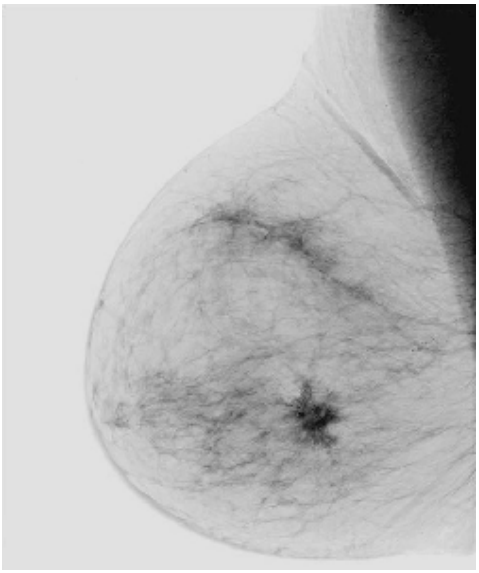
Text figure (3.4): Negative Transform

```
%Matlab Code:
clear all; close all;
img  = imread('breast.jpg');
img2 = 1 - im2double(img);
figure; subplot(1,2,1); imshow(img); title('Original Image');
subplot(1,2,2); imshow(img2); title('Image after Negative Transform');
```

Original Image

Image after Negative Transform



Text figure (3.5) Log Transformation:

```
%Matlab Code
clear all; close all;
img  = imread('fourierspectrum.jpg');
img2 = log10(1+256*im2double(img));
img2 = [img2 - min(img2(:))] ./ max(img2(:) - min(img2(:)));
figure; subplot(1,2,1); imshow(img); title('Original Image');
subplot(1,2,2); imshow(img2); title('Image after Logarithmic
       Transform');
```

Original Image

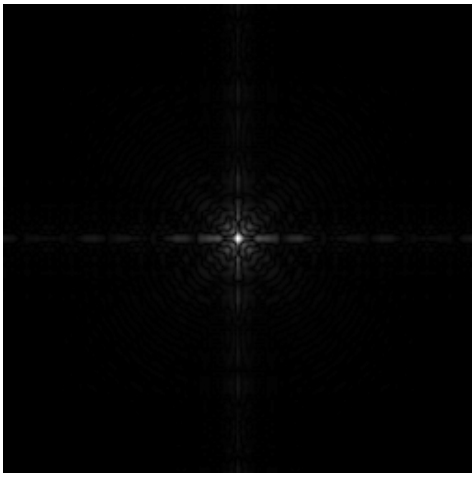
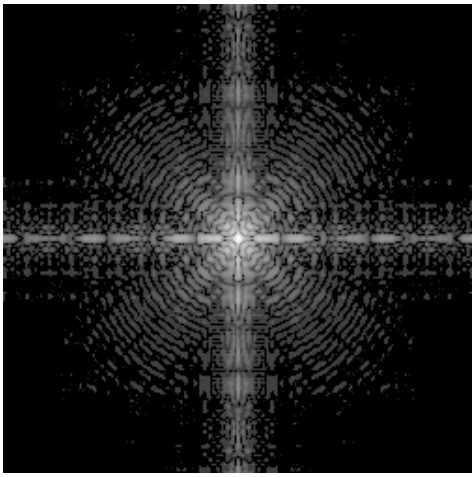


Image after Logarithmic Transform



Text figure (3.8) Gamma Correction

```
%Matlab Code
clear all; close all;
img  = imread('spine.jpg');
img2 = double(img).^(0.6); img3 = double(img).^(0.4); img4 =
double(img).^(0.3);
img2 = [img2 - min(img2(:))] ./ max(img2(:) - min(img2(:)));
img3 = [img3 - min(img3(:))] ./ max(img3(:) - min(img3(:)));
img4 = [img4 - min(img4(:))] ./ max(img4(:) - min(img4(:)));

figure; subplot(2,2,1); imshow(img); title('Original Image');
subplot(2,2,2); imshow(img2); title('Image after Gamma Transform,
       \gamma = 0.6');
subplot(2,2,3); imshow(img3); title('Image after Gamma Transform,
       \gamma = 0.4');
subplot(2,2,4); imshow(img4); title('Image after Gamma Transform,
       \gamma = 0.3');
```

Original Image

Image after Gamma Transform, γ = 0.6

Image after Gamma Transform, γ = 0.4

Image after Gamma Transform, γ = 0.3

Text figure (3.9) Gamma Correction

```
%Matlab Code
clear all; close all;
img  = imread('aerial.jpg');
img2 = double(img).^(3); img3 = double(img).^(4); img4 =
double(img).^(5);
img2 = [img2 - min(img2(:))] ./ max(img2(:) - min(img2(:)));
img3 = [img3 - min(img3(:))] ./ max(img3(:) - min(img3(:)));
img4 = [img4 - min(img4(:))] ./ max(img4(:) - min(img4(:)));

figure; subplot(2,2,1); imshow(img); title('Original Image');
subplot(2,2,2); imshow(img2); title('Image after Gamma Transform,
       \gamma = 4');
subplot(2,2,3); imshow(img3); title('Image after Gamma Transform,
       \gamma = 5');
subplot(2,2,4); imshow(img4); title('Image after Gamma Transform,
       \gamma = 6');
```

Original Image

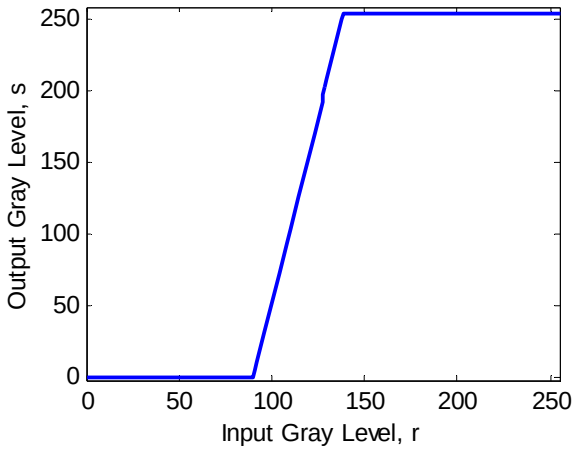
Image after Gamma Transform, $\gamma = 4$

Image after Gamma Transform, $\gamma = 5$

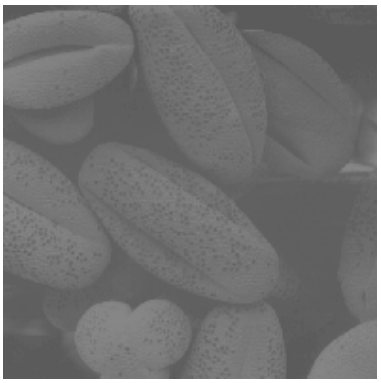
Image after Gamma Transform, $\gamma = 6$

Text figure (3.10) Contrast Stretching

```
%Matlab Code
clear all; close all;
img  = imread('pollen.jpg');
rmin = min(img(:)); rmax = max(img(:));
r = 0:255;
s = zeros(size(r));
s(1:find(s==rmin)) = 0;
step = length(r(find(r==rmin):find(r==rmax)));
s(find(r==rmin):find(r==rmax)) = 0:255./step:255-255./step;
s(find(r==rmax)+1:end) = 255;
img2 = double(img); img2 = [img2 - min(img2(:))] ./ max(img2(:) –
       min(img2(:)));
immean = round(mean(img(:))); img3 = img;
img3(find(img>=immean)) = 255; img3(find(img< immean)) = 0;

figure;
subplot(2,2,1); plot(r,s); axis([0 255 -2 259]);
xlabel('Input Gray Level, r'); ylabel('Output Gray Level, s');
subplot(2,2,2); imshow(img); title('Original Image');
subplot(2,2,3); imshow(img2); title('Contrast-Stretched Image');
subplot(2,2,4); imshow(img3); title('Thresholded Image');
```



