ECE280 - Lab 5 Image Processing 1

Sylvester Johannes Arizie (sna31)

November 2, 2024

I have adhered to the Duke Community Standard in completing this assignment.

Signature

Contents

1 Exercise 1 - Color blending

Increasing the value of p causes the RGB (circles of the Venn diagram) points to be sharper and more concentrated, thus the overall radius appears to be reduced for a larger value of p whereas for a small value of p, the image (circles of the Venn diagram) appears blurry and less sharp and the colors seem to radiate over a larger area thus a seemingly larger radius.

Component	k	$x_{c,k}$	$y_{c,k}$
Red	1	rad	0
Green	2	$rad \cdot \cos(\frac{2\pi}{3})$	$rad \cdot \sin(\frac{2\pi}{3})$
Blue	3	$rad \cdot \cos(\frac{4\pi}{3})$	$rad \cdot \sin(\frac{4\pi}{3})$

Table 1: Color combination

2 Exercise 2 – Random colors

The image produced is a pixelation of a series of randomly generated color points that appear to have equal distribution for each of RGB. With each run, a new image is produced due to the randomization in the generation of the array.

3 Exercise 3 – 1D moving average

The 5-point moving average appears to be smoother than the 2-point moving average since it uses more points in its approximation. Even still, the 5-point curve appears to be collapsed as compared to the blue original.

4 Exercise 4 – Derivative approximation

The plot with 51 appears to be more accurate relative to the plot generated using 11 points. The error may be due to the need to average approximation over larger distances when using 11 points whereas with 51 points, we have the freedom of making approximations that are close enough. As seen from the plots, with 51 pints, all of the points are properly aligned such that our approximations have negligible error, whereas with 11 points significant error is observed.

5 Exercise 5 – Boxes and Rectangles and Voids (Oh My!)

Of all three blurred images, the 'sharpest' is that which uses a 10x10 matrix in creating a blur. Between the remaining 2, the 50x2 blurred image appears to blur vertically(top-down) whilst the 2x50 blurred image appears to be horizontally blurred(from right to left).

6 Exercise 6 – Same Song, Different Verse, These Images Aren't Quite As Big As At First

The blur from both convolutions are significant. However, there seems to be sharper transition in image points in the blur generated using 'valid' as compared to 'same'. Additionally, with same , there seems to be a consistent blur across the whole image whilst with 'valid', the blur seems more localized without extending to the edges.

7 Exercise 7 – Fun With Convolution

1. 5x5 Gaussian filter as on graphs 9.7.
   The Gaussian filter appears to have a more natural looking/ dreamy effect on blurring the image. It has a really smooth look and this could be as a result of its tendency to take out higher frequencies and 'normalizing' remnants. It additionally has a sense of depth since the main elements of the image(coins) are in focus.

2. Prewitt vertical edge detection filter as on graphs 9.7.
   The vertical edges of the coin are sharpened and thus more distinct compared to the other components of the image.

3. Prewwit horizontal edge detection filter as on graphs 9.7.
   The horizontal edges of the coin are sharpened and thus more noticeable as compared to the rest of the image.

4. Prewitt edge detection filter as on graphs 9.7.
   The Prewitt edge detection highlights the edges of the coin all round and darkens the rest of the image, making it easy to see the coin in its completeness, whilst hiding its internal details.

5. Wikipedia edge detection filter as on graphs 9.7.
   This filter shows the coins edges all round whilst highlighting the details internally also. Overall, it appears to be an enhanced picture of the original as it highlights edges of the coin and makes it more visible while creating a sense of depth without losing or rather almost improving the internal details of the original image.

8 Exercise 8 – Putting It All Together

The box blur creates an image that is significantly blurred hiding all details behind a haze. The transitions are dull making it hard to identify distinctions in the image components. The Gaussian blur is not as blurred as the box blur but rather creates a sense of depth in the image while highlighting localized components of the image. Thus it shows a more smoother blur that is easier on the eyes. The Sobel vertical edge filter picks out high intensity vertical edges and highlights them out of the entire image, darkening other portions of the image. The Sobel horizontal edge filter picks out high intensity horizontal edges and highlights them out of the entire image, darkening other portions of the image. The last image, the result of normalizing both the horizontal and vertical Sobel filters picks out all the edges of the image and highlights it.

