Today we will look more closely at an important issue – scale. Scale has come up indirectly in our discussions. For example, in the Canny edge analysis, we considered what happens when we stretch the size of a filter. More recently, when we discussed image registration and Harris corners, we discussed image distances (length scales) over which we can or cannot assume a first order model of intensity. Today we will look more closely at scale issues.

Gaussian scale space

We begin with a 1D image $I(x)$. Define family of images

$$I(x,\sigma) = I(x) * G_\sigma(x).$$

The function $I(x,\sigma)$ is called a Gaussian *scale space*. You can think of it as a sequence of blurred 1D images, indexed by the amount of blur σ which is the standard deviation of the Gaussian.

Note that the Gaussian has the property that

$$G_\sigma(x) = sG_{s\sigma}(sx) \tag{1}$$

i.e.

$$\frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \;=\; \frac{s}{\sqrt{2\pi}s\sigma} e^{-\frac{s^2x^2}{2s^2\sigma^2}}$$

For example, if we squeeze the Gaussian by some factor then the peak of the Gaussian is multiplied by the same factor. Remember, the Gaussian is a probability function and so it has area 1, and so if we squeeze it in the horizontal direction then we must expand it in the vertical direction to preserve area.

Now suppose we have a second image $J(x)$ such that $J(x) = I(sx)$. How is the scale space for $J(x)$ related to the scale space for $I(x)$?

$$\begin{aligned}
J(x) * G_\sigma(x) &= \int G_\sigma(x-x')J(x')dx' \\
&= \int G_\sigma(x-x')I(sx')dx' \\
&= \int sG_{s\sigma}(s(x-x'))I(sx')dx', \quad \text{from Eq. (1)} \\
&= \int G_{s\sigma}(sx-sx')I(sx')d(sx') \\
&= \int G_{s\sigma}(sx-w)I(w)d(w), \quad \text{where } w = sx' \\
&= (I * G_{sx})(sx) \qquad\qquad (*)
\end{aligned}$$

The scale space for $J(x)$ is squeezed by a factor s relative to the scale space for $I(x)$ and this squeeze occurs for both dimensions x, σ.

Similar arguments hold in 2D. For a 2D Gaussian, you can verify that

$$G_\sigma(x,y) = s^2 G_{s\sigma}(sx,sy).$$

Go through the steps in the above derivation for 2D. You will see that if $I(x,y) = J(sx, sy)$ then

$$(J * G_\sigma)(x,y) = (I * G_{s\sigma})(sx, sy).$$

This says that if you have a second image $J()$ that is a squeezed version of $I()$ and if you blur $J()$ with some Gaussian, then the resulting image is the same as if you first blur $I()$ with a stretched Gaussian and then squeeze it.

Edge detection and normalized Gaussian derivatives

Consider a noise-free image edge $I(x) = u(x - x_0)$. To detect edges previously, we convolved them with a derivative of Gaussian and then looked for a peak in the response. Suppose we define a scale space by convolving $I(x)$ with a family of first derivative of Gaussian filters:

$$I(x) * \frac{dG_\sigma}{dx}(x) = G_\sigma(x) * \frac{dI}{dx}(x) = G_\sigma(x) * \delta(x - x_0) = G_\sigma(x - x_0).$$

At the location of the edge $x = x_0$, we have

$$(I * \frac{dG_\sigma}{dx})(x_0) = G_\sigma(0) = \frac{1}{\sqrt{2\pi}\sigma}$$

which depends on σ.

We can get a response that does *not* depend on σ by using a slightly different filter, namely $\sigma \frac{d}{dx} G_\sigma(x)$. Then $(I * \sigma \frac{dG_\sigma}{dx})(x)$ will take value $\frac{1}{\sqrt{2\pi}}$ at $x = x_0$ which obviously does not depend on σ. We refer to this filter as *normalized Gaussian derivative* and the resulting scale space as the *normalized Gaussian derivative* scale space.[1]

If we have a 2D image, then we define the normalized derivative filter in the same way, namely $\sigma \frac{\partial G_\sigma}{\partial x}(x,y)$ and similarly for y. Exactly the same arguments as above are used to show that the value at a horizontal or vertical edge will be independent of σ. Using these filters, one defines the normalized gradient scale space in the obvious way, and one will find that the gradient at an edge of arbitrary orientation will be independent of σ. That is, we construct a scale space of image gradients

$$\nabla I(x,y) = (\sigma \frac{\partial G_\sigma}{\partial x}, \sigma \frac{\partial G_\sigma}{\partial y}) * I(x,y)$$

If we have an edge in the image, namely a line across which there is a change in intensity then (because we are using normalized derivatives), we expect that *at the location of the edge*, the magnitude of the gradient ∇I will be independent of the scale σ. Note: if you are at a point *near* the edge, but *not on* the edge, then the magnitude of the gradient *will* depend on sigma.