Original Image

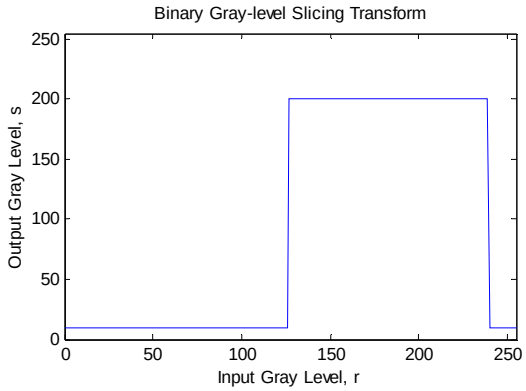


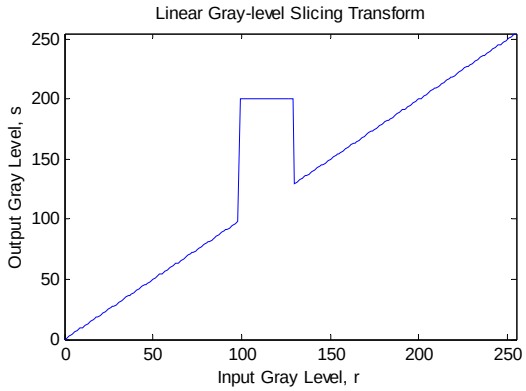
Contrast-Stretched Image

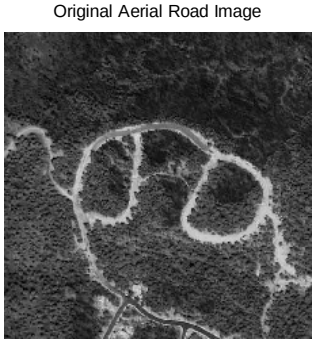
Thresholded Image

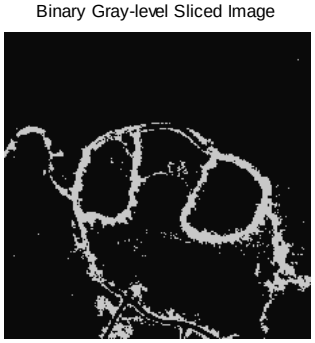
Text figure (3.11) Gray-level Slicing

```
%Matlab Code
clear all; close all;
img1 = rgb2gray(imread('road.jpg'));
img2 = img1;

x  = 0:255;
y1 = 10*ones(size(x));
y1(128:240) = 200;
y2 = x;
y2(100:130) = 200;

img2(find(img1>=128 & img1<=240)) = 200;
img2(find(img1<128)) = 10;
img2(find(img1>240)) = 200;

figure;
subplot(2,2,1); plot(x,y1); title('Binary Gray-level Slicing
Transform');
axis([0 255 0 255]); xlabel('Input Gray Level, r');
ylabel('Output Gray Level, s');
subplot(2,2,2); plot(x,y2); title('Linear Gray-level Slicing
Transform');
axis([0 255 0 255]); xlabel('Input Gray Level, r');
ylabel('Output Gray Level, s');
subplot(2,2,3); imshow(img1); title('Original Aerial Road Image');
subplot(2,2,4); imshow(img2); title('Binary Gray-level Sliced Image');
```





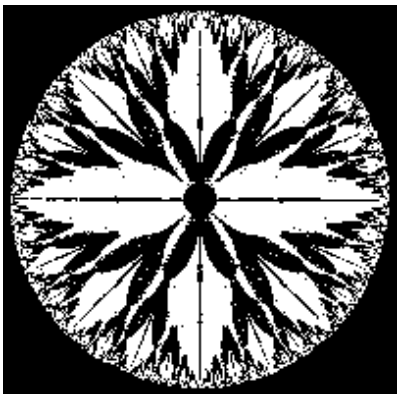




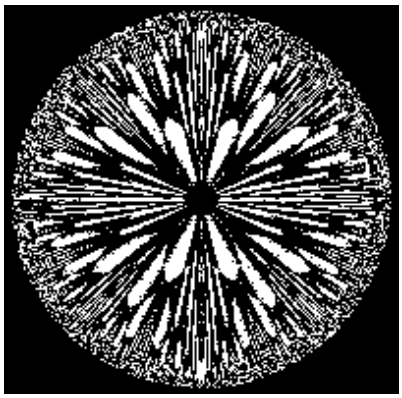
Text figure (3.14) Bit-Plane Slicing

```
%Matlab Code
clear all; close all;
img  = imread('fractal.jpg');
img7 = img; img6 = img; img5 = img; img4 = img;
img3 = img; img2 = img; img1 = img; img0 = img;

img7 = double(bitand(img,128)); img6 = double(bitand(img,64));
img5 = double(bitand(img,32));  img4 = double(bitand(img,16));
img3 = double(bitand(img,8));   img2 = double(bitand(img,4));
img1 = double(bitand(img,2));   img0 = double(bitand(img,1));

figure;
subplot(2,2,1); imshow(img7); title('Bit-plane 7');
subplot(2,2,2); imshow(img6); title('Bit-plane 6');
subplot(2,2,3); imshow(img5); title('Bit-plane 5');
subplot(2,2,4); imshow(img4); title('Bit-plane 4');

figure;
subplot(2,2,1); imshow(img3); title('Bit-plane 3');
subplot(2,2,2); imshow(img2); title('Bit-plane 2');
subplot(2,2,3); imshow(img1); title('Bit-plane 1');
subplot(2,2,4); imshow(img0); title('Bit-plane 0');
```

