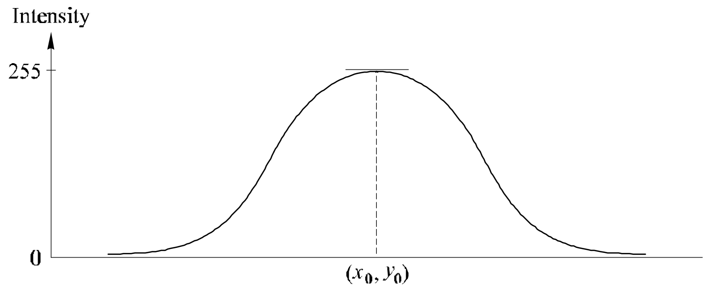
Problem 2.7

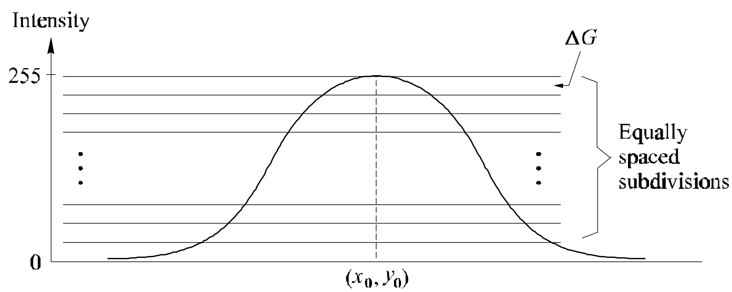
The image in question is given by

$$\begin{aligned} f(x,y) &= i(x,y)r(x,y) \\ &= 255e^{-[(x-x_0)^2+(y-y_0)^2]}(1.0) \\ &= 255e^{-[(x-x_0)^2+(y-y_0)^2]} \end{aligned}$$

A cross section of the image is shown in Fig. P2.7(a). If the intensity is quantized using m bits, then we have the situation shown in Fig. P2.7(b), where $\triangle G = (255+1)/2^m$. Since an abrupt change of 8 gray levels is assumed to be detectable by the eye, it follows that $\triangle G = 8 = 256/2m$, or $m = 5$. In other words, 32, or fewer, gray levels will produce visible false contouring.



(a)



(b)

Figure P2.7

Problem 2.8

The use of two bits ($m = 2$) of intensity resolution produces four gray levels in the range 0 to 255. One way to subdivide this range is to let all levels between 0 and 63 be coded as 63, all levels between 64 and 127 be coded as 127, and so on. The image resulting from this type of subdivision is shown in Fig. P2.8. Of course, there are other ways to subdivide the range $[0, 255]$ into four bands.

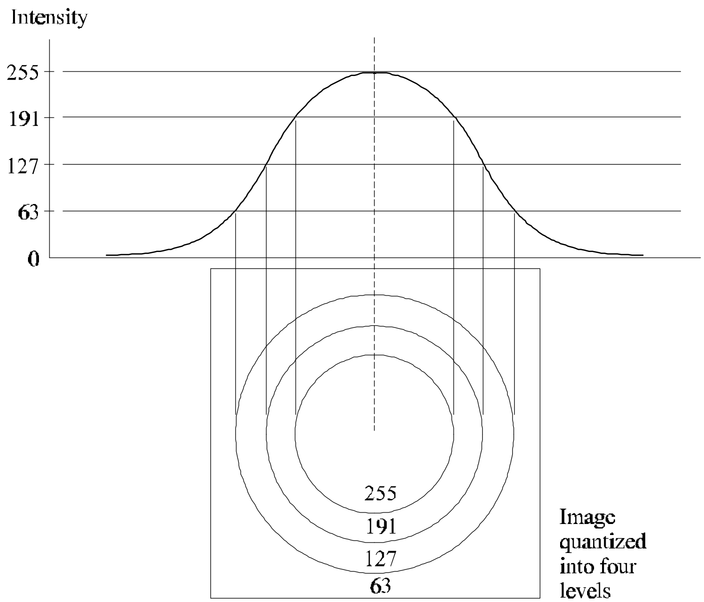


Figure P2.8

Problem 2.11

Let p and q be as shown in Fig. P2.11. Then, (a) S_1 and S_2 are not 4-connected because q is not in the set $N_4(p)$; (b) S_1 and S_2 are 8-connected because q is in the set $N_8(p)$; (c) S_1 and S_2 are m-connected because (i) q is in $N_D(p)$, and (ii) the set $N_4(p) \cap N_4(q)$ is empty.

