CIS 580 Spring 2012 - Lecture 6
Filters for detection

February 6, 2012

Notes and figures by Matthieu Lecce.

In this section, we are interested in building filters to detect patterns in a signal. By pattern we mean a portion of the signal with a distinctive behavior: edge, corner, portion oscillating at a specific frequency.

Gabor functions

Gabor functions can be considered as local band-pass filters: they are designed to detect portions of the signal that oscillate at a specific frequency ω_1: namely the output of the convolution tells us how much each position of the signal looks locally like a sinusoidal wave of frequency ω_1.

Gaussian filter

As an example, let's first consider a simple Gaussian filter:

$$g(t) = \frac{1}{\sigma\sqrt{(2\pi)}} e^{-t^2/2\sigma^2} \circ\!\!-\!\!\bullet\; G(\omega) = e^{-\sigma^2\omega^2/2}$$

What is the output of this filter for a cosine input? We have:

$$f(t) = \cos(\omega_0 t) \circ\!\!-\!\!\bullet\; \frac{1}{2}(\delta(\omega-\omega_0) + \delta(\omega+\omega_0))$$

Therefore in the Fourier domain the output spectrum is (see figure 1):

$$F(\omega)G(\omega) = e^{-\sigma^2\omega_0^2/2}(\frac{1}{2}(\delta(\omega-\omega_0) + \delta(\omega+\omega_0)))$$

And the output signal is (by taking the inverse transform and noticing that $e^{-\sigma^2\omega_0^2/2}$ *acts as a constant*):

$$\cos\omega_0 t * g(t) = e^{-\sigma^2\omega_0^2/2}\cos\omega_0 t$$

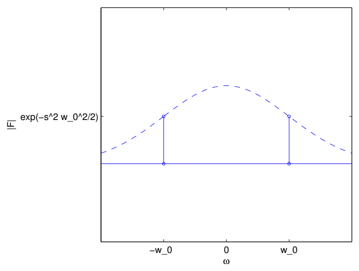

Figure 1: Spectrum of a cosine filtered by a Gaussian: $F(\omega)G(\omega)$

Gabor functions are modulated Gaussians

The Gaussian filter can be seen as a low-pass filter: it eliminates frequencies too far away from its mean, zero. How to obtain a band-pass filter instead? We can simply *modulate* the Gaussian filter, and we obtain what we call a *Gabor function*:

$$\begin{aligned} h(t) =& \frac{1}{\sigma\sqrt{2\pi}} e^{-t^2/(2\sigma^2)} e^{j\omega_1 t} \\ &\circ\!\!-\!\!\bullet \\ H(\omega) =& e^{-\sigma^2(\omega-\omega_1)^2/2} \end{aligned}$$

See figure 2 for an example of Gabor filter (real and imaginary parts). What is the output of this modulated filter for $\cos(\omega_0 t)$? In the Fourier domain we obtain (see figure 3):

$$\frac{1}{2}\left(e^{-\sigma^2(\omega_1+\omega_0)^2/2}\delta(\omega+\omega_0)+e^{-\sigma^2(\omega_1-\omega_0)^2/2}\delta(\omega-\omega_0)\right)$$

When the Gaussian is relatively narrow in the frequency domain, the small peak at $-\omega_0$ in figure 3 can be neglected, and the output is essentially a complex exponential of frequency ω_0, and of magnitude $e^{\sigma^2(\omega_1-\omega_0)^2}$. The good news is that as ω_0 gets away from ω_1, the magnitude decreases, producing the desired band-pass effect. The bad news is that the real and imaginary part of the output depend on t: they are *"phase-dependent"*, which means that even when we convolve a sine wave at fixed frequency with a Gabor filter, the real/imaginary components of the output are not constant but are waves of same frequency. Because the components are in *quadrature* (the phase difference is $\pi/4$), we simply take the norm of the output and obtain a non-oscillating response, as demonstrated in figure 4.