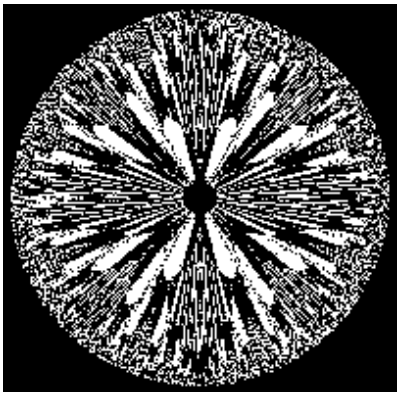
Bit-plane 7



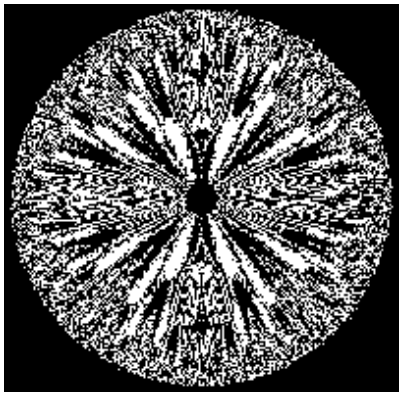
Bit-plane 6



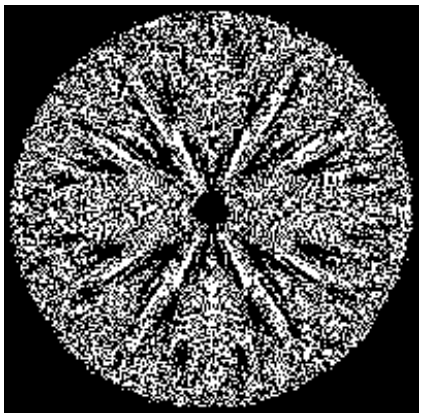
Bit-plane 5



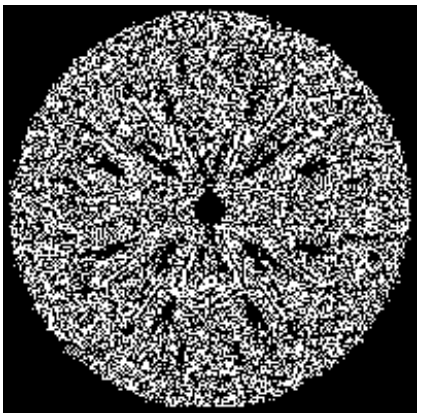
Bit-plane 4



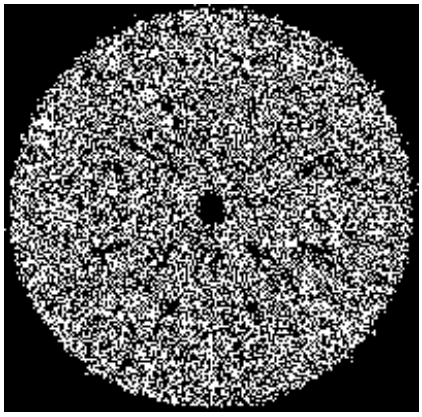
Bit-plane 3



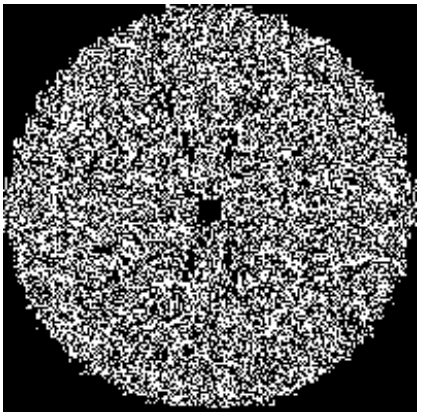
Bit-plane 2



Bit-plane 1

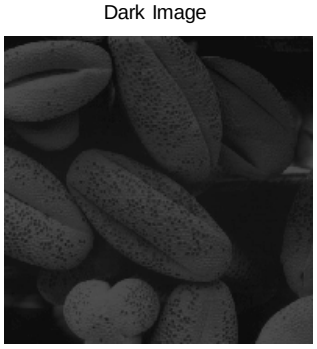


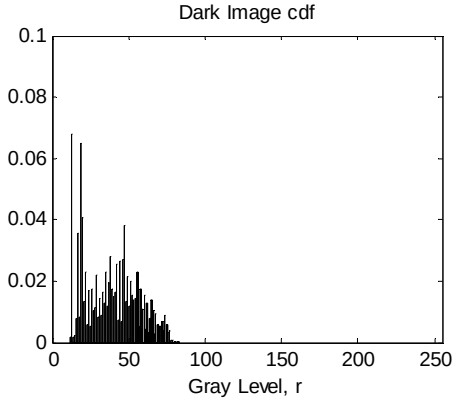
Bit-plane 0

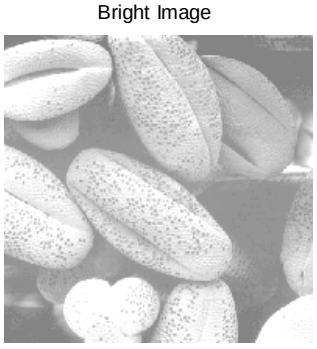


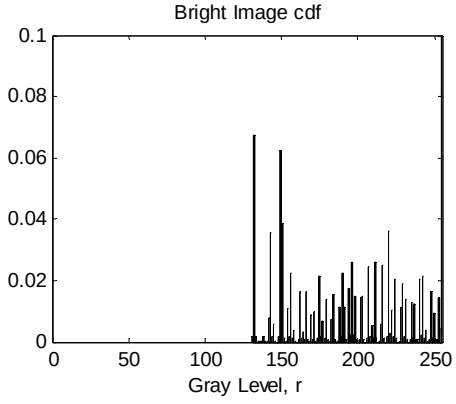
Text figure (3.15) Image Histograms

```
%Matlab Code
clear all; close all;
img1 = imread('pollen_dark.jpg'); img2 = imread('pollen_bright.jpg');
img3 = imread('pollen.jpg'); img4 = imread('pollen_highcontrast.jpg');
[hist1, bins1] = hist(double(img1(:)),256);
[hist2, bins2] = hist(double(img2(:)),256);
[hist3, bins3] = hist(double(img3(:)),256);
[hist4, bins4] = hist(double(img4(:)),256);
hist1 = hist1./length(img1(:)); hist2 = hist2./length(img2(:));
hist3 = hist3./length(img3(:)); hist4 = hist4./length(img4(:));

figure;
subplot(2,2,1); imshow(img1); title('Dark Image');
subplot(2,2,2); bar(bins1,hist1); title('Dark Image cdf');
axis([0 255 0 .1]); xlabel('Gray Level, r');
subplot(2,2,3); imshow(img2); title('Bright Image');
subplot(2,2,4); bar(bins2,hist2); title('Bright Image cdf');
axis([0 255 0 .1]); xlabel('Gray Level, r');
figure;
subplot(2,2,1); imshow(img3); title('Low-contrast Image');
subplot(2,2,2); bar(bins3,hist3); title('Low-contrast Image cdf');
axis([0 255 0 .1]); xlabel('Gray Level, r');
subplot(2,2,3); imshow(img4); title('High-contrast Image');
subplot(2,2,4); bar(bins4,hist4); title('High-contrast Image cdf');
axis([0 255 0 .1]); xlabel('Gray Level, r');
```

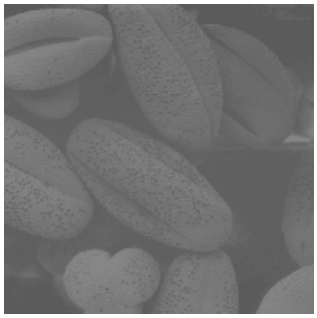




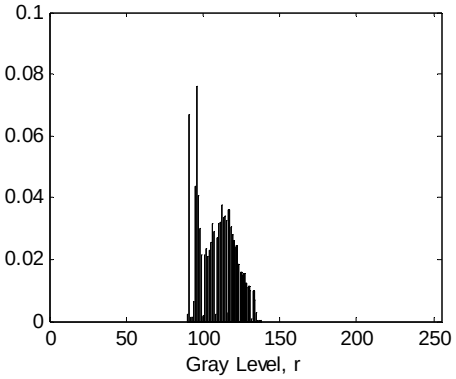




Low-contrast Image

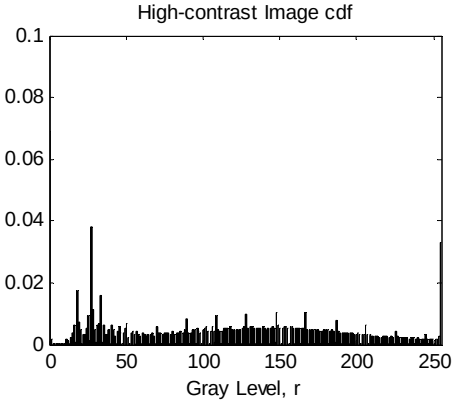


Low-contrast Image cdf



High-contrast Image

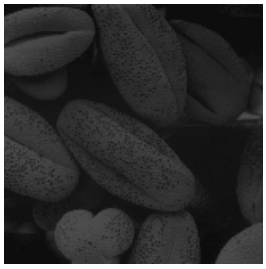
High-contrast Image cdf



Text figure (3.17) Histogram Equalization (Uniform pdf)