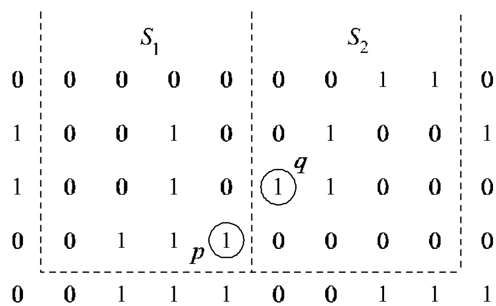


Figure P2.11

Problem 2.15

(a) When $V = \{0, 1\}$, 4-path does not exist between p and q because it is impossible to get from p to q by traveling along points that are both 4-adjacent and also have values from V. Figure P2.15(a) shows this condition; it is not possible to get to q. The shortest 8-path is shown in Fig. P2.15(b); its length is 4. The length of the shortest m- path (shown dashed) is 5. Both of these shortest paths are unique in this case. (b) One possibility for the shortest 4-path when $V = \{1, 2\}$ is shown in Fig. P2.15(c); its length is 6. It is easily verified that another 4-path of the same length exists between p and q. One possibility for the shortest 8-path (it is not unique) is shown in Fig. P2.15(d); its length is 4. The length of a shortest m-path (shown dashed) is 6. This path is not unique.

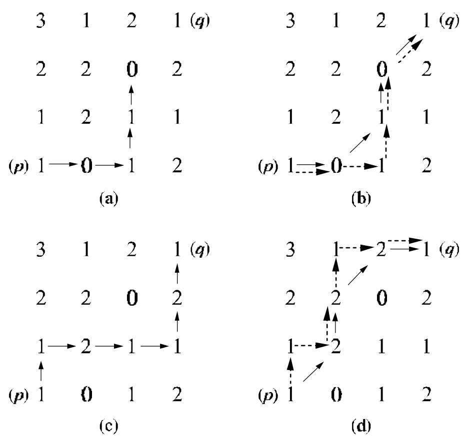


Figure P2.15

Problem 3.3

The transformations required to produce the individual bit planes are nothing more than mappings of the truth table for eight binary variables. In this truth table, the values of the 7th bit are 0 for byte values 0 to 127, and 1 for byte values 128 to 255, thus giving the transformation mentioned in the problem statement. Note that the given transformed values of either 0 or 255 simply indicate a binary image for the 7th bit plane. Any other two values would have been equally valid, though less conventional.

Continuing with the truth table concept, the transformation required to produce an image of the 6th bit plane outputs a 0 for byte values in the range [0, 63], a 1 for byte values in the range [64, 127], a 0 for byte values in the range [128, 191], and a 1 for byte values in the range [192, 255]. Similarly, the transformation for the 5th bit plane alternates between eight ranges of byte values, the transformation for the 4th bit plane alternates between 16 ranges, and so on. Finally, the output of the transformation for the 0th bit plane alternates between 0 and 255 depending as the byte values are even or odd. Thus, this transformation alternates between 128 byte value ranges, which explains why an image of the 0th bit plane is usually the busiest looking of all the bit plane images.

Problem 3.7

The general histogram equalization transformation function is

$$s = T(r) = \int_0^r p_r(w)\, dw.$$

There are two important points to which the student must show awareness in answering this problem. First, this equation assumes only positive values for r. However, the Gaussian density extends in general from $-\infty$ to ∞. Recognition of this fact is important. Once recognized, the student can approach this difficulty in several ways. One good answer is to make some assumption, such as the standard deviation being small enough so that the area of the curve under $p_r(r)$ for negative values of r is negligible. Another is to scale up the values until the area under the negative tail is negligible. The second major point is to recognize is that the transformation function itself,

$$s = T(r) = \frac{1}{\sqrt{2\pi}\sigma} \int_0^r e^{-\frac{(w-m)^2}{2\sigma^2}} dw$$

has no closed-form solution. This is the cumulative distribution function of the Gaussian density, which is either integrated numerically, or its values are looked up in a table. A third, less important point, that the student should address is the high-end values of r. Again, the Gaussian PDF extends to $+\infty$. One possibility here is to make the same

assumption as above regarding the standard
enough value so that the area under the po
scaling reduces the standard deviation).