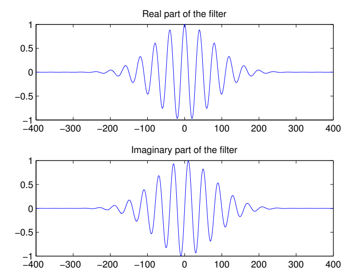

Figure 2: Example of Gabor filter: notice the exponential envelope, and the real and imaginary part are in *quadrature* (cos and sin)

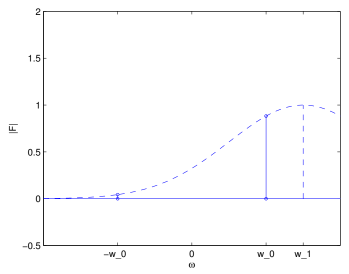

Figure 3: Spectrum of a cosine filtered by a modulated Gaussian: $F(\omega)G(\omega)$

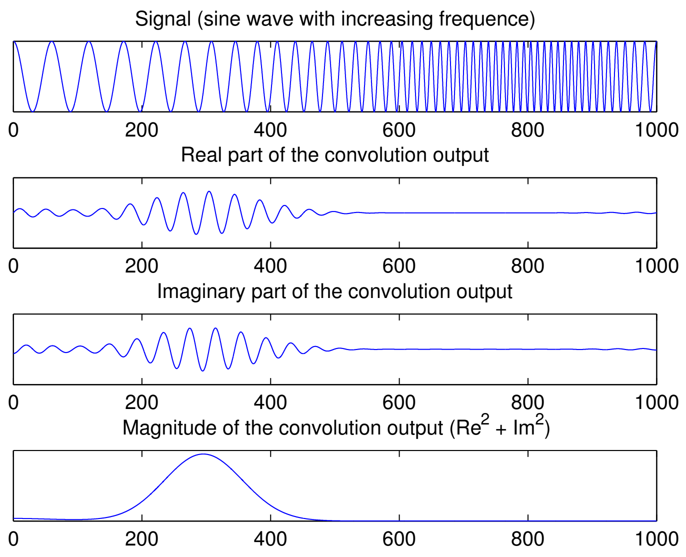

Figure 4: Convolution of our Gabor example (fig. 2) with a signal at increasing frequency: notice that the outputs have same frequency as the input, only the magnitude change: it reaches a maximum exactly when the signal is oscillating at the frequency of the filter. *All these plots are in the time domain.*

The phase-invariance of the norm of the Gabor output is important, since we want to be able to detect frequency patterns (and potentially edges and ridges) no matter what their phase is (how much they locally look like a sine or a cosine, or how they are translated).

Signal/spectrum spread: time localization VS frequency accuracy

We can use Gabor filters to localize pieces of a signal that oscillate at a certain frequency: for instance, if we were looking for the character "Waldo" in a 2D image, we could use the fact that he is wearing a distinctive red-and-white striped shirt and use a Gabor filter of appropriate frequency on the red channel of the image.

WHEN USING SUCH A FILTER FOR DETECTION, we are interested in the **spread** of a filter and its spectrum, because:

- a filter of small spread in the time/space domain enables precise localization of the pattern
- a filter whose *spectrum* has a narrow spread (think of it as a narrow band-pass filter) will enable precise detection of a specific frequency.

(for example in figure 4, the localization of the desired frequency is rough, as the filter in figure 4 has quite a large support)

The spread of a signal of finite support f is the second order moment $\int_{-\infty}^{\infty} t^2 f^2(t)dt$ and gives a measure of how wide the support is.

As we have observed in the past lectures, signals of small spread tend to have a wide spectrum.

This is formalized by a result named the **"uncertainty principle of signal processing"**:

- For a signal/filter of spread $\Delta t^2 = \int_{-\infty}^{\infty} t^2 f^2(t)dt$...
- ...whose spectrum has spread $\Delta\omega^2 = \int_{-\infty}^{\infty} \omega^2 F^2(\omega)d\omega$...
- ... we have $\Delta t \Delta\omega \geq$ lower bound.

2D Gabor function

We can derive a 2D version of the Gabor function to detect oscillating patters in images, at a given frequency ω_1 and *orientation* θ. Let's start from the **horizontal** version of the filter:

$$h(x, y, \sigma_1, \sigma_2, \omega_1) = \frac{1}{\sigma_1\sigma_2 2\pi} e^{-(x^2/2\sigma_1^2 + y^2/2\sigma_2^2)} e^{j\omega_1 x}$$

Notice that the Gaussian envelope is 2D, but the harmonic exponential is along one direction (x-xaxis here). If we want to detect oscillations at a different orrientation, we need a *rotated Gabor*:

$$h(x, y, \sigma_1, \sigma_2, \omega_1, \theta) = \frac{1}{\sigma_1\sigma_2 2\pi} e^{-1/2(x\ y)R^T()R(x\ y)^T} e^{j\omega_1(x\cos(\theta) - y\sin(\theta))}$$

