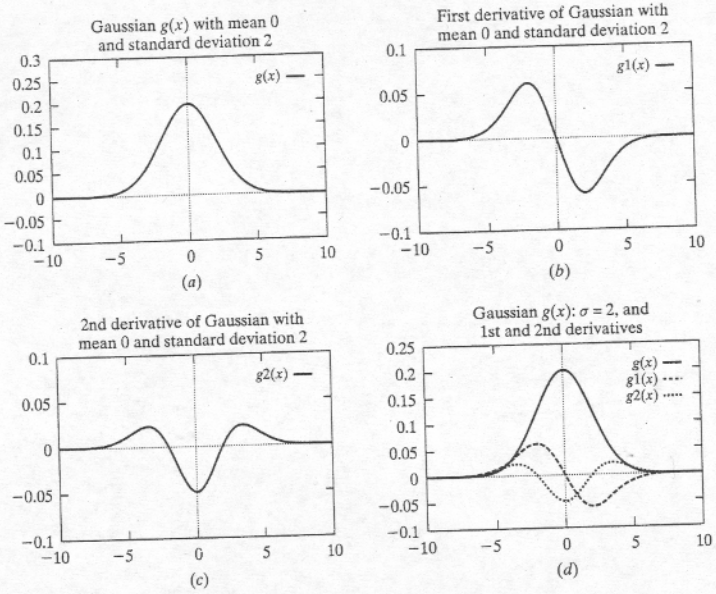


Figure 5.19 (*a*) Gaussian $g(x)$ with spread $\sigma = 2$; (*b*) first derivative $g'(x)$; (*c*) second derivative $g''(x)$, which looks like the cross section of a sombrero upside down from how it would be worn; (*d*) all three plots superimposed to show how the extreme slopes of $g(x)$ align with the extremas of $g'(x)$ and the zero crossings of $g''(x)$.

$x = \pm\sigma$, in agreement with the plots in Figure 5.19.

$$g(x) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{x^2}{2\sigma^2}} \tag{5.17}$$

$$g'(x) = \frac{-1}{\sqrt{2\pi}\,\sigma^3} x e^{-\frac{x^2}{2\sigma^2}} \tag{5.18}$$

$$= \frac{-x}{\sigma^2}\, g(x) \tag{5.19}$$

$$g''(x) = \left(\frac{x^2}{\sqrt{2\pi}\,\sigma^5} - \frac{1}{\sqrt{2\pi}\,\sigma^3}\right) e^{-\frac{x^2}{2\sigma^2}} \tag{5.20}$$

$$= \frac{x^2}{\sigma^4}\, g(x) - \frac{1}{\sigma^2}\, g(x) \tag{5.21}$$

$$= \left(\frac{x^2}{\sigma^4} - \frac{1}{\sigma^2}\right) g(x) \tag{5.22}$$

Understanding the properties of the 1D Gaussian, we can now intuitively create the corresponding 2D function $g(x, y)$ and its derivatives by making the substitution

$r = \sqrt{x^2 + y^2}$. This creates the 2D forms by just spinning the 1D form about the vertical axis yielding *isotropic* functions which have the same 1D Gaussian cross section in any cut through the origin. The second derivative form is well known as a sombrero or Mexican hat. From the mathematical derivations, the cavity for the head will be pointed upward along the $z = g(x, y)$ axis; however, it is usually displayed and used in filtering with the cavity pointed downward, or equivalently, with the center lobe positive and the rim negative.