Figure 1: Original image of coins

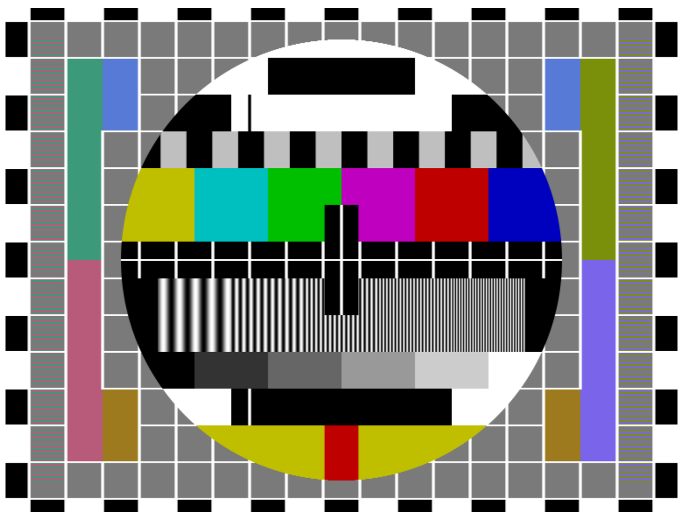
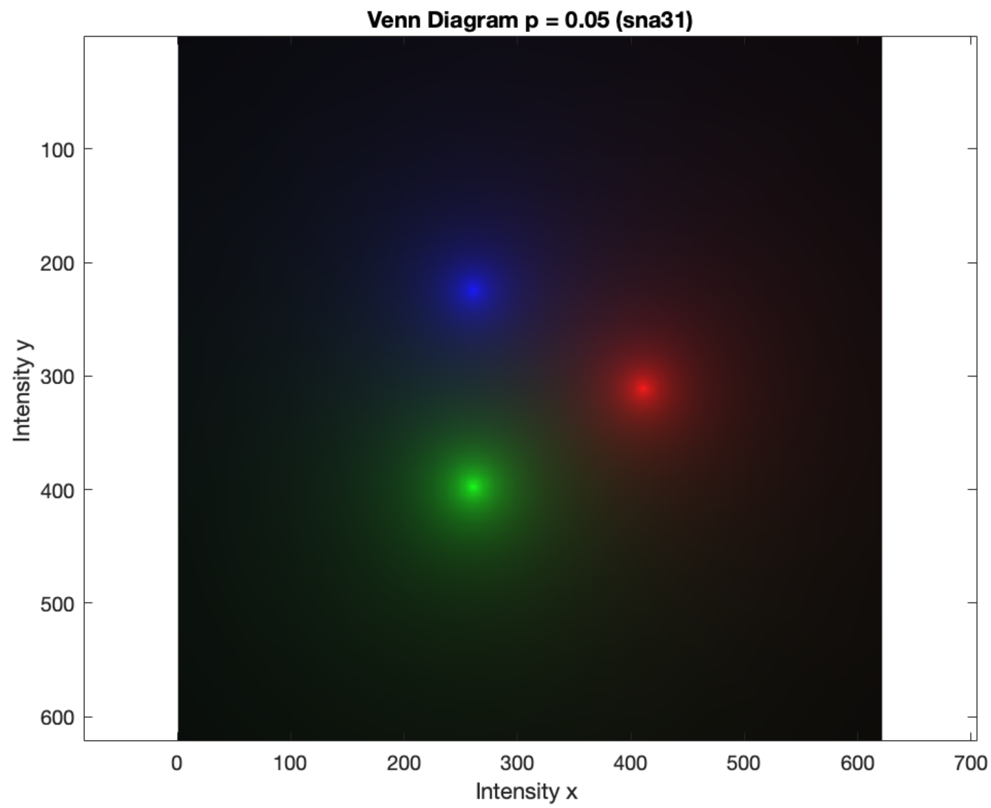
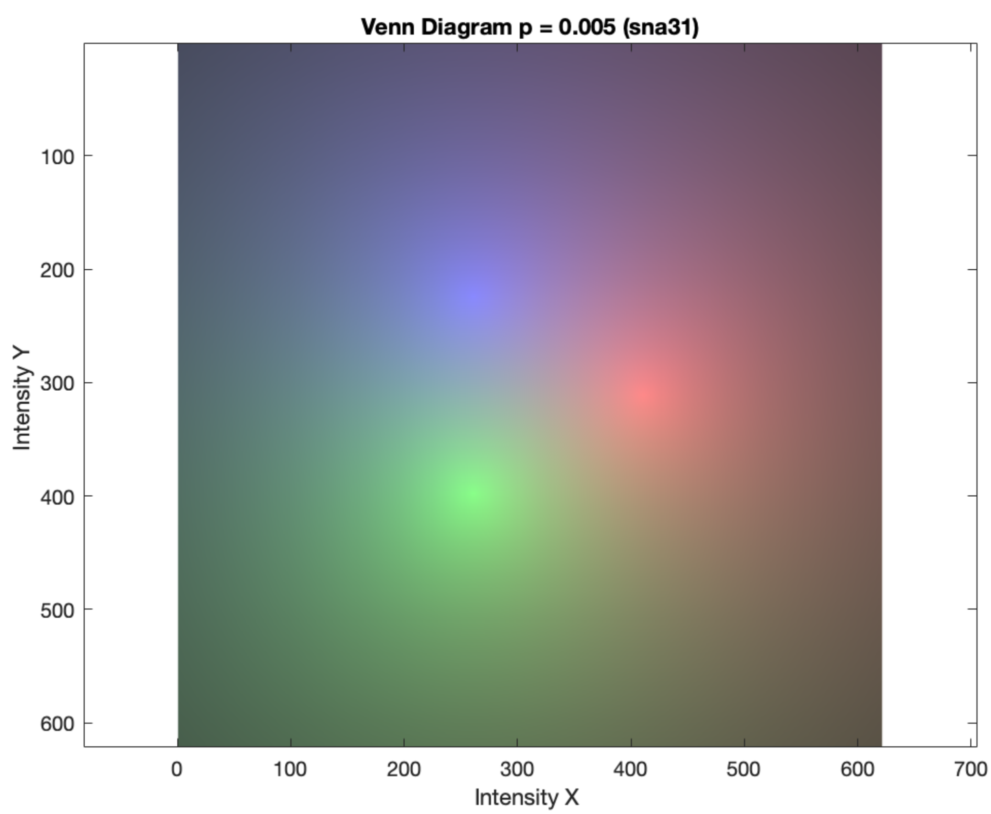