Another principal approach the student can take is to work with histograms, in which case the transformation function would be in the form of a summation. The issue of negative and high positive values must still be addressed, and the possible answers suggested above regarding these issues still apply. The student needs to indicate that the histogram is obtained by sampling the continuous function, so some mention should be made regarding the number of samples (bits) used. The most likely answer is 8 bits, in which case the student needs to address the scaling of the function so that the range is $[0, 255]$.

Problem 3.10

First, we obtain the histogram equalization transformation:

$$s = T(r) = \int_0^r p_r(w)\,dw = \int_0^r (-2w+2)\,dw = -r^2 + 2r.$$

Next we find

$$v = G(z) = \int_0^z p_z(w)\ dw = \int_0^z 2w\ dw = z^2.$$

Finally,

$$z = G^{-1}(v) = \pm\sqrt{v}.$$

But only positive gray levels are allowed, so $z = \sqrt{v}$. Then, we replace v with s, which in turn is $-r^2 + 2r$, and we have

$$z = \sqrt{-r^2 + 2r}.$$

Problem 3.13

Using 10 bits (with one bit being the sign bit) allows numbers in the range -511 to 511. The process of repeated subtractions can be expressed as

$$\begin{aligned} d_K(x,y) &= a(x,y) - \sum_{k=1}^{K} b(x,y) \\ &= a(x,y) - K \times b(x,y) \end{aligned}$$

where K is the largest value such that $d_K(x,y)$ does not exceed -511 at any coordinates (x,y), at which time the subtraction process stops. We know nothing about the images, only that both have values ranging from 0 to 255. Therefore, all we can determine are the maximum and minimum number of times that the subtraction can be carried out and the possible range of gray-level values in each of these two situations.

Because it is given that $g(x,y)$ has at least one pixel valued 255, the maximum value that K can have before the subtraction exceeds -511 is 3. This condition occurs when, at some pair of coordinates (s,t), $a(s,t) = b(s,t) = 255$. In this case, the possible range of values in the difference image is -510 to 255. The latter condition can occur if, at some pair of coordinates (i,j), $a(i,j) = 255$ and $b(i,j) = 0$.

The minimum value that K will have is 2, which occurs when, at some pair of coordinates, $a(s,t) = 0$ and $b(s,t) = 255$. In this case, the possible range of values in the difference image again is -510 to 255. The latter condition can occur if, at some pair of coordinates (i,j), $a(i,j) = 255$ and $b(i,j) = 0$.

roblem 3.17

(a) Consider a 3×3 mask first. Since all the coefficients are 1 (we are ignoring the 1/9 scale factor), the net effect of the lowpass filter operation is to add all the gray levels of pixels under the mask. Initially, it takes 8 additions to produce the response of the mask. However, when the mask moves one pixel location to the right, it picks up only one new column. The new response can be computed as

$$R_{\text{new}} = R_{\text{old}} - C_1 + C_3$$

where C_1 is the sum of pixels under the first column of the mask before it was moved, and C_3 is the similar sum in the column it picked up after it moved. This is the basic box-filter or moving-average equation. For a 3×3 mask it takes 2 additions to get C_3 (C_1 was already computed). To this we add one subtraction and one addition to get R_{new}. Thus, a total of 4 arithmetic operations are needed to update the response after one move. This is a recursive procedure for moving from left to right along one row of the image. When we get to the end of a row, we move down one pixel (the nature of the computation is the same) and continue the scan in the opposite direction.

For a mask of size $n \times n$, $(n-1)$ additions are needed to obtain C_3, plus the single subtraction and addition needed to obtain R_{new}, which gives a total of $(n+1)$ arithmetic operations after each move. A brute-force implementation would require $n^2 - 1$ additions after each move.

(b) The computational advantage is

$$A = \frac{n^2 - 1}{n + 1} = \frac{(n+1)(n-1)}{(n+1)} = n - 1.$$

The plot of A as a function of n is a simple linear function starting at $A = 1$ for $n = 2$.