Next, let's go back to the 1D case and reconsider our two images $I()$ and $J()$ which are related by $I(sx) = J(x)$. We saw on p. 1 that if make a Gaussian scale space of each image, then these scale spaces will be identical except for a scaling of the x and σ dimensions. This property holds for the scale space defined by a derivative of a Gaussian too, provided we used the normalized derivative. That is, if $I(sx) = J(x)$ then

$$(J * \sigma \frac{dG_\sigma}{dx})(x) = (I * s\sigma \frac{dG_{s\sigma}}{dx})(sx).$$

[1] The idea of using normalized derivative scale spaces is generally attributed to T. Lindeberg, e.g. "Feature detection with automatic scale selection," International Journal of Computer Vision, 1998.

Why? First, here is a quick derivation. (This was not presented in class, but rather was suggested to me by a student after class):

$$\begin{aligned} J(x) * \sigma \frac{dG_\sigma(x)}{dx} &= \sigma \frac{d}{dx}(J(x) * G_\sigma(x)) \\ &= \sigma \frac{d}{dx}(I(sx) * sG_{s\sigma}(sx)), \quad \text{by substituting from Eq. 1} \\ &= I(sx) * s\sigma \frac{dG_{s\sigma}}{dx}(sx). \end{aligned}$$

This derivation is correct, as long as you *interpret* it correctly. When you push the derivative in on the last line, you must interpet $\frac{dG_{s\sigma}}{dx}(sx)$ to be the derivative of function $G_{s\sigma}(x)$, evaluated on the scaled domain sx instead of x. You cannot interpret it to mean $\frac{dG_{s\sigma}(sx)}{dx}$ which is the derivative of $G_{s\sigma}(sx)$ with respect to x. The two are not the same, namely the latter is s times the former.

Just to be sure we understand, let's a closer look at the step from the second to third lines of the quick derivation. It just uses the commutativity and associativity of convolution and says :

$$D(x) * (J(x) * g(x)) = J(x) * (D(x) * g(x))$$

where we have ignored the constants $s\sigma$ and we are defining:

$$f(x) \equiv I(sx)$$

$$g(x) \equiv G_{s\sigma}(sx)$$

$$D(x) \equiv lim_{\epsilon \to 0} \ \frac{1}{2\epsilon}(\delta(x+\epsilon) - \delta(x-\epsilon)).$$

With these definitions, the derivative must be interpreted as:

$$\begin{aligned} D(x) * g(x) &= lim_{\epsilon \to 0} \ (\delta(x+\epsilon) - \delta(x-\epsilon)) * g(x) \\ &= lim_{\epsilon \to 0} \ \frac{1}{2\epsilon} \ (G_{s\sigma}(sx+\epsilon) - G_{s\sigma}(sx-\epsilon) \\ &= \frac{dG_{s\sigma}}{dx}(sx). \end{aligned}$$

Finally, for completeness, here is the derivation that I did give in class:

$$\begin{aligned} J(x) * \sigma \frac{dG_\sigma}{dx}(x) &= \sigma \frac{d}{dx}(J(x) * G_\sigma(x)) \\ &= \sigma \frac{d}{dx} \int J(x')G_\sigma(x-x')dx', \\ &= \sigma \frac{d}{dx} \int I(sx')sG_{s\sigma}(sx-sx')dx', \quad \text{from Eq. (1)} \\ &= \sigma \int I(w) \frac{dG_{s\sigma}(sx-w)}{dx} dw, \qquad w = sx', dw = sdx' \\ &= \sigma \int I(w) \ \frac{dG_{s\sigma}(sx-w)}{d(sx-w)} \frac{d(sx-w)}{dx} dw \\ &= \sigma \int I(w) \ s \ \frac{dG_{s\sigma}}{dx}(sx-w)dw \qquad\qquad (**) \\ &= (I * \ s\sigma \ \frac{dG_{s\sigma}}{dx})(sx) \end{aligned}$$

where by $\frac{dG_{s\sigma}}{dx}(sx-w)$ in (**), we mean the derivative of $G_{s\sigma}(x)$, but with domain remapped $x \rightarrow sx - w$.