Some Useful Properties of Gaussians

1. Weight decreases smoothly to zero with distance from the origin, meaning that image values nearer the central location are more important than values that are more remote; moreover, the spread parameter σ determines how broad or focused the neighborhood will be. 95 percent of the total weight will be contained within 2σ of the center.
2. Symmetry about the abscissa; flipping the function for convolution produces the same kernel.
3. Fourier transformation into the frequency domain produces another Gaussian form, which means convolution with a Gaussian mask in the spatial domain reduces high frequency image trends smoothly as spatial frequency increases.
4. The second derivative of a 1D Gaussian $g''(x)$ has a smooth center lobe of negative area and two smooth side lobes of positive area: the zero crossings are located at $-\sigma$ and $+\sigma$, corresponding to the inflection points of $g(x)$ and the extreme points of $g'(x)$.
5. A second derivative filter based on the Laplacian of the Gaussian is called a **LOG filter.** A LOG filter can be approximated nicely by taking the difference of two Gaussians $g''(x) \approx c_1 e^{-\frac{x^2}{2\sigma_1^2}} - c_2 e^{-\frac{x^2}{2\sigma_2^2}}$, which is often called a **DOG** filter (for **D**ifference **O**f **G**aussians). For a positive center lobe, we must have $\sigma_1 < \sigma_2$; also, σ_2 must be carefully related to σ_1 to obtain the correct location of zero crossings and so that the total negative weight balances the total positive weight.
6. The LOG filter responds well to intensity differences of two kinds—small blobs coinciding with the center lobe, and large step edges very close to the center lobe.

Two different masks for Gaussian smoothing are shown in Figure 5.20. Masks for edge detection are given below.

5.7.1 Detecting Edges with the LOG Filter

Two different masks implementating the LOG filter are given in Figures 5.21 and 5.22. The first is a 3×3 mask: The smallest possible implementation detects image details nearly the

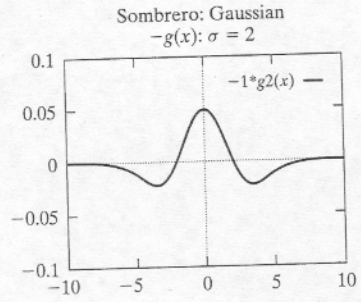
$G_{3\times3}$ =

1	2	1
2	4	2
1	2	1

; $G_{7\times7}$ =

1	3	7	9	7	3	1
3	12	26	33	26	12	3
7	26	55	70	55	26	7
9	33	70	90	70	33	9
7	26	55	70	55	26	7
3	12	26	33	26	12	3
1	3	7	9	7	3	1

Figure 5.20 (left) A 3×3 mask approximating a Gaussian obtained by matrix multiplication $[1, 2, 1]^t \otimes [1, 2, 1]$; and (right) a 7×7 mask approximating a Gaussian with $\sigma^2 = 2$ obtained by using Equation 5.16 to generate function values for integers x and y and then setting $c = 90$ so that the smallest mask element is 1.



0	−1	0
−1	4	−1
0	−1	0

5	5	5	5	5	5
5	5	5	5	5	5
5	5	10	10	10	10
5	5	10	10	10	10
5	5	5	10	10	10
5	5	5	5	10	10

–	–	–	–	–	–
–	0	−5	−5	−5	–
–	−5	10	5	5	–
–	−5	10	0	0	–
–	0	−10	10	0	–
–	–	–	–	–	–

Figure 5.21 (top row) Cross section of the LOG filter and a 3×3 mask approximating it; and (bottom row) input image and result of applying the mask to it.

Exercise 5.10: Properties of the LOG filter.

Suppose that the 9 intensities of a 3×3 image neighborhood are perfectly fit by the planar model $I[r, c] = z = pr + qc + z_0$. (Recall that the 9 samples are equally spaced in terms of r and c.) Show that the simple LOG mask

0	−1	0
−1	4	−1
0	−1	0

has zero response on such a neighborhood. This means that the LOG filter has zero response on both constant regions and ramps.

0	0	0	−1	−1	−2	−1	−1	0	0	0
0	0	−2	−4	−8	−9	−8	−4	−2	0	0
0	−2	−7	−15	−22	−23	−22	15	−7	−2	0
−1	−4	−15	−24	−14	−1	−14	−24	−15	−4	−1
−1	−8	−22	−14	52	103	52	−14	−22	−8	−1
−2	−9	−23	−1	103	178	103	−1	−23	−9	−2
−1	−8	−22	−14	52	103	52	−14	−22	−8	−1
−1	−4	−15	−24	−14	−1	−14	−24	−15	−4	−1
0	−2	−7	−15	−22	−23	−22	15	−7	−2	0
0	0	−2	−4	−8	−9	−8	−4	−2	0	0
0	0	0	−1	−1	−2	−1	−1	0	0	0

Figure 5.22 An 11×11 mask approximating the Laplacian of a Gaussian with $\sigma^2 = 2$. (From Haralick and Shapiro, vol. I, p. 349.)

size of a pixel. The 11×11 mask computes a response by integrating the input of 121 pixels and thus responds to larger image features and not smaller ones. Integrating 121 pixels can take a lot more time than integrating 9 of them done in software.

