequencies that will produce aliasing.

Without antialising filter.

With antialising filter.
Evidence for filter-based analysis of motion in the human visual system
Square wave Fourier components

Using Fourier series we can write an ideal square wave as an infinite series of the form

\[ x_{\text{square}}(t) = \frac{4}{\pi} \sum_{k=1}^{\infty} \frac{\sin((2k-1)2\pi ft)}{(2k-1)} \]

\[ = \frac{4}{\pi} \left( \sin(2\pi ft) + \frac{1}{3} \sin(6\pi ft) + \frac{1}{5} \sin(10\pi ft) + \cdots \right) \].
filters to analyze motion
Spatio-temporal aliasing

\[ \sin(2\pi ft) \]
\[ \frac{1}{3}\sin(6\pi ft) \]

space \quad time

space \quad time

spatial frequency \quad spatial frequency

temporal frequency \quad temporal frequency
Spatio-temporal aliasing

\[ \frac{1}{3} \sin(6\pi ft) \]

\[ \sin(6\pi ft) \]
end