Exactly the same derivations are applied in 2D when $I(sx, sy) = J(x, y)$, yielding

$$(J * \sigma \frac{\partial G_\sigma}{\partial x})(x,y) = (I * s\sigma \frac{\partial G_{s\sigma}}{\partial x})(sx, sy)$$

$$(J * \sigma \frac{\partial G_\sigma}{\partial y})(x,y) = (I * s\sigma \frac{\partial G_{s\sigma}}{\partial y})(sx, sy).$$

In a nutshell, this says that if we scale an image and take normalized derivatives, then the scale spaces are identical, except that the *domains* (x, y, σ) are scaled (stretched or squeezed).

The above "scale invariance" property of Gaussian derivatives is appealing, since now the height of the peak response to the edge depends only on the amplitude A of the edge. (We have been assuming this lecture that the edge amplitude is 1.) Note that Canny's analysis is unaffected if we used normalized Gaussian derivatives since his S:N and localization analysis is at a single scale only and doesn't change if we multiply the filter by some number i.e. $f(x) \rightarrow \sigma f(x)$.

Normalized second derivative of Gaussian

We know that if we filter a (noise-free) edge, $I(x) = u(x - x_0)$, with a first derivative of a Gaussian then we get a peak response at the location of the edge. It follows immediately that if we were to filter the edge with a *second* derivative of a Gaussian

$$I(x) * \frac{d^2 G_\sigma(x)}{dx^2}$$

then the response would be zero at the location of the edge.

The second derivative of a Gaussian filter, and its 2D equivalent, have been very important in computer vision as well as in human vision modelling, and was the basis for an influential early theory of edge detection[2]. The response of this filter to the edge image is:

$$\begin{aligned} u(x-x_0) * \frac{d^2 G_\sigma(x)}{dx^2} &= \frac{d}{dx}(u(x-x_0) * \frac{dG_\sigma(x)}{dx}) \\ &= \frac{dG_\sigma(x-x_0)}{dx} \\ &= \frac{-(x-x_0)}{\sqrt{2\pi}\sigma^3} e^{\frac{-(x-x_0)^2}{2\sigma^2}} \end{aligned}$$

and note that the response is indeed 0 when $x = x_0$, as we expect.

Where do the peaks occur? Taking the derivative we get

$$\frac{d}{dx} \ (\ u(x-x_0) * \frac{d^2 G_\sigma(x)}{dx^2} \ ) \ = -(1 - \frac{(x-x_0)^2}{\sigma^2}) \frac{1}{\sqrt{2\pi}\sigma^3} e^{\frac{-(x-x_0)^2}{2\sigma^2}}$$

[2]..which attempted to model image processing in the eye and brain. See D. Marr, E. Hildreth, "Theory of Edge Detection", Proc. Royal Soc. Lond. (Series B), 1980.

and set it to 0. The peak thus occurs when

$$1 - \frac{(x - x_0)^2}{\sigma^2} = 0$$

that is, $x = x_0 \pm \sigma$. Substituting, we see that the value of the peak is $\pm \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-1}{2}}$.

As we did with the first derivative filter, we can normalize the second derivative filter by multiplying by σ^2, so the *normalized second derivative* filter is defined $\sigma^2 \frac{d^2 G_\sigma}{dx^2}(x)$. This cancels the σ^{-2} dependence in the height of the peaks. Thus we see that if we filter an edge with a normalized second derivative of a Gaussian (as defined above), then there is a zero-crossing at the location of the edge and there are peaks (positive and negative) at a distance $\pm\sigma$ away from the edge, but the height of the peaks doesn't depend on σ. Let's now apply this normalized second derivative to a problem we have not seen before.

Blob detection

Suppose we have a 1D intensity function that has value 1 between x_0 and x_1 and zero otherwise. We can define such a pattern as the difference of step edges, namely

$$I(x) = u(x - x_0) - u(x - x_1).$$

We refer to it as a 1D "blob".