Problem 3.23

There are at most q^2 points in the area for which we want to reduce the gray level of each pixel to one-tenth its original value. Consider an averaging mask of size $n \times n$ encompassing the $q \times q$ neighborhood. The averaging mask has n^2 points of which we are assuming that q^2 points are from the object and the rest from the background. Note that this assumption implies separation between objects at least the area of the mask all around each object. The problem becomes intractable unless this assumption is made. This condition was not given in the problem statement on purpose in order to force the student to arrive at that conclusion. If the instructor wishes to simplify the problem, this should then be mentioned when the problem is assigned. A further simplification is to tell the students that the gray level of the background is 0.

Let B represent the gray level of background pixels, let a_i denote the gray levels of points inside the mask and o_i the levels of the objects. In addition, let S_a denote the set of points in the averaging mask, S_o the set of points in the object, and S_b the set of points in the mask that are not object points. Then, the response of the averaging mask at any point on the image can be written as

$$\begin{aligned} R &= \frac{1}{n^2}\sum_{a_i \in S_a} a_i \\ &= \frac{1}{n^2}\left[\sum_{o_j \in S_o} o_j + \sum_{a_k \in S_b} a_k\right] \\ &= \frac{1}{n^2}\left[\frac{q^2}{q^2}\sum_{o_j \in S_o} o_j\right] + \frac{1}{n^2}\left[\sum_{a_k \in S_b} a_k\right] \\ &= \frac{q^2}{n^2}\overline{Q} + \frac{1}{n^2}\left[(n^2 - q^2)B\right] \end{aligned}$$

where $\overline{Q}$ denotes the average value of object points. Let the maximum expected average value of object points be denoted by $\overline{Q}_{\max}$. Then we want the response of the mask at any point on the object under this maximum condition to be less than one-tenth $\overline{Q}_{\max}$, or

$$\frac{q^2}{n^2}\overline{Q}_{\max} + \frac{1}{n^2}\left[(n^2 - q^2)B\right] < \frac{1}{10}\overline{Q}_{\max}$$

from which we get the requirement

$$n > q\left[\frac{10(\overline{Q}_{\max} - B)}{(\overline{Q}_{\max} - 10B)}\right]^{1/2}$$

for the minimum size of the averaging mask. Note that if the background gray-level is 0, we the minimum mask size is $n < \sqrt{10}q$. If this was a fact specified by the instructor,

Problem 4.3

The inverse DFT of a constant A in the frequency domain is an impulse of strength A in the spatial domain. Convolving the impulse with the image copies (multiplies) the value of the impulse at each pixel location in the image.

Problem 4.9

The complex conjugate simply changes j to $-j$ in the inverse transform, so the image on the right is given by

$$\begin{aligned}
\Im^{-1}\left[F^*(u,v)\right] &= \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} F(u.v)e^{-j2\pi(ux/M\,+\,vy/N)} \\
&= \sum_{x=0}^{M-1}\sum_{y=0}^{N-1} F(u.v)e^{j2\pi(u(-x)/M\,+\,v(-y)/N)} \\
&= f(-x,-y)
\end{aligned}$$

which simply mirrors $f(x,y)$ about the origin, thus producing the image on the right.

Problem 4.12

(a) The ring in fact has a dark center area as a result of the highpass operation only (the following image shows the result of highpass filtering only). However, the dark center area is averaged out by the lowpass filter. The reason the final result looks so bright is that the discontinuity (edge) on boundaries of the ring are much higher than anywhere else in the image, thus giving an averaged area whose gray level dominates.

(b) Filtering with the Fourier transform is a linear process. The order does not matter.

Figure P4.12

Problem 4.16

(a) The key for the student to be able to solve the problem is to treat the number of applications (denoted by K) of the highpass filter as 1 minus K applications of the corresponding lowpass filter, so that

$$\begin{aligned} H_K(u,v) &= H_K(u,v)F(u,v) \\ &= \left[1 - e^{-KD^2(u,v)/2D_0^2}\right] H(u,v) \end{aligned}$$

where the Gaussian lowpass filter is from Problem 4.13. Students who start directly with the expression of the Gaussian highpass filter $\left[1 - e^{-KD^2(u,v)/2D_0^2}\right]$ and attempt to raise it to the Kth power will run into a dead end.