Figure 2: Original image of Test Card

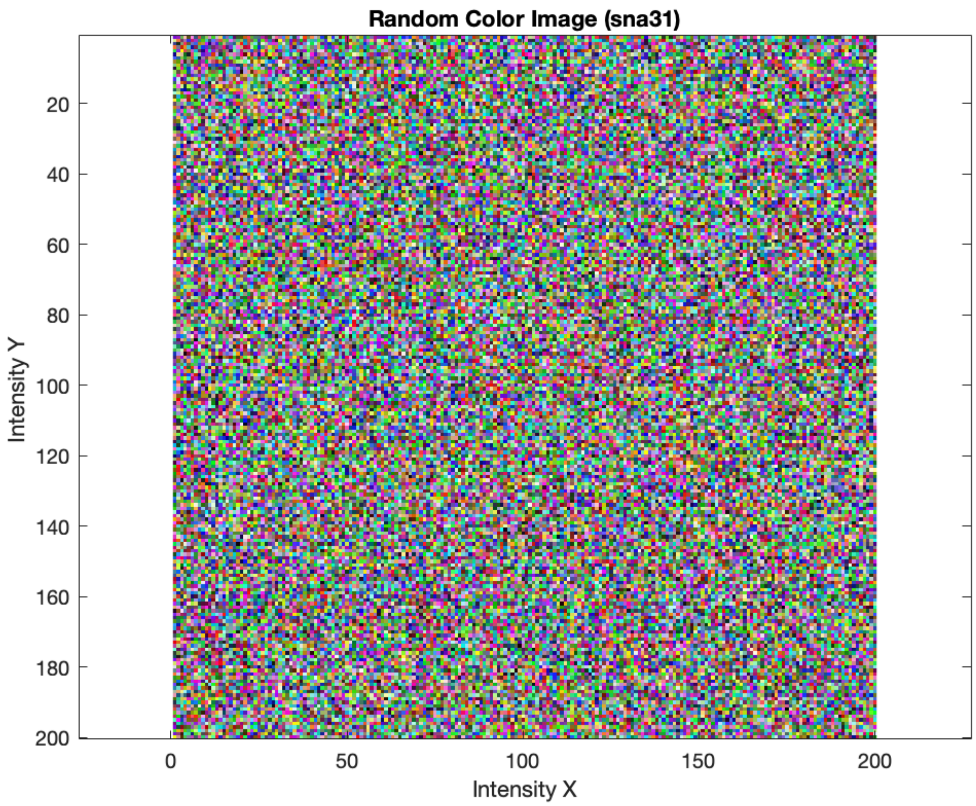
9 Graphs

9.1 Exercise 1

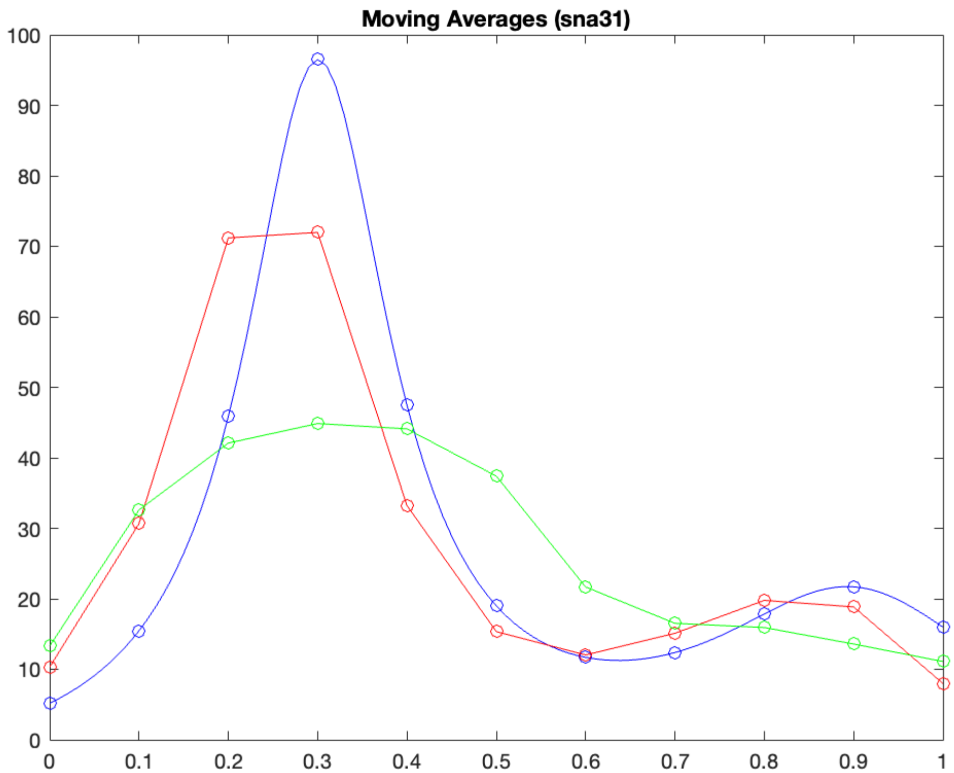