Let's compute the Fourier of the 2D Gabor function:

$$e^{-(x^2/2\sigma_1^2 + y^2/2\sigma_2^2)} \circ\!\!-\!\!\bullet\ e^{-(\sigma_1^2\omega_x^2 + \sigma_2^2\omega_y^2)/2}$$

$$e^{-(x^2/2\sigma_1^2 + y^2/2\sigma_2^2)} e^{j\omega_1 x} \circ\!\!-\!\!\bullet\ e^{-(\sigma_1^2(\omega_1 - \omega_x)^2 + \sigma_2^2\omega_y^2)/2}$$

Now for the rotated version, notice that:

$$\frac{x^2}{2\sigma_1^2} + \frac{y^2}{2\sigma_2^2} = \frac{1}{2}\begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} 1/\sigma_1^2 & 0 \\ 0 & 1/\sigma_2^2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix},$$

and we apply a rotation R_θ to x and y [1]:

$$\begin{bmatrix} x' & y' \end{bmatrix} \begin{bmatrix} 1/\sigma_1^2 & 0 \\ 0 & 1/\sigma_2^2 \end{bmatrix} \begin{bmatrix} x' \\ y' \end{bmatrix}$$

$$= \begin{bmatrix} x & y \end{bmatrix} R_\theta \begin{bmatrix} 1/\sigma_1^2 & 0 \\ 0 & 1/\sigma_2^2 \end{bmatrix} R_\theta^T \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} 1/\sigma_1^2 & 0 \\ 0 & 1/\sigma_2^2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

[1] $\begin{bmatrix} x \\ y \end{bmatrix} = R_\theta \begin{bmatrix} x' \\ y' \end{bmatrix}$ where $R_\theta = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$)

Since the diagonal matrix commutes and $R_\theta R_\theta^T = I$. Intuitively the Fourier of the rotation is the original Fourier, rotated of the same amount. Indeed, remember this result from last lecture:

$$h\left(R\begin{bmatrix} x \\ y \end{bmatrix}\right) \circ\!\!-\!\!\bullet\ H\left(R\begin{bmatrix} \omega_x \\ \omega_y \end{bmatrix}\right)$$

Edge detection with Gaussian derivative filters

We now look at a different set of filters: Gaussian derivatives. They can also be seen as band-pass filters, but here instead of modulating a Gaussian, we look at its derivatives. Intuitively, because of the derivation rule for convolution, convolving an image with a Gaussian derivative is the same as smoothing the image (applying a Gaussian filter) and then looking at the derivatives of the output.

1D edge

An ideal edge is a step function $u(t)$. Instead of detecting the ideal edge, we can detect a smoothed edge. We have the following result from HW1:

$$\frac{d}{dt}(u(t) * g(t)) = g(t)$$

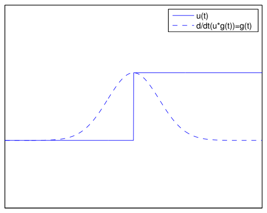

Figure 5: Edge detection with a Gaussian filter: $\frac{d}{dt}(u(t) * g(t)) = g(t)$.

Therefore the location of the edge can be given by $\max_t \frac{d}{dt}(f * g)(t)$, as shown in figure 5.

2D edge detection

The 2D version can be see as an image intensity that goes from a low to a high along a direction θ.

The isotropic Gaussian is the following:

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

We have:

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

Now again, when we apply a rotation of angle $\theta > 0$

The function $g_x(R^T(x;y))$ takes the following very simple form:

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

$$g_x(R^T(x;y)) = \cos\theta g_x + \sin\theta g_y = (\cos\theta \ \sin\theta)\nabla g.$$

To compute the derivative of an image in direction θ, all we need to do is the following:

$$\cos\theta(g_x * I) + \sin\theta(g_y * I)$$

This property is called **steerability**: we will cover it more extensively in the next lecture.

Recall that the 1D step edge can be localized as the maximum of $\frac{d}{dt}(u(t) * g(t))$.

In the 2D case, there is an edge at (x,y) if the first derivative of the convolution with the Gaussian has a maximum in the direction of the gradient.

Remark: the maximum is a non-linear operation. Is there a linear way to find edges? Yes, by taking the derivative of the above quantity: $\frac{d^2}{dt^2}(u(t) * g(t))$. Therefore an edge corresponds to a **zero-crossing of the second derivative of the smoothed signal.**