The solution to this problem parallels the solution to Problem 4.13. Here, however, the filter will approach a notch filter that will take out $F(0,0)$ and thus will produce an image with zero average values (this implies negative pixels). So, there is a value of K after which the result of repeated highpass filtering will simply produce a constant image.

Problem 4.18

The answer is no. The Fourier transform is a linear process, while the square and square roots involved in computing the gradient are nonlinear operations. The Fourier transform could be used to compute the derivatives (as differences—see Prob.4.15), but the squares, square root, or absolute values must be computed directly in the spatial domain.

Problem 4.20

From Eq. (4.4-3), the transfer function of a Butterworth highpass filter is

$$H(u,v) = \frac{1}{1 + \left[\frac{D_0}{D(u,v)}\right]^{2n}}.$$

We want the filter to have a value of γ_L when $D(u,v) = 0$, and approach γ_H for high values of $D(u,v)$. The preceding equation is easily modified to accomplish this:

$$H(u,v) = \gamma_L + \frac{(\gamma_H - \gamma_L)}{1 + \left[\frac{D_0}{D(u,v)}\right]^{2n}}.$$

The value of n controls the sharpness of the transition between γ_L and γ_H.

Problem 4.22

(a) Padding an image with zeros increases its size, but not its gray-level content. Thus, the average gray-level of the padded image is lower than that of the original image. This implies that $F(0,0)$ in the spectrum of the padded image is less than $F(0,0)$ in the original image (recall that $F(0,0)$ is the average value of the corresponding image). Thus, we can visualize $F(0,0)$ being lower in the spectrum on the right, with all values away from the origin being lower too, and covering a narrower range of values. That's the reason the overall contrast is lower in the picture on the right.

(b) Padding an image with 0's introduces significant discontinuities at the borders of the original images. This process introduces strong horizontal and vertical edges, where the image ends abruptly and then continues with 0 values. These sharp transitions correspond to the strength of the spectrum along the horizontal and vertical axes of the spectrum.

Problem 5.1

The solutions to (a), (b), and (c) are shown in Fig. P5.1, from left to right:

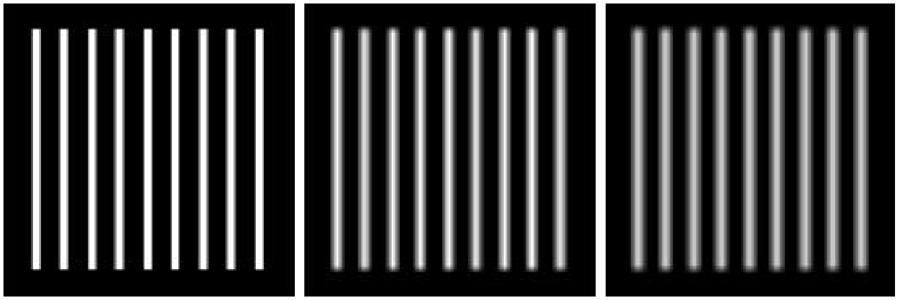


Figure P5.1

Problem 5.10

(a) The key to this problem is that the geometric mean is zero whenever any pixel is zero. Draw a profile of an ideal edge with a few points valued 0 and a few points valued 1. The geometric mean will give only values of 0 and 1, whereas the arithmetic mean will give intermediate values (blur).

(b) Black is 0, so the geometric mean will return values of 0 as long as at least one pixel in the window is black. Since the center of the mask can be outside the original black area when this happens, the figure will be thickened.

Problem 5.26

One possible solution: (1) Average images to reduce noise. (2) obtain blurred image of a bright, single star to simulate an impulse (the star should be as small as possible in the field of view of the telescope to simulate an impulse as closely as possible. (3) The Fourier transform of this image will give $H(u, v)$. (4) Use a Wiener filter and vary K until the sharpest image possible is obtained.

Problem 6.5

At the center point we have

$$\frac{1}{2}R + \frac{1}{2}B + G = \frac{1}{2}(R + G + B) + \frac{1}{2}G = \text{midgray} + \frac{1}{2}G$$

which looks to a viewer like pure green with a boot in intensity due to the additive gray component.

Problem 6.15

The hue, saturation, and intensity images are shown in Fig. P6.15, from left to right.