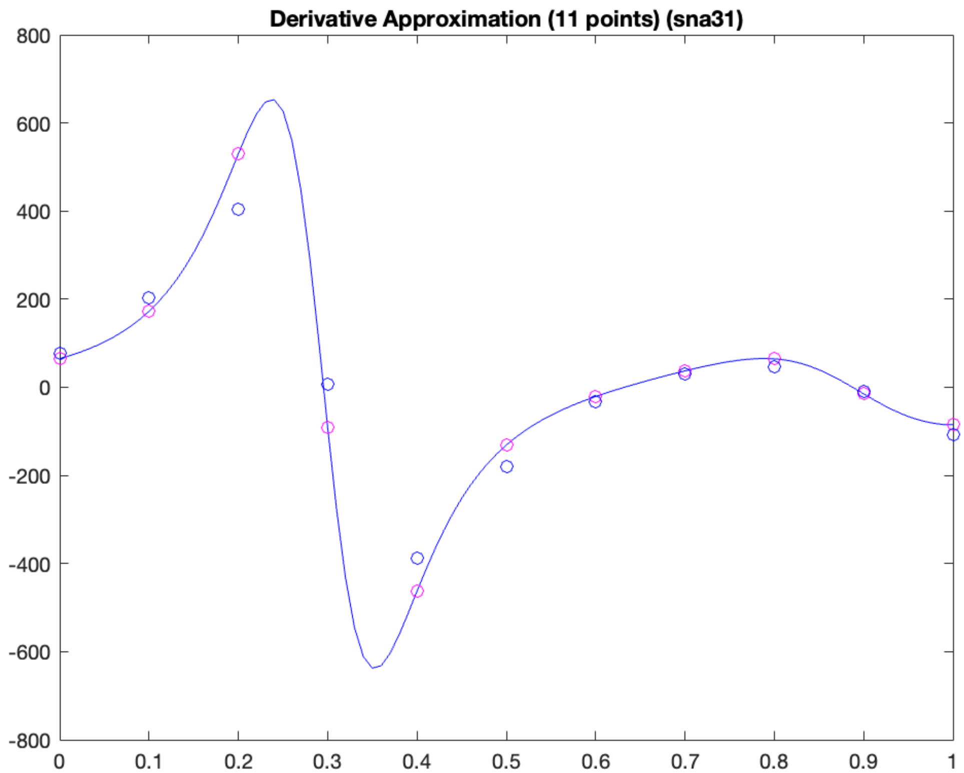
9.2 Exercise 2

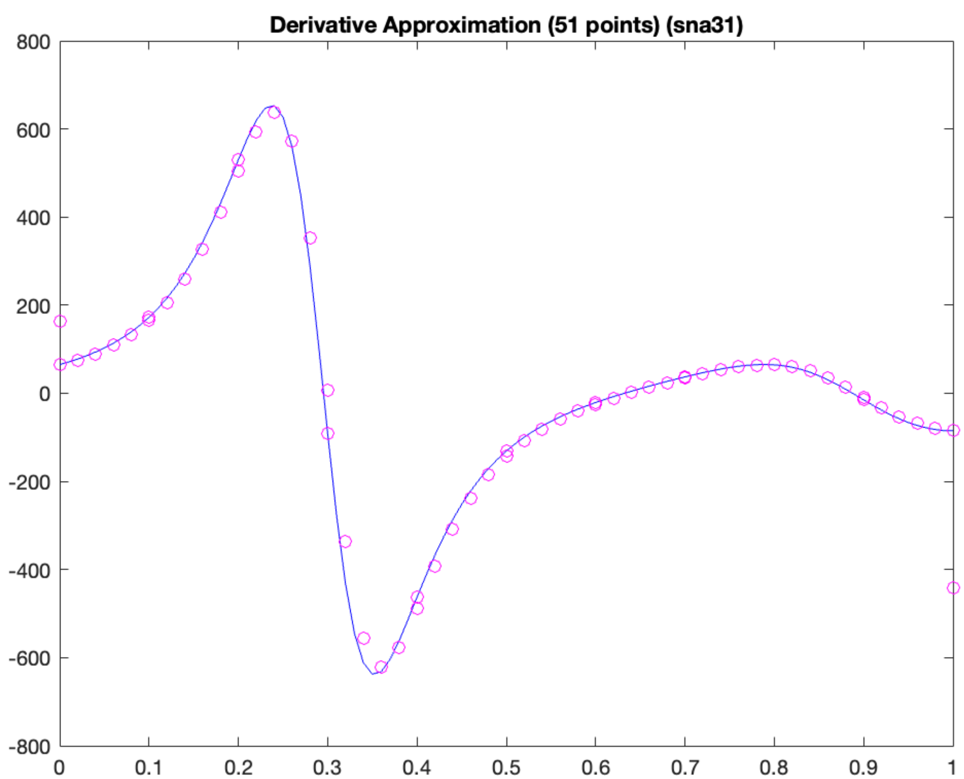


9.3 Exercise 3

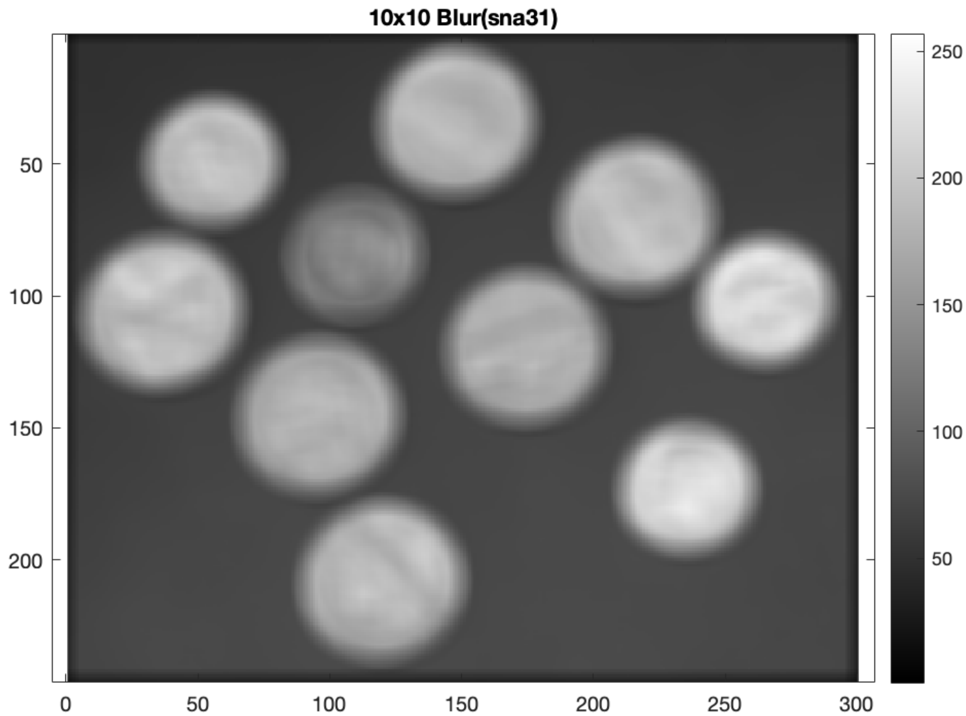