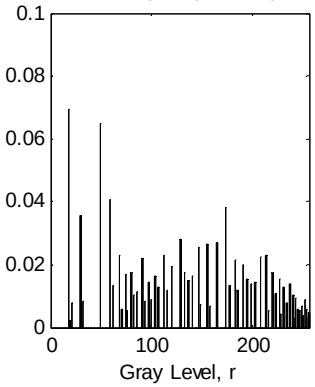
```
%Matlab Code
clear all; close all;
img1 = imread('pollen_dark.jpg'); img2 = imread('pollen_bright.jpg');
img3 = imread('pollen.jpg'); img4 = imread('pollen_highcontrast.jpg');

[hist1, bins1] = hist(double(img1(:)),0:255);
[hist2, bins2] = hist(double(img2(:)),0:255);
[hist3, bins3] = hist(double(img3(:)),0:255);
[hist4, bins4] = hist(double(img4(:)),0:255);
hist1 = hist1./length(img1(:)); hist2 = hist2./length(img2(:));
hist3 = hist3./length(img3(:)); hist4 = hist4./length(img4(:));

CDF1 = cumsum(hist1); CDF2 = cumsum(hist2);
CDF3 = cumsum(hist3); CDF4 = cumsum(hist4);

img1eq = zeros(size(img1)); img2eq = zeros(size(img2));
img3eq = zeros(size(img3)); img4eq = zeros(size(img4));
for i=0:255
    img1eq(find(img1==i)) = CDF1(i+1);
    img2eq(find(img2==i)) = CDF2(i+1);
    img3eq(find(img3==i)) = CDF3(i+1);
    img4eq(find(img4==i)) = CDF4(i+1);
end

[hist1eq, bins1eq] = hist(255*double(img1eq(:)),0:255);
[hist2eq, bins2eq] = hist(255*double(img2eq(:)),0:255);
[hist3eq, bins3eq] = hist(255*double(img3eq(:)),0:255);
[hist4eq, bins4eq] = hist(255*double(img4eq(:)),0:255);
hist1eq = hist1eq./length(img1eq(:)); hist2eq =
       hist2eq./length(img2eq(:));
hist3eq = hist3eq./length(img3eq(:)); hist4eq =
       hist4eq./length(img4eq(:));

figure;
subplot(2,3,1); imshow(img1); title('Dark Image');
subplot(2,3,2); imshow(img1eq); title('Dark Image Equalized');
subplot(2,3,3); bar(bins1eq,hist1eq); title('Dark Image Equalized
       pdf');
axis([0 255 0 .1]); xlabel('Gray Level, r');
subplot(2,3,4); imshow(img2); title('Bright Image');
subplot(2,3,5); imshow(img2eq); title('Bright Image Equalized');
subplot(2,3,6); bar(bins2eq,hist2eq); title('Bright Image Equalized
       pdf');
axis([0 255 0 .1]); xlabel('Gray Level, r');

figure;
subplot(2,3,1); imshow(img3); title('Low-contrast Image');
subplot(2,3,2); imshow(img3eq); title('Low-contrast Image Equalized');
subplot(2,3,3); bar(bins3eq,hist3eq); title('Low-contrast Image
       Equalized pdf');
axis([0 255 0 .1]); xlabel('Gray Level, r');
subplot(2,3,4); imshow(img4); title('High-contrast Image');
subplot(2,3,5); imshow(img4eq); title('High-contrast Image Equalized');
subplot(2,3,6); bar(bins4eq,hist4eq); title('High-contrast Image
       Equalized pdf');
axis([0 255 0 .1]); xlabel('Gray Level, r');
```

Dark Image

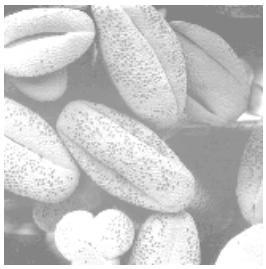


Dark Image Equalized

Dark Image Equalized pdf

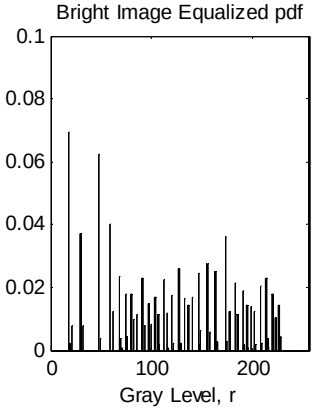


Bright Image

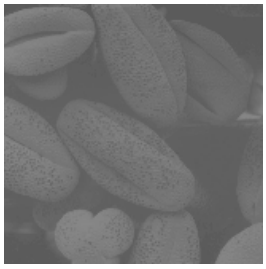


Bright Image Equalized

Bright Image Equalized pdf

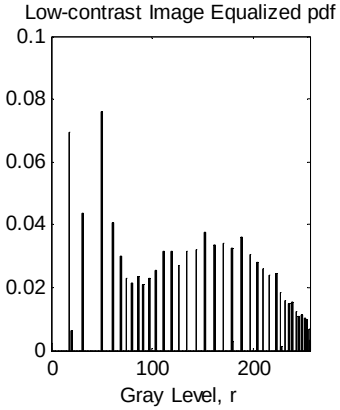


Low-contrast Image



Low-contrast Image Equalized

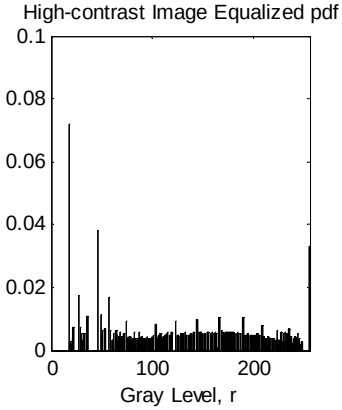
Low-contrast Image Equalized pdf



High-contrast Image

High-contrast Image Equalized

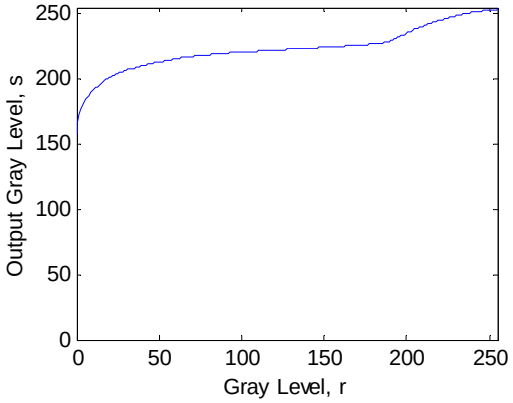
High-contrast Image Equalized pdf



Text figure (3.21) Histogram Equalization

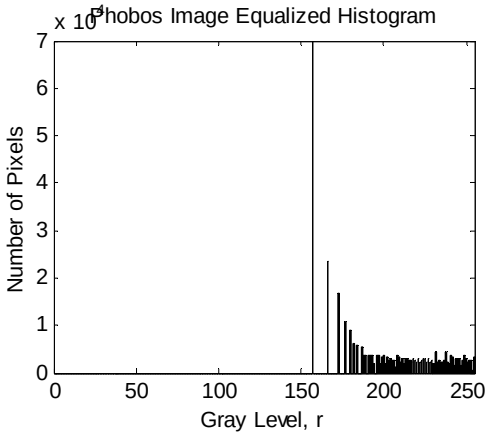
```
%Matlab Code
clear all; close all;
img1 = imread('phobos.jpg');

[hist1, bins1] = hist(double(img1(:)),0:255);
hist1 = hist1./length(img1(:));
CDF1 = cumsum(hist1);
img1eq = zeros(size(img1));
for i=0:255
    img1eq(find(img1==i)) = CDF1(i+1);
end
[hist1eq, bins1eq] = hist(255*double(img1eq(:)),0:255);
hist1eq = hist1eq;

figure;
subplot(2,2,1); plot(bins1, 255*CDF1); title('Phobos Image
       Transformation Function');
axis([0 255 0 255]); xlabel('Gray Level, r'); ylabel('Output Gray
       Level, s');
subplot(2,2,2); imshow(img1eq); title('Phobos Image Equalized');
subplot(2,2,3); bar(bins1eq,hist1eq); title('Phobos Image Equalized
       Histogram');
axis([0 255 0 70000]); xlabel('Gray Level, r'); ylabel('Number of
       Pixels');
```