5.7.2 On Human Edge Detection

We now describe an artificial neural network (ANN) architecture which implements the LOG filtering operation in a highly parallel manner. The behavior of this network has been shown to simulate some of the known behaviors of the human visual system. Moreover, invasive techniques have also shown that the visual systems of cats and monkeys produce electrical signals consistent with the behavior of the neural network architecture. Figure 5.23 shows the processing of 1D signals. A step edge is sensed at various points by cells of the

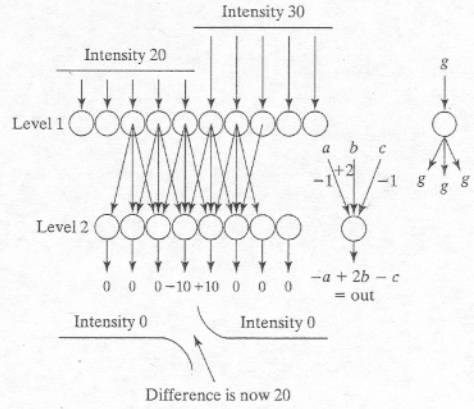


Figure 5.23 Producing the *Mach band effect* using an ANN architecture. Intensity is sensed by cells of the retina (Level 1), which then stimulate integrating cells in a higher level layer (Level 2).

retinal array. These level 1 cells stimulate the integrating cells at level 2. Each physical connection between level 1 cell i and level 2 cell j has an associated weight w_{ij} which is multiplied by the stimulus being communicated before it is summed in cell j. The output of cell j is $y_j = \sum_{i=1}^{N} w_{ij}x_i$ where x_i is the output of the *ith* first level cell and N is the total number of first level cells. (Actually, we need only account for those cells i directly connected to the second level cell j). By having the weights associated with each connection, it is possible, and common, to have the same cell i give positive input to cell j and negative input to cell $k \neq j$. Figure 5.23 shows that each cell j of the second level computes its output as $-a + 2b - c$, corresponding to the mask $[-1, 2, -1]$: the weight 2 is applied to the central input, while the (inhibitory) inputs a and b are each weighted by -1.

This kind of architecture can be defined for any kind of mask and permits a highly parallel implementation of cross correlation for filtering or feature detection. The psychologist Mach noted that humans perceive an edge between two regions as if it were pulled apart to exaggerate the intensity difference, as is done in Figure 5.23. Note that this architecture and mask create the *zero crossing* at the location of the edge between two cells, one of which produces positive output and the other negative output. The Mach band effect changes the perceived shape of joining surfaces and is evident in computer graphics systems that display polyhedral objects via shaded faces. Figure 5.24 shows seven constant regions, stepping from gray level 31 to 255 in steps of 32. Do you perceive concave panels in 3D such as on a Doric column from a Greek temple?

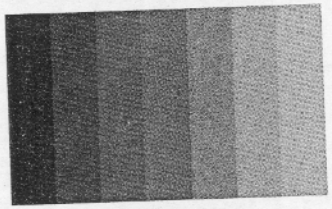


Figure 5.24 Seven constant stripes generated with gray levels $31 + 32k$, $k = 1, 7$. Due to the Mach band effect, humans perceive scalloped, or concave, panels.

Figure 5.25 extends Figure 5.23 to 2D images. Each set of retinal cells connected to integrating cell j comprise what is called the *receptive field* of that cell. To perform edge detection via a second derivative operation, each receptive field has a center set of cells which have positive weights w_{ij} with respect to cell j and a surrounding set of cells with negative weights. Retinal cells b and c are in the center of the receptive field of integrating cell A, whereas retinal cells a and d are in the surround and provide inhibitory input. Retinal cell d is in the center of the receptive field of integrating cell B, however, and cell c is in its surround. The sum of the weights from the center and the surround should be zero so that the integrating cell has neutral output on constant regions. Because the center and surround are circular, output will not be neutral whenever a straight region boundary just nips the center area, regardless of the angle. Thus, each integrating cell is an *isotropic* edge detector cell. Additionally, if a small region contrasting with the background images within the center of the receptive field the integrating cell will also respond, making it a spot