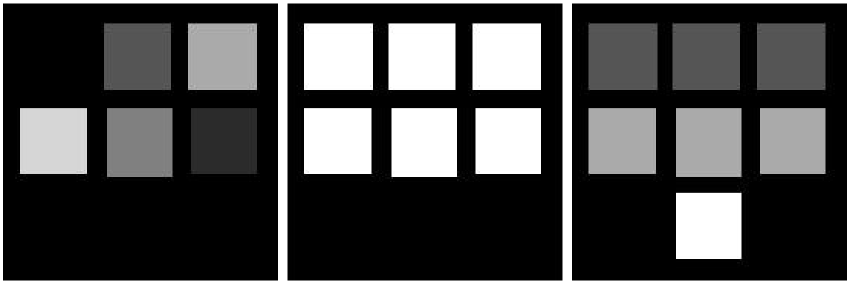


Figure P6.15

Problem 6.17

(a) Because the infrared image which was used in place of the red component image has very high gray-level values.

(b) The water appears as solid black (0) in the near infrared image [Fig. 6.27(d)]. Threshold the image with a threshold value slightly larger than 0. The result is shown in Fig. P6.17. It is clear that coloring all the black points in the desired shade of blue presents no difficulties.

(c) Note that the predominant color of natural terrain is in various shades of red. We already know how to take out the water from (b). Thus a method that actually removes the ”background” of red and black would leave predominantly the other man-made structures, which appear mostly in a bluish light color. Removal of the red [and the black if you do not want to use the method as in (b)] can be done by using the technique discussed in Section 6.7.2.

Problem 7.1

Following the explanation in Example 7.1, the decoder is as shown in Fig. P7.1

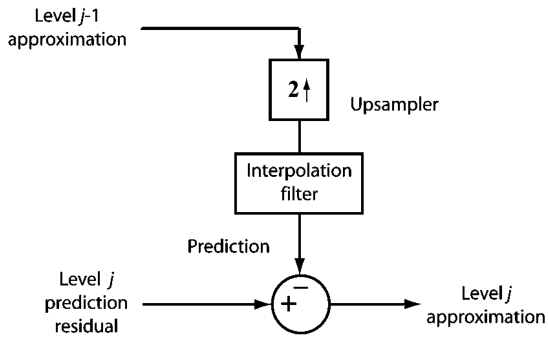


Figure P7.1

Problem 7.24

As can be seen in the sequence of images that are shown, the DWT is not shift invariant. If the input is shifted, the transform changes. Since all original images in the problem are 128×128, they become the $W_\varphi(7, m, n)$ inputs for the FWT computation process. The filter bank of Fig. 7.22(a) can be used with $j + 1 = 7$. For a single scale transform, transform coefficients $W_\varphi(6, m, n)$ and $W_\psi^i(6, m, n)$ for $i = H, V, D$ are generated. With Haar wavelets, the transformation process subdivides the image into non-overlapping 2×2 blocks and computes 2-point averages and differences (per the scaling and wavelet vectors). Thus, there are no horizontal, vertical, or diagonal detail coefficients in the first two transforms shown; the input images are constant in all 2×2 blocks (so all differences are 0). If the original image is shifted by 1 pixel, detail coefficients are generated since there are then 2×2 areas that are not constant. This is the case in the third transform shown.

Problem 9.6

Refer to Fig. P9.6. The center of each structuring element is shown as a black dot. Solution (a) was obtained by eroding the original set (shown dashed) with the structuring element shown (note that the origin is at the bottom, right). Solution (b) was obtained by eroding the original set with the tall rectangular structuring element shown. Solution (c) was obtained by first eroding the image shown down to two vertical lines using the rectangular structuring element; this result was then dilated with the circular structuring element. Solution (d) was obtained by first dilating the original set with the large disk shown. Then dilated image was then eroded with a disk of half the diameter of the disk used for dilation.