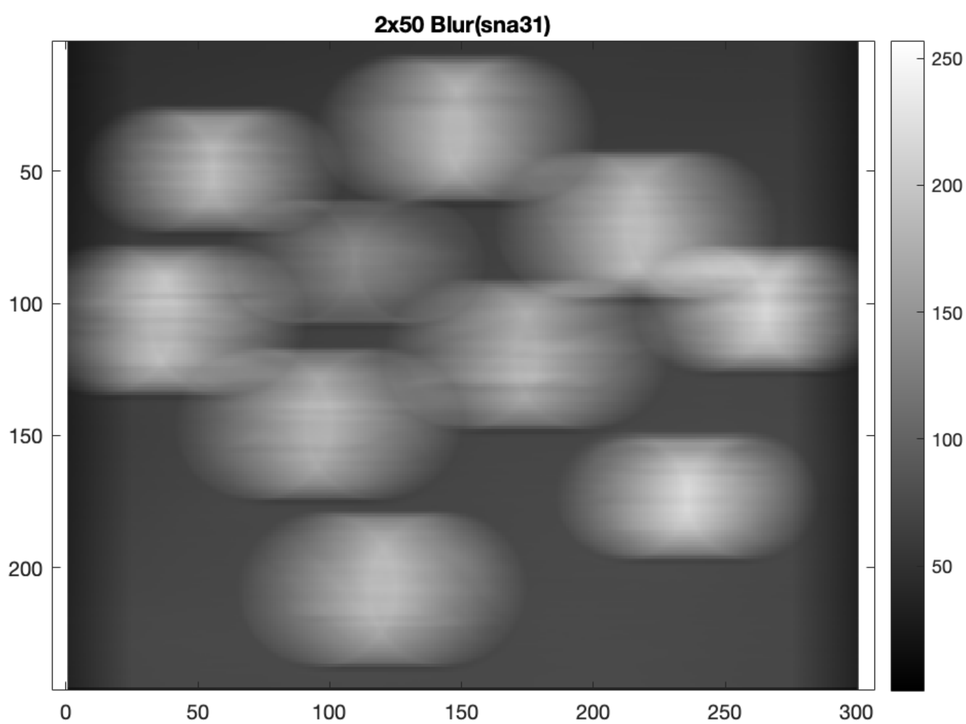
9.4 Exercise 4

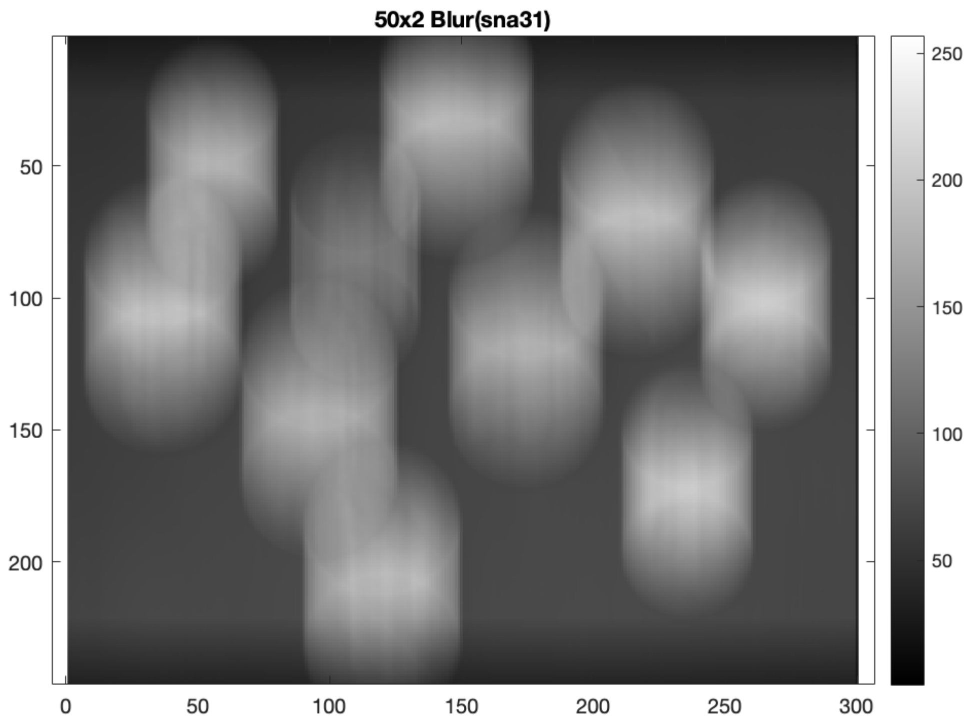




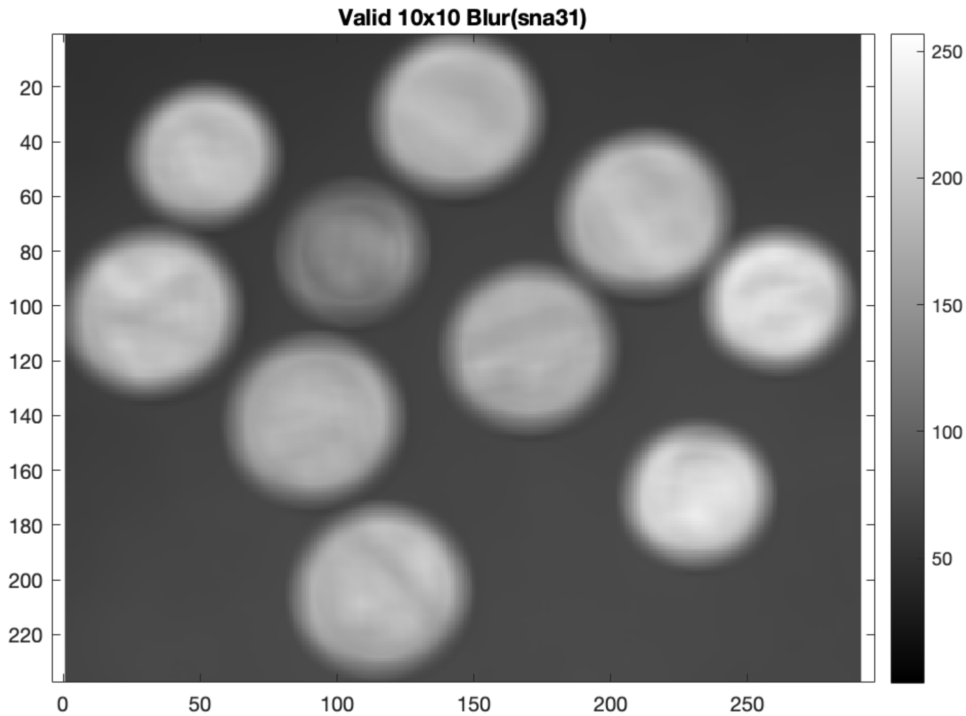