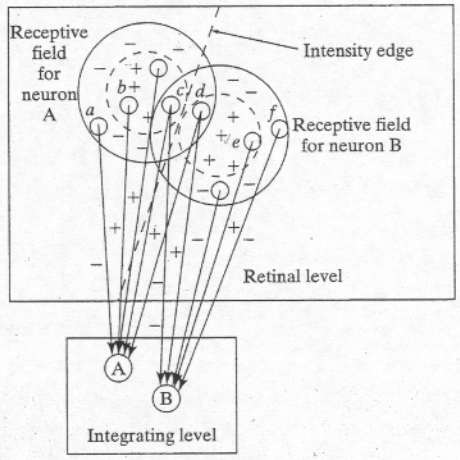


Figure 5.25 A 3D ANN architecture for LOG filtering.

detector as well. Figure 5.21 shows the result of convolving the smallest of the LOG masks with an image containing two regions. The result at the right of the figure shows how the operation determines the boundary between the regions via zero crossings. An 11×11 mask corresponding to the Laplacian of a Gaussian with $\sigma^2 = 2$ is shown in Figure 5.22. The tiny mask is capable of finding the boundary between tiny regions and is sensitive to high curvature bouundaries but will also respond to noise texture. The larger mask performs a lot of smoothing and will only respond to boundaries between larger regions with smoother perimeter.

Exercise 5.11

Give more detailed support to the above arguments that the integrating cells shown in Figure 5.25 (*a*) respond to contrasting spots or blobs that image within the center of the field, and (*b*) respond to boundaries between two large regions that just barely cross into the center of the field.

5.7.3 Marr-Hildreth Theory

David Marr and Ellen Hildreth proposed LOG filtering to explain much of the low level behavior of human vision. Marr proposed that the objective of low level human visual processing was the construction of a primal sketch which was a 2D description containing lines, edges, and blobs. (The primal sketches derived from the two eyes would then be processed further to derive 3D interpretations of the scene.) To derive a primal sketch, Marr and Hildreth proposed an organization based on LOG filtering with 4 or 5 different spreads σ. The mathematical properties outlined above explained the results of both perceptual experiments on humans and invasive experiments with animals. LOG filters with large σ would

detect broad edges while those with small σ would focus on small detail. Coordination of the output from the different scales could be done at a higher level, perhaps with the large scale detections guiding those at the small scale. Subsequent work has produced different practical *scale space* methods of integrating output from detectors of different sizes.

Figure 5.26 shows an image processed at two different levels of Gaussian smoothing. The center image shows good representation of major objects and edges, whereas the rightmost image shows both more image detail and more noise. Note that the ship and part of the sand/water boundary is represented in the rightmost image but not in the center image. Marr's primal sketch also contained descriptions of virtual lines, which are formed by similar detected features organized along an image curve. These might be the images of a dotted line, a row of shrubs, etc. A synthetic image containing a virtual line is shown in Figure 5.27 along with the output of two different LOG filters. Both LOG filters give a

Original Smoothed $\sigma = 4$ Smoothed $\sigma = 1$

Figure 5.26 An input image (a) is smoothed using Gaussian filters of size (b) $\sigma = 4$ and (c) $\sigma = 1$ before performing edge detection. More detail and more noise is shown for the smaller filter. (Photo by David Shaffer 1998.)

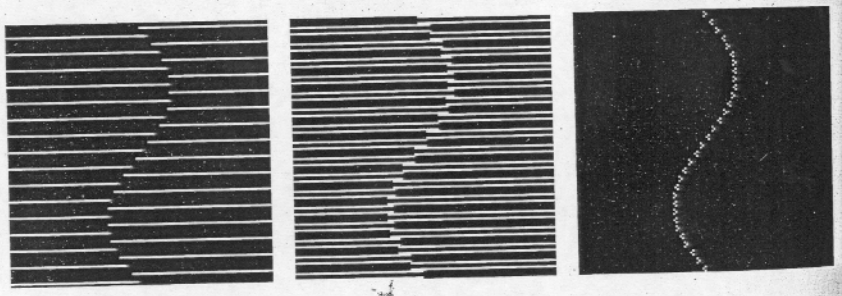


Figure 5.27 (left) A virtual line formed by the termination points of a line texture—perhaps this is two pieces of wrapping paper overlaid; (center) output from a specific 4×4 LOG filter responds to both lines and endpoints; and (right) a different 3×3 LOG filter responds only to the endpoints.