Phobos Image Transformation Function



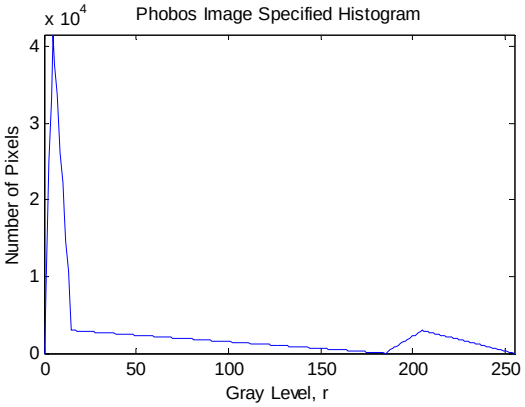
Phobos Image Equalized

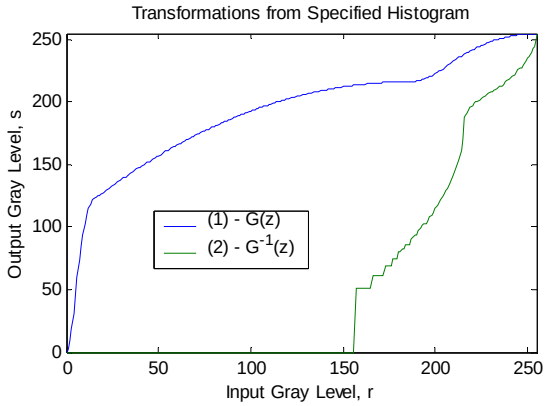
Phobos Image Equalized Histogram



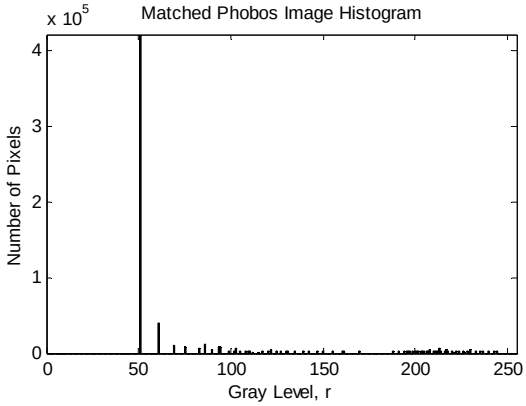
Text figure (3.22) Histogram Matching

```
%Matlab Code
clear all; close all;
img1 = imread('phobos.jpg');

%Transform to Uniform Distribution
[hist1, bins1] = hist(double(img1(:)),0:255);
hist1 = hist1./length(img1(:));
T     = cumsum(hist1);
img1eq = zeros(size(img1));
for i=0:255
    img1eq(find(img1==i)) = T(i+1);
end
[hist1eq, bins1eq] = hist(double(255*img1eq(:)),0:255);
hist1eq = hist1eq./length(img1(:)); S = cumsum(hist1eq);

%Specify New Histogram
x = 0:255; y = 1:256;
y(1:6)     = 0:70000/(6-1):70000; y(6:16) = 70000:(5000-70000)/(16-6):5000;
y(16:186)  = 5000:(-5000)/(186-16):0; y(186:206) = 0:5000/(206-186):5000;
y(206:256) = 5000:(-5000)/(256-206):0;
hsum = sum(y); y = y./hsum; y = y*length(img1(:));

%Compute New CDF's from Specified Histogram (Iterative)
G = (cumsum(y)/length(img1(:)));
Ginv = zeros(size(G));
for k=1:256
    dff = -1; z = 0;
    while(dff < 0)
        z = z+1; dff = G(z) - S(k);
    end
    Ginv(k) = z-1;
end

img1mt = zeros(size(img1eq)); ieq    = floor(255*img1eq);
for i=0:255
    img1mt(find(ieq==i)) = Ginv(i+1);
end

[hist1mt, bins1mt] = hist(double(img1mt(:)),0:255);
img1mt = img1mt/255;

figure;
subplot(2,2,1); plot(x,y); title('Phobos Image Specified Histogram');
axis([0 255 0 max(y)]); xlabel('Gray Level, r'); ylabel('Number of Pixels');
subplot(2,2,2); plot(x,255*G,x,Ginv); title('Transformations from Specified
        Histogram');
legend('(1) - G(z)','(2) - G^{-1}(z)'); axis([0 255 0 255]);
xlabel('Input Gray Level, r'); ylabel('Output Gray Level, s');
subplot(2,2,3); imshow(img1mt); title('Histogram Matched Phobos Image');
subplot(2,2,4); bar(bins1mt,hist1mt); title('Matched Phobos Image Histogram');
axis([0 255 0 max(hist1mt)]); xlabel('Gray Level, r'); ylabel('Number of
        Pixels');
```





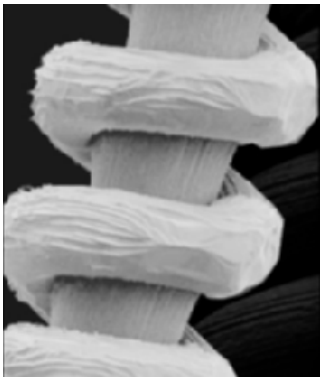
Histogram Matched Phobos Image



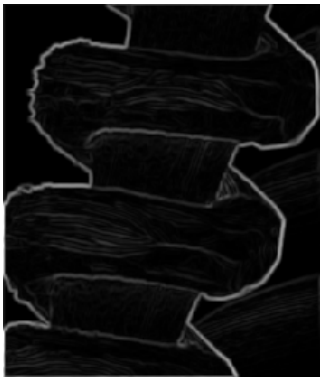
Text figure (3.25) Histogram Statistics Enhancement

```
%Matlab Code
clear all; close all;
img = im2double(imread('filament.jpg'));
M = mean(img(:)); D = sqrt(var(img(:)));
E = 4; k0 = 0.4; k1=0.02; k2=0.4;

%Compute Local Means
h = ones(3,3)/9;
img1 = conv2(img,h,'same');

%Compute Local Standard Deviations
imgb = zeros(size(img1,1)+2, size(img1,2)+2);
imgb(2:size(imgb,1)-1, 2:size(imgb,2)-1) = img1;
img2 = zeros(size(img1));
block = zeros(1,9);
for i=1:size(img,1)
    for j=1:size(img,2)
        block(:) = imgb(i:i+2,j:j+2);
        img2(i,j) = sqrt(var(block));
    end
end

%Compute Multiplication Mask
imgmean = zeros(size(img1));
imgsdev = zeros(size(img2));
imgmean(find(img1<=k0*M)) = 1;
imgsdev(find(img2>=k1*D & img2<=k2*D)) = 1;
img3=E*imgmean.*imgsdev;

img2 = [img2 - min(img2(:))] ./ max(img2(:) - min(img2(:)));
img2 = [img2 - min(img2(:))] ./ max(img2(:) - min(img2(:)));
img3 = [img3 - min(img3(:))] ./ max(img3(:) - min(img3(:)));
figure;
subplot(1,3,1); imshow(img1); title('Local 3x3 Mean Image');
subplot(1,3,2); imshow(img2); title('Local 3x3 Std. Deviation Image');
subplot(1,3,3); imshow(img3); title('Multiplication Mask');
```

Local 3x3 Mean Image



Local 3x3 Std. Deviation Image

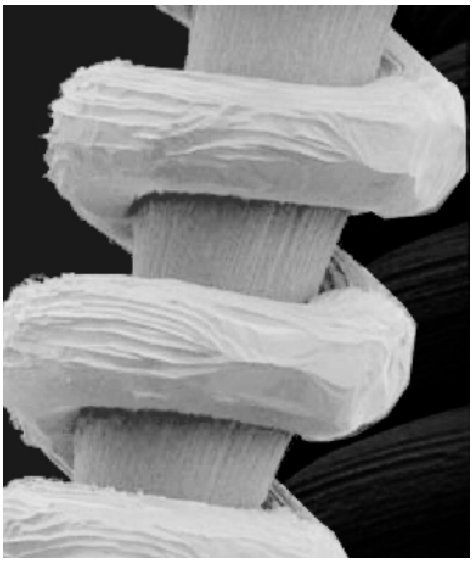


Multiplication Mask

Text figure (3.26) Histogram Statistics Enhancement

```
%Matlab Code
clear all; close all;
img = im2double(imread('filament.jpg'));
M = mean(img(:)); D = sqrt(var(img(:)));
E = 4; k0 = 0.4; k1=0.02; k2=0.4;

%Compute Local Means
h = ones(3,3)/9; img1 = conv2(img,h,'same');

%Compute Local Standard Deviations
imgb = zeros(size(img1,1)+2, size(img1,2)+2);
imgb(2:size(imgb,1)-1, 2:size(imgb,2)-1) = img1;
img2 = zeros(size(img1));
block = zeros(1,9);
for i=1:size(img,1)
    for j=1:size(img,2)
        block(:) = imgb(i:i+2,j:j+2);
        img2(i,j) = sqrt(var(block));
    end
end

%Compute Multiplication Mask
imgmean = zeros(size(img1)); imgsdev = zeros(size(img2));
imgmean(find(img1<=k0*M)) = 1;
imgsdev(find(img2>=k1*D & img2<=k2*D)) = 1;
img3=E*imgmean.*imgsdev;
img3(find(img3<E))=1;

%Enhance Image
img4 = img.*img3;

figure;
subplot(1,2,1); imshow(img); title('Original Filament Image');
subplot(1,2,2); imshow(img4); title('Enhanced Filament Image');
```