9.5 Exercise 5

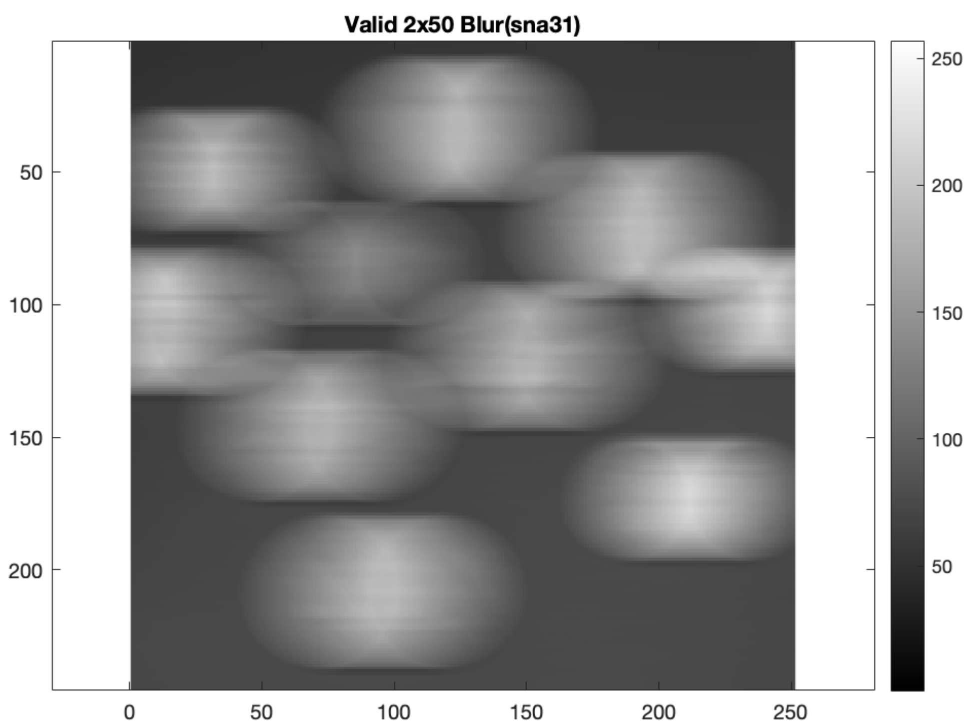


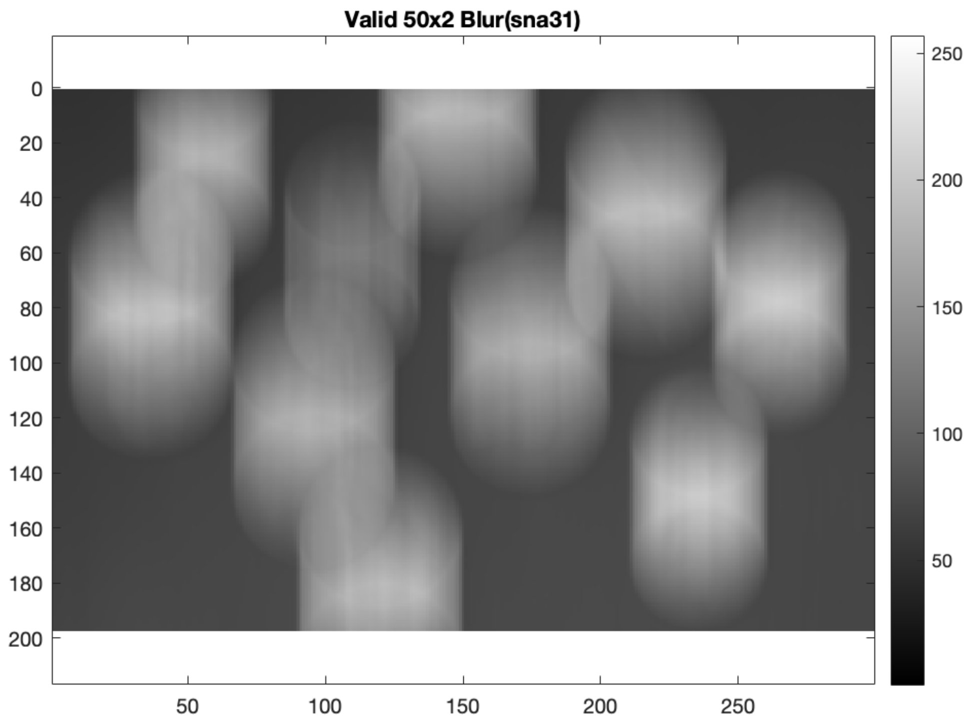



50x2 Blur(sna31)
50
100
150
200
0
50
100
150
200
250
300