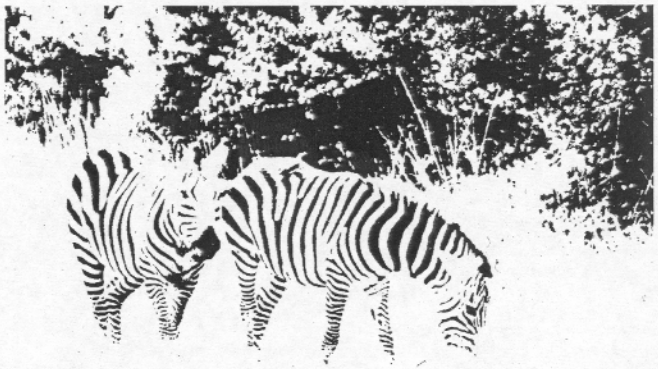


Figure 5.28 Picture obtained by thresholding: object boundaries are formed by virtual curves formed by the ends of stripes denoting cross sections of generalized cylinders. (Original photo by Eleanor Harding.)

response to the ends of the stripes; one is sensitive to the edges of the stripes as well, but the other is not. Figure 5.28 shows the same principle in a real image that was thresholded to obtain an artistic texture. Recent progress in researching the human visual system and brain has been rapid: results seem to complicate the interpretations of earlier work on which Marr and Hildreth based their mathematical theory. Nevertheless, use of variable-sized Gaussian and LOG filters is firmly established in computer vision.

5.8 THE CANNY EDGE DETECTOR

The Canny edge detector is a very popular and effective operator, and it is important to introduce it here, although details are placed in Chapter 10. The Canny operator first smoothes the intensity image and then produces extended contour segments by following high gradient magnitudes from one neighborhood to another. Figure 5.29 shows edge detection in fairly difficult outdoor images. The contours of the arch of St. Louis are detected quite well in Figure 5.29: using a parameter of $\sigma = 1$ detects some of the metal seams of the arch as well as some internal variations in the tree, whereas use of $\sigma = 4$ detects only the exterior boundaries of these objects. As shown in the bottom row of Figure 5.29, the operator isolates many of the texture elements of the checkered flags. For comparison, use of the Roberts operator with a low threshold on gradient magnitude is shown: more texture elements of the scene (grass and fence) are evident, although they have less structure than those in the Canny output. The algorithm for producing contour segments is treated in detail in Section 10.3.2.

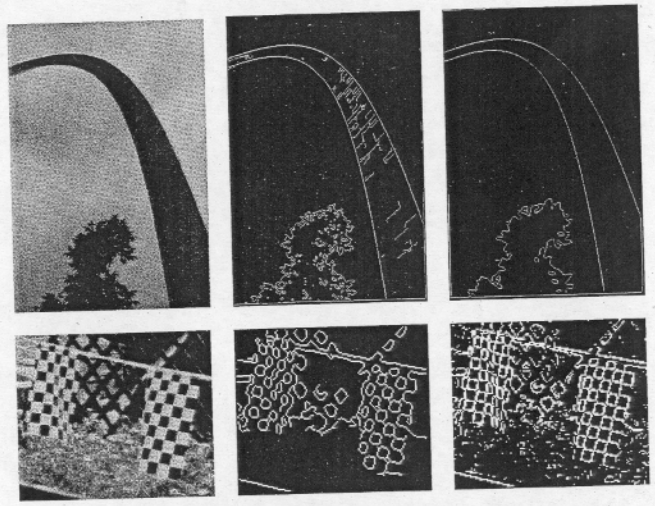


Figure 5.29 (top left) Image of the great arch in St. Louis; (top center) results of Canny operator with $\sigma = 1$; (top right) results of Canny operator with $\sigma = 4$; (bottom left) image with textures; (bottom center) results of Canny operator with $\sigma = 1$; and (bottom right) results of Roberts operator thresholded to pass the top 20 percent of the pixels in gradient magnitude.