If we filter the blob with the normalized second derivative of a Gaussian, then the scale space has a very nice property. When $\sigma = \frac{x_1 - x_0}{2}$, the peaks in the response to the filter $\sigma^2 \frac{d^2 G_\sigma}{dx^2}(x)$ will coincide at $x = x_0 + \sigma = x_1 - \sigma = \frac{x_0 + x_1}{2}$, and the two peaks reinforce each other rather than cancelling. The result is we get a large minimum in scale space at the center of the blob. (See figure below. Matlab code for generating this figure is online next to PDF for these notes.) Note that the peak value in scale space depends only on the amplitude A of the blob, which we are assuming is 1 in this example. The scale space peak occurs at the center of the blob (in x) and at the half width of the blob (in σ).

If we did not use a normalized second derivative of Gaussian, then we would not have this nice correspondence between the width of the blob and the scale where the peak occurs.

The 2D case is similar. One uses the normalized *Laplacian*[3] filter,

$$\nabla^2_{norm} G_\sigma(x, y) \equiv \sigma^2 \left(\frac{\partial^2 G_\sigma}{\partial x^2}(x, y) + \frac{\partial^2 G_\sigma}{\partial y^2}(x, y) \right)$$

and defines the scale space $\nabla^2_{norm} G_\sigma(x, y) * I(x, y)$. In particular, if $I(x, y)$ is a 2D square blob, namely $I(x, y)$ is constant in $(x, y) \in [x_0, x_0 + 2\sigma_0] \times [y_0, \sigma + 2\sigma_0]$ and zero outside this square, then one gets a peak in the scale space at $(x, y, \sigma) = (x_0 + \sigma_0, y_0 + \sigma_0, \sigma_0)$.

[3] The "Laplacian operator" is defined to be the sum $\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$

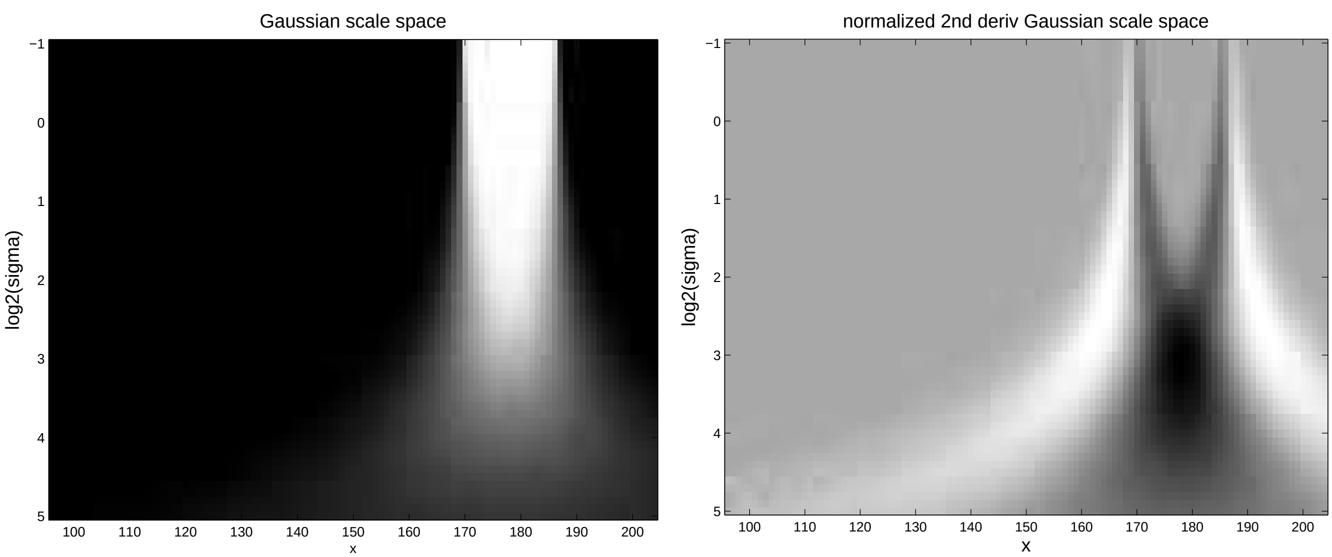

Figure 1: Blob from $x_0 = 170$ to $x_1 = 186$. Peak in scale space occurs at $\sigma = 8$ or $\log_2 \sigma = 3$