250
200
150
100
50

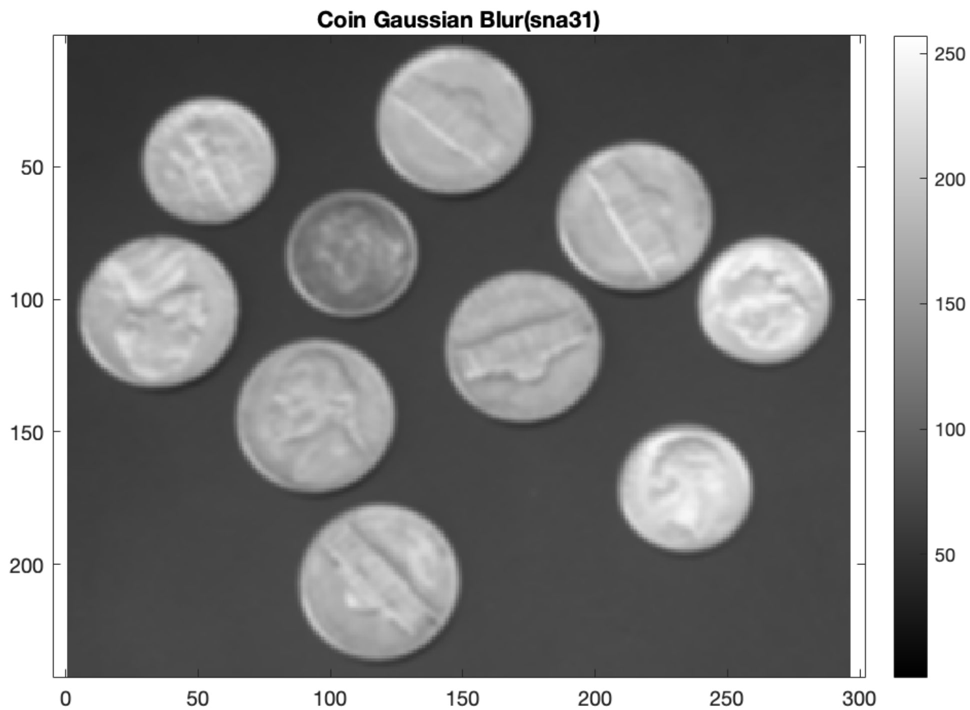
9.6 Exercise 6

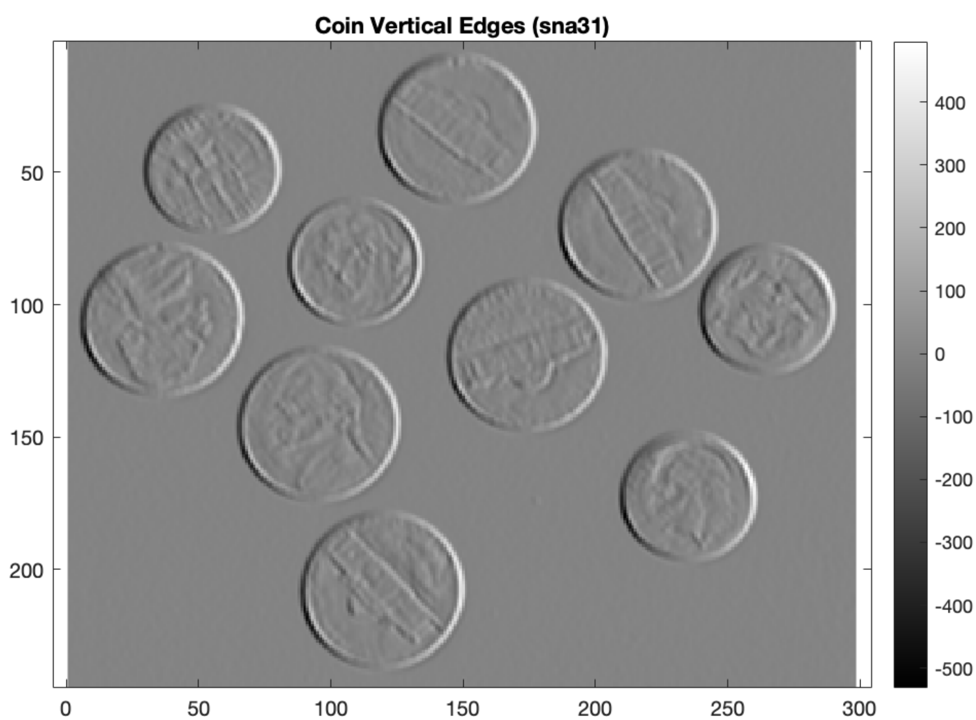


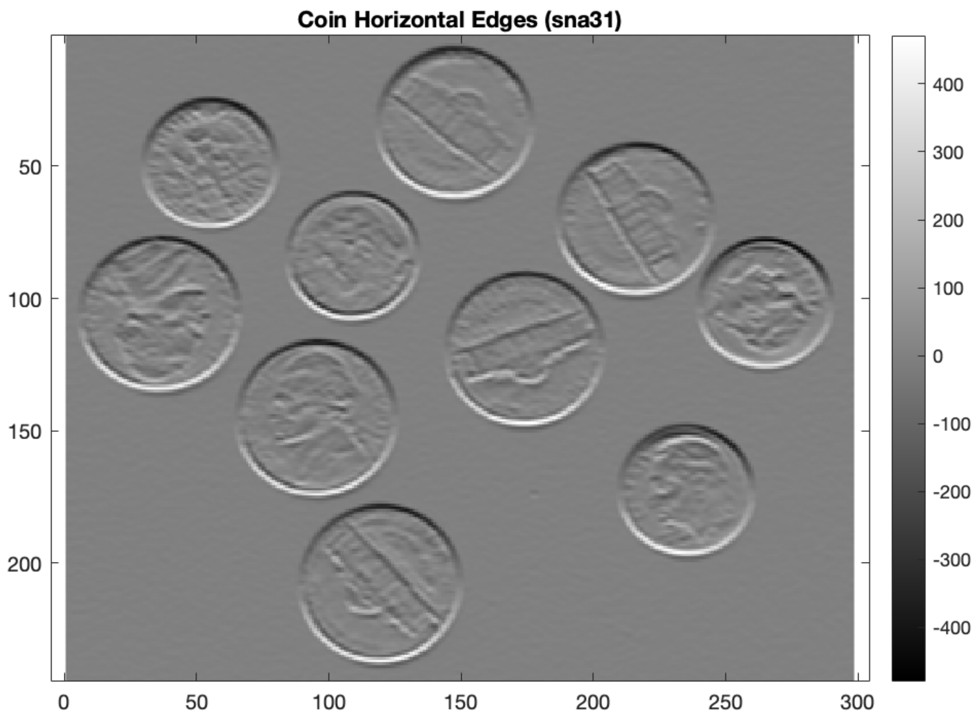