Original Filament Image



Enhanced Filament Image

Image Subtraction Enhancement

These are two consecutive images from a video sequence with the camera fixed while there is traffic moving across the scene. The two images were subtracted, and the error image thresholded to create a mask image that contains significant differences. The false-color of these differences are marked in red. Notice in the original images the van moving across the scene, which has been successfully detected.

```
%Matlab Code
clear all; close all;
imga = im2double(imread('scene1.jpg'));
imgb = im2double(imread('scene2.jpg'));
img1 = imga(10:size(imga,1)-9, 10:size(imgb,2)-9);
img2 = imgb(10:size(imga,1)-9, 10:size(imgb,2)-9);

idff = abs(img1 - img2); idx  = find(idff>.2);
mask = zeros(size(idff)); mask(idx) = 1;
red = img1; green = img1; blue = img1;
red(idx) = 1; green(idx) = 0; blue(idx) = 0;
img3 = cat(3, cat(3,red,green), blue);

figure;
subplot(2,2,1); imshow(img1); title('Scene Image 1');
subplot(2,2,2); imshow(img2); title('Scene Image 2');
```

```
subplot(2,2,3); imshow(mask); title('Difference Mask');
subplot(2,2,4); imshow(img3); title('Image With Detected Motion (False-
        Colored)');
```

Scene Image 1

Scene Image 2

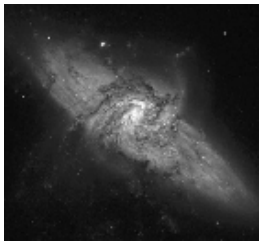
Difference Mask

Image With Detected Motion (False-Colored)

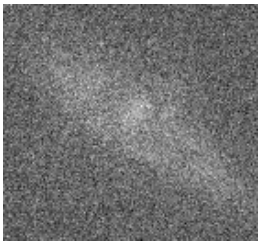
Text figure (3.30) Image Averaging

```
%Matlab Code
clear all; close all;
img1 = double(imread('galaxy.jpg'));
icor = img1 + 64*randn(size(img1,1), size(img1,2));
avg8 = zeros(size(img1,1), size(img1,2)); avg16 = avg8;
avg64 = avg8; avg128 = avg8;
%I did averaging in this way because of memory constraints.
for i=1:8
    avg8 = avg8 + img1 + 64*randn(size(img1,1), size(img1,2));
end
for i=1:16
    avg16 = avg16 + img1 + 64*randn(size(img1,1), size(img1,2));
end
for i=1:64
    avg64 = avg64 + img1 + 64*randn(size(img1,1), size(img1,2));
end
for i=1:128
    avg128 = avg128 + img1 + 64*randn(size(img1,1), size(img1,2));
end
avg8 = avg8/8; avg16 = avg16/16; avg64 = avg64/64; avg128 = avg128/128;

img1    = [img1   - min(img1(:))]   ./ max(img1(:)   - min(img1(:)));
icor    = [icor   - min(icor(:))]   ./ max(icor(:)   - min(icor(:)));
avg8    = [avg8   - min(avg8(:))]   ./ max(avg8(:)   - min(avg8(:)));
avg16   = [avg16  - min(avg16(:))]  ./ max(avg16(:)  - min(avg16(:)));
avg64   = [avg64  - min(avg64(:))]  ./ max(avg64(:)  - min(avg64(:)));
avg128  = [avg128 - min(avg128(:))] ./ max(avg128(:) - min(avg128(:)));

figure;
subplot(3,2,1); imshow(img1); title('Original Galaxy Image');
subplot(3,2,2); imshow(icor); title('Corrupted Galaxy Image w/ Gaussian
       Noise');
subplot(3,2,3); imshow(avg8); title('Galaxy Image After Averaging 8
       Corrupted Images');
subplot(3,2,4); imshow(avg16); title('Galaxy Image After Averaging 16
       Corrupted Images');
subplot(3,2,5); imshow(avg64); title('Galaxy Image After Averaging 64
       Corrupted Images');
subplot(3,2,6); imshow(avg128); title('Galaxy Image After Averaging 128
       Corrupted Images');
```

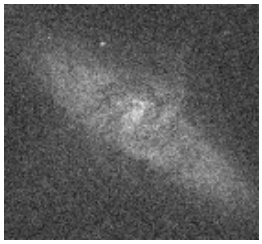
Original Galaxy Image



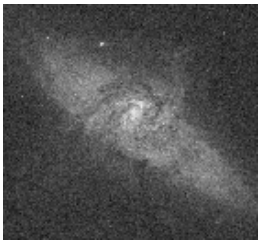
Corrupted Galaxy Image w/ Gaussian Noise



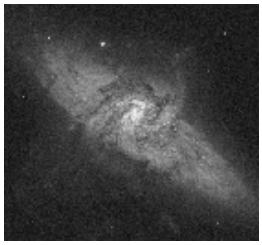
Galaxy Image After Averaging 8 Corrupted Images



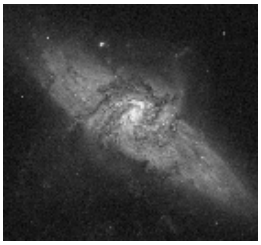
Galaxy Image After Averaging 16 Corrupted Images



Galaxy Image After Averaging 64 Corrupted Images



Galaxy Image After Averaging 128 Corrupted Images

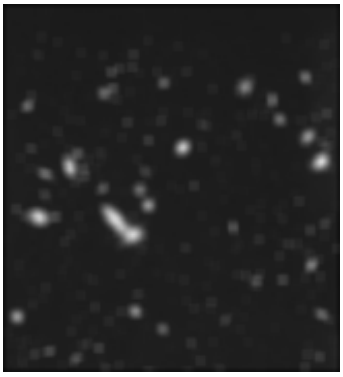


Text figure (3.36) Smoothing Linear Filters

```
%Matlab Code
clear all; close all;
img1 = imread('hubble.jpg');
img2 = im2double(img1);
h    = ones(15,15)/(15^2);
img2 = conv2(img2,h,'same');
th   = 0.25*max(img2(:));
img3 = img2;
img3(find(img2>=th)) = 1;
img3(find(img2<th))  = 0;

figure;
subplot(1,3,1); imshow(img1); title('Original Hubble Image');
subplot(1,3,2); imshow(img2); title('Image after 15x15 Smoothing
       Filter');
subplot(1,3,3); imshow(img3); title('Thresholded Smoothed Hubble
       Image');
```

Original Hubble Image

Image after 15x15 Smoothing Filter



Thresholded Smoothed Hubble Image

Text figure (3.37) Median Filtering

```
%Matlab Code
clear all; close all;
img1 = imread('circuitboard.jpg');
img2 = im2double(img1);
h    = ones(3,3)/(3^2);
img2 = conv2(img2,h,'same');
img3 = medfilt2(im2double(img1),[3 3]);

figure;
subplot(1,3,1); imshow(img1); title('Original Circuit Board Image');
subplot(1,3,2); imshow(img2); title('Image after 3x3 Smoothing
Filter');
subplot(1,3,3); imshow(img3); title('Image after 3x3 Median Filter');
```