5.9 MASKS AS MATCHED FILTERS*

Here we give a theoretical basis for the concept that the response of a mask to a certain image neighborhood is proportional to how similar that neighborhood is to the mask. The important practical result of this is that we now know how to design masks to detect specific features—we just design a mask that is similar to the feature[s] we want to detect. This will serve us well in both edge and texture detection and also in detecting other special patterns such as holes or corners. We introduce the concepts using 1-dimensional signals, which are important in their own right, and which might correspond to rows or columns or any other cut through a 2D image. The concepts and mathematical theory immediately extend to the 2D case.

5.9.1 The Vector Space of all Signals of n Samples

For a given $n \geq 1$, the set of all vectors of n real coordinates forms a *vector space*. The practical and powerful vector space operations which we use are summarized next. The reader has probably already worked with vectors with $n = 2$ or $n = 3$ when studying analytical geometry or calculus. For $n = 2$ or $n = 3$, the notion of vector length is the same as that used in plane geometry and 3D analytical geometry since Euclid. In the domain of signals, length is related to energy of the signal, defined as the squared length of the signal,

or equivalently, just the sum of the squares of all coordinates. Signal energy is an extremely useful concept as will be seen.

47 Definition. The **energy** of signal $S = [s_1, s_2, \ldots, s_n]$ is $\|S\|^2 = s_1^2 + s_2^2 + \cdots + s_n^2$.

Note that in many applications, the full range of real-valued signals do not arise because negative values are impossible for the coordinates. For example, a 12-dimensional vector recording the rainfall at a particular location for each of the 12 months should have no negative coordinates. Similarly, intensities along an image row are commonly kept in a non-negative integer range. Nevertheless, the vector space interpretation is still useful, as we shall see. Often, the mean signal value is subtracted from all coordinates in making an interpretation, and this may shift some of the coordinate values below zero. Moreover, it is quite common to have negative values in masks, which are templates or models of the shape of parts of signals.

Basic definitions for a vector space with a defined vector length

Let U and V be any two vectors; u_i and v_i be real numbers denoting the coordinates of these vectors; and let a, b, c, etc. be any real numbers denoting scalars.

48 Definition. For vectors $U = [u_1, u_2, \ldots, u_n]$ and $V = [v_1, v_2, \ldots, v_n]$ their **vector sum** is the vector $U \oplus V = [u_1 + v_1, u_2 + v_2, \ldots, u_n + v_n]$.

49 Definition. For vector $V = [v_1, v_2, \ldots, v_n]$ and real number (scalar) a the **product of the vector and scalar** is the vector $aV = [av_1, av_2, \ldots, av_n]$.

50 Definition. For vectors $U = [u_1, u_2, \ldots, u_n]$ and $V = [v_1, v_2, \ldots, v_n]$ their **dot product, or scalar product** is the real number $U \circ V = u_1v_1 + u_2v_2 + \cdots + u_nv_n$.

51 Definition. For vector $V = [v_1, v_2, \ldots, v_n]$ its **length, or norm**, is the non-negative real number $\|V\| = V \circ V = (v_1v_1 + v_2v_2 + \cdots + v_nv_n)^{1/2}$.

52 Definition. Vectors U and V are **orthogonal** if and only if $U \circ V = 0$.

53 Definition. The **distance between vectors** $U = [u_1, u_2, \ldots, u_n]$ and $V = [v_1, v_2, \ldots, v_n]$ is the length of their difference $d(U, V) = \|U - V\|$.

54 Definition. A **basis** for a vector space of dimension n is a set of n vectors $\{w_1, w_2, \ldots, w_n\}$ that are independent and that span the vector space. The spanning property means that any vector V can be expressed as a linear combination of basis vectors: $V = a_1w_1 \oplus a_2w_2 \oplus \cdots \oplus a_nw_n$. The independence property means that none of the basis vectors w_i can be represented as a linear combination of the others.