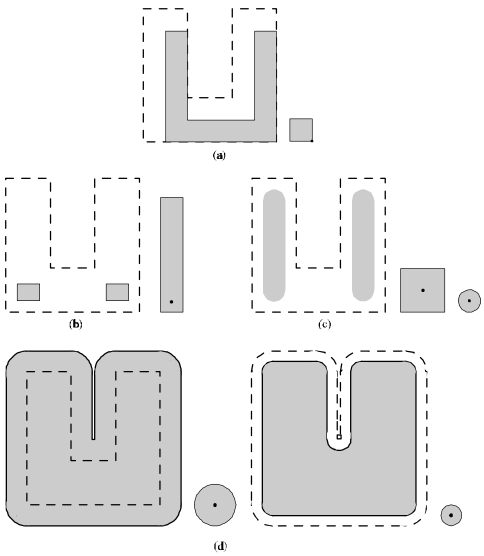


Figure P9.6

Problem 9.7

The solutions to (a) through (d) are shown from top to bottom in Fig. P9.7.

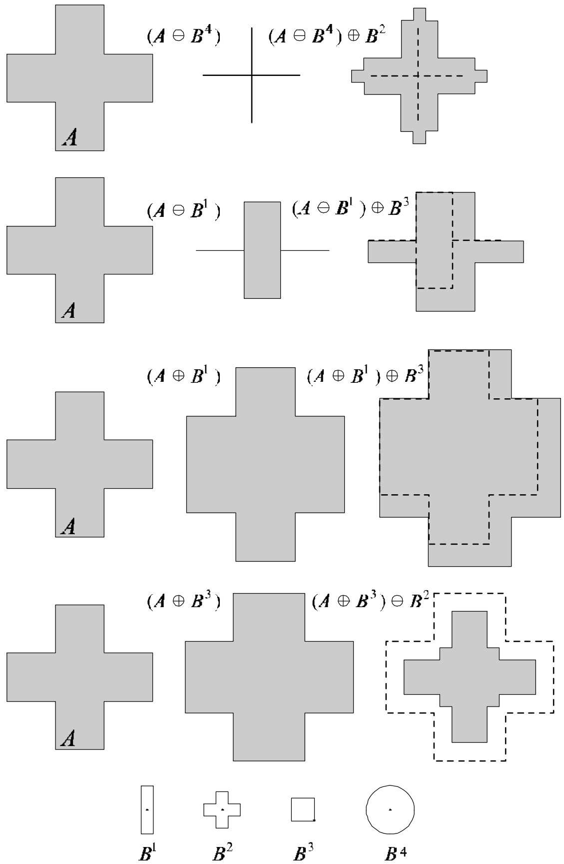


Figure P9.7

Problem 10.18

(a) The number of boundary points between black and white regions is much larger in the image on the right. When the images are blurred, the boundary points will give rise to a larger number of different values for the image on the right, so the histograms of the two blurred images will be different.

(b) To handle border effects, we surround the image with a border of 0's. We assume that the image is of size $N \times N$ (the fact that the image is square is evident from the right image in the problem statement). Blurring is implemented by a 3×3 mask whose coefficients are $1/9$. Figure P10.18 shows the different types of values that the blurred left image (see problem statement) will have. These values are summarized in Table P10.18-1. It is easily verified that the sum of the numbers on the left column of the table is N^2.

Table P10.18-1

No. of Points	Value
$N\left(\frac{N}{2}-1\right)$	0
2	$2/9$
$N-2$	$3/9$
4	$4/9$
$3N-8$	$6/9$
$(N-2)\left(\frac{N}{2}-2\right)$	1

A histogram is easily constructed from the entries in this table. A similar (tedious, but not difficult) procedure yields the results shown in Table P10.18-2 for the checkerboard image.

Table P10.18-2

No. of Points	Value
$\frac{N^2}{2} - 14N + 98$	0
28	$2/9$
$14N - 224$	$3/9$
128	$4/9$
98	$5/9$
$16N - 256$	$6/9$
$\frac{N^2}{2} - 16N + 128$	1

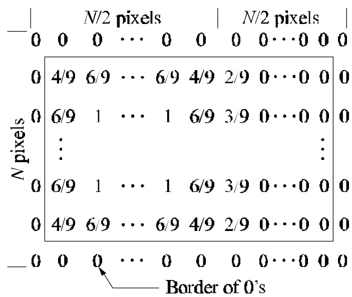


Figure P10.18

Problem 11.10

(a) The number of symbols in the first difference is equal to the number of segment primitives in the boundary, so the shape order is 12.

(b) Starting at the top left corner,

Chain code:	000332123211
Difference:	300303311330
Shape no.:	003033113303