Original Circuit Board Image

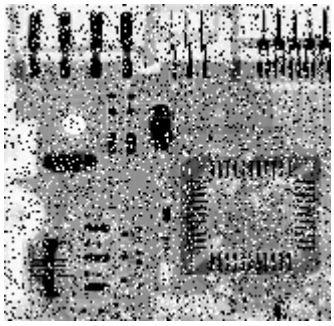


Image after 3x3 Smoothing Filter

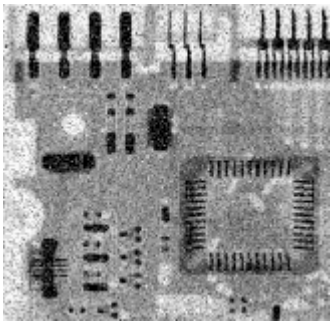
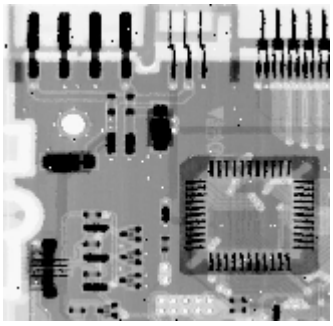


Image after 3x3 Median Filter

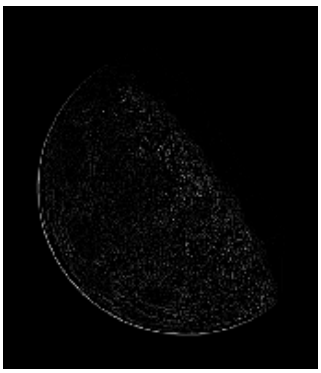


Text figure (3.40) Laplacian Filtering

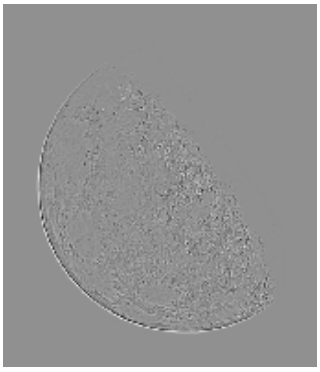
```
%Matlab Code
clear all; close all;
img1 = im2double(imread('moon.jpg'));
lap  = [1 1 1; 1 -8 1; 1 1 1;];
img2 = conv2(img1, lap, 'same');
img3 = [img2 - min(img2(:))] ./ max(img2(:) - min(img2(:)));
img4 = img1 - img2;

figure;
subplot(2,2,1); imshow(img1); title('Moon Image');
subplot(2,2,2); imshow(img2); title('Laplacian Filtered Moon Image');
subplot(2,2,3); imshow(img3); title('Laplacian Image Scaled to Full
       Dynamic Range');
subplot(2,2,4); imshow(img4); title('Laplacian-enhanced Original
       Image');
```

Moon Image

Laplacian Filtered Moon Image



Laplacian Image Scaled to Full Dynamic Range

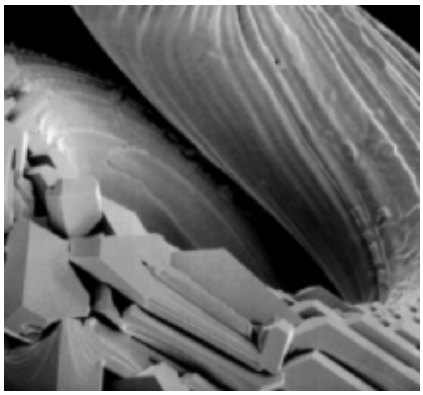


Laplacian-enhanced Original Image

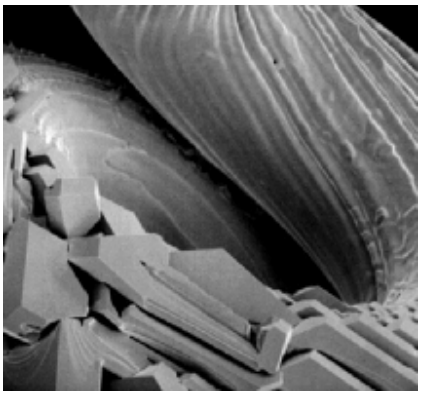
Text figure (3.41) Laplacian Filtering Comparison

```
%Matlab Code
clear all; close all;
img1 = im2double(imread('SEM.jpg'));
lapa  = [0 -1 0; -1 5 -1; 0 -1 0;];
lapb  = [-1 -1 -1; -1 9 -1; -1 -1 -1;];
img2 = conv2(img1, lapa, 'same');
img3 = conv2(img1, lapb, 'same');

figure;
subplot(2,2,1); imshow(img1); title('SEM Image');
subplot(2,2,2); imshow(img2); title('Composite 4-neighbor Laplacian
       Image');
subplot(2,2,3); imshow(img3); title('Composite 8-neighbor Laplacian
       Image');
```

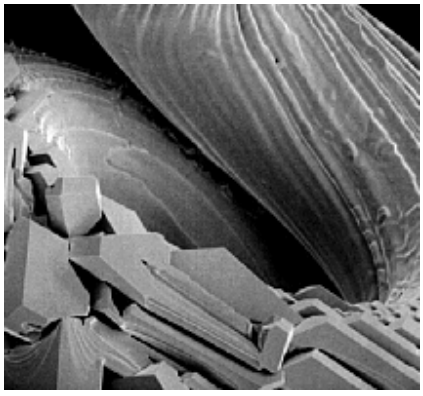
SEM Image



Composite 4-neighbor Laplacian Image



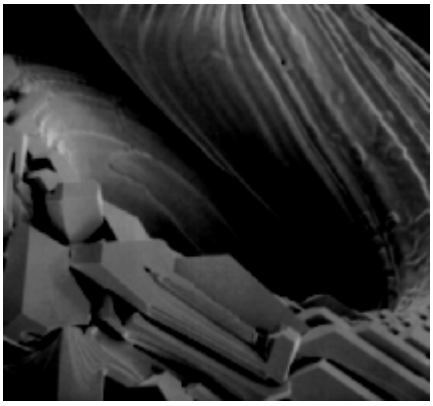
Composite 8-neighbor Laplacian Image



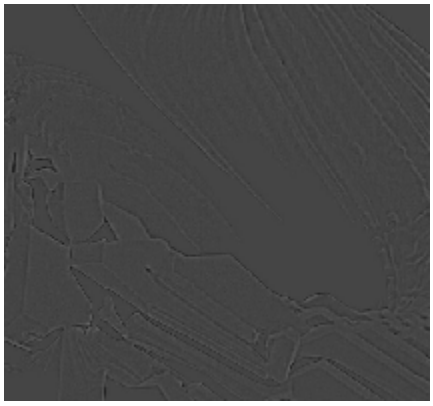
Text figure (3.43) Laplacian Enhancement

```
%Matlab Code
clear all; close all;
img1 = im2double(imread('SEM_dark.jpg'));
lapa  = [-1 -1 -1; -1 8 -1; -1 -1 -1;];
lapb  = [-1 -1 -1; -1 9 -1; -1 -1 -1;];
lapc  = [-1 -1 -1; -1 9.7 -1; -1 -1 -1;];
img2 = conv2(img1, lapa, 'same');
img2 = [img2 - min(img2(:))] ./ max(img2(:) - min(img2(:)));
img3 = conv2(img1, lapb, 'same');
img4 = conv2(img1, lapc, 'same');

figure;
subplot(2,2,1); imshow(img1); title('SEM Image');
subplot(2,2,2); imshow(img2); title('Laplacian Image');
subplot(2,2,3); imshow(img3); title('Laplacian Enhanced Image');
subplot(2,2,4); imshow(img4); title('High Boost Laplacian Enhanced
       Image');
```

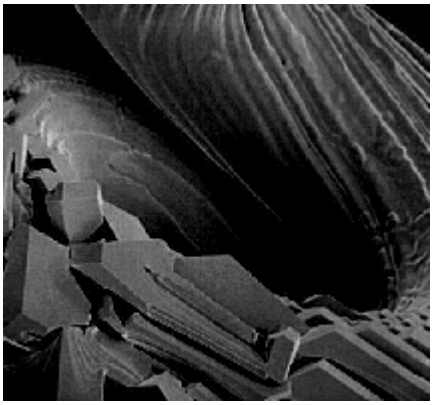
SEM Image



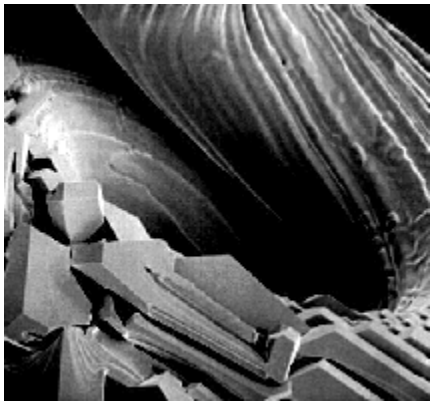
Laplacian Image



Laplacian Enhanced Image



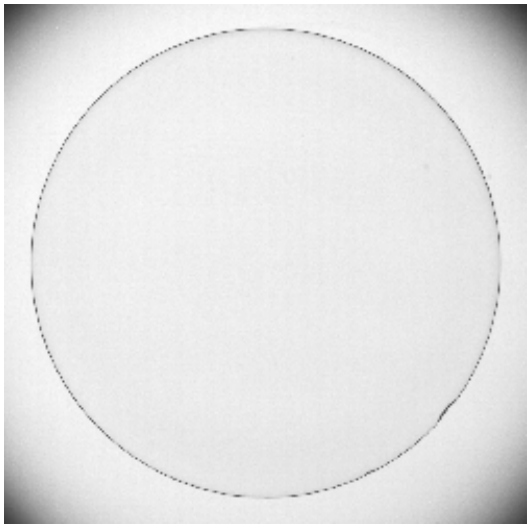
High Boost Laplacian Enhanced Image



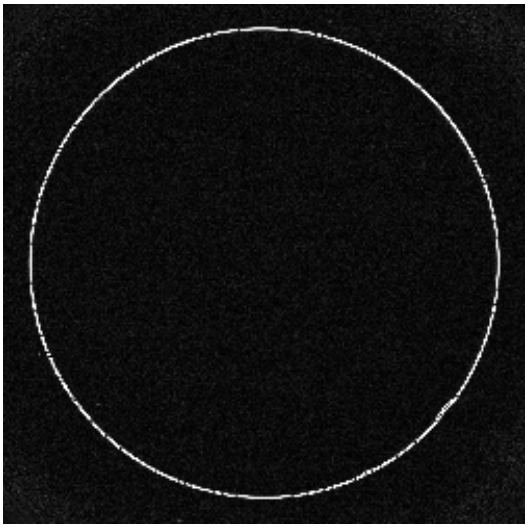
Text figure (3.45) Sobel Gradient Filtering

```
%Matlab Code
clear all; close all;
img1 = im2double(imread('contact.jpg'));
soba = [-1 -2 -1; 0 0 0; 1 2 1];
sobb = [-1 0 1; -2 0 2; -1 0 1];
imga = abs(conv2(img1, soba, 'same'));
imgb = abs(conv2(img1, sobb, 'same'));
img2 = imga + imgb;

figure;
subplot(1,2,1); imshow(img1); title('Contact Lens Image');
subplot(1,2,2); imshow(img2); title('Sobel Image');
```

Contact Lens Image



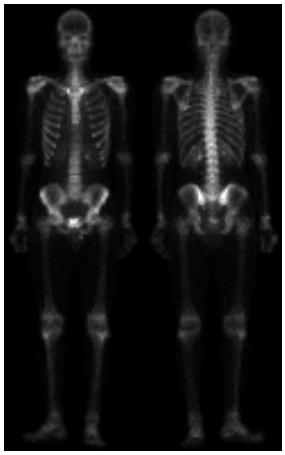
Sobel Image



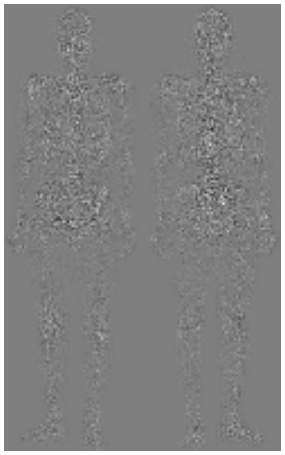
Text figure (3.46) Combining Spatial Enhancements

```
%Matlab Code
clear all; close all;
lap  = [-1 -1 -1; -1 8 -1; -1 -1 -1];
soba = [-1 -2 -1; 0 0 0; 1 2 1];
sobb = [-1 0 1; -2 0 2; -1 0 1];
h    = ones(5,5)/25;

img1 = im2double(imread('bonescan.jpg'));
img2 = conv2(img1, lap, 'same');
img3 = img1 + img2;
img4 = abs(conv2(img1, soba, 'same')) + abs(conv2(img1, sobb, 'same'));
img5 = conv2(img4, h, 'same');
img6 = img3.*img5;
img7 = img1 + img6;
img8 = img7.^.5;
img2 = [img2 - min(img2(:))] ./ max(img2(:) - min(img2(:)));

figure;
subplot(2,2,1); imshow(img1); title('Bone Scan Image');
subplot(2,2,2); imshow(img2); title('Laplacian Image');
subplot(2,2,3); imshow(img3); title('Enhanced Laplacian Image');
subplot(2,2,4); imshow(img4); title('Sobel Image');
figure;
subplot(2,2,1); imshow(img5); title('Sobel 5x5 Smoothed');
subplot(2,2,2); imshow(img6); title('Mask Image');
subplot(2,2,3); imshow(img7); title('Sharpened Image');
subplot(2,2,4); imshow(img8); title('Power-law Image');
```

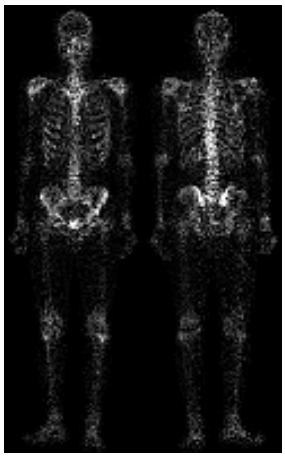
Bone Scan Image



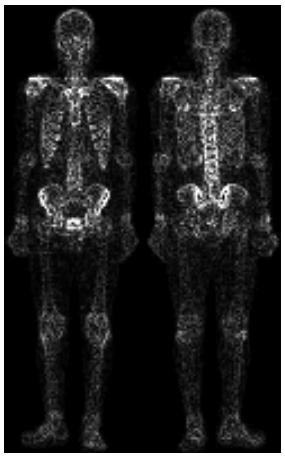
Laplacian Image



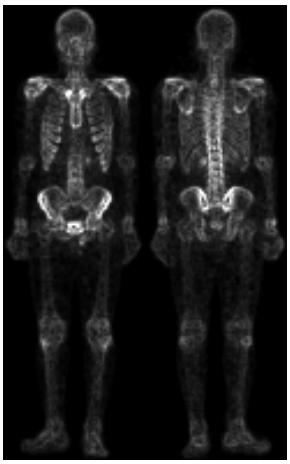
Enhanced Laplacian Image



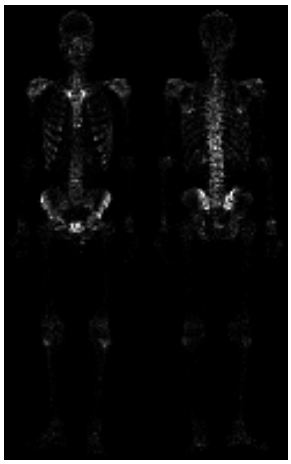
Sobel Image



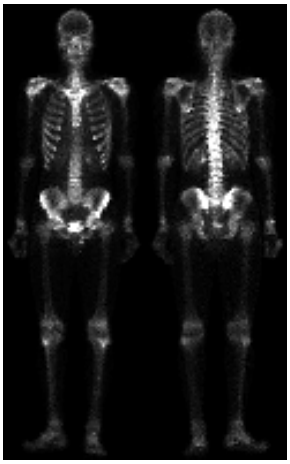
Sobel 5x5 Smoothed



Mask Image



Sharpened Image



Power-law Image