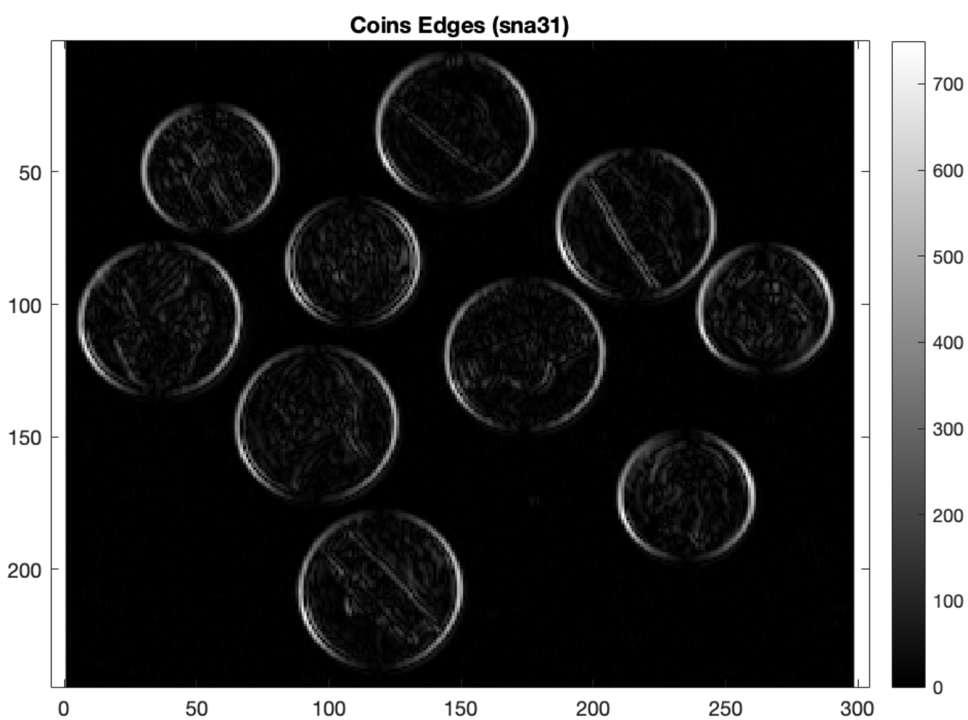

Valid 50x2 Blur(sna31)
0
20
40
60
80
100
120
140
160
180
200
50
100
150
200
250
250
200
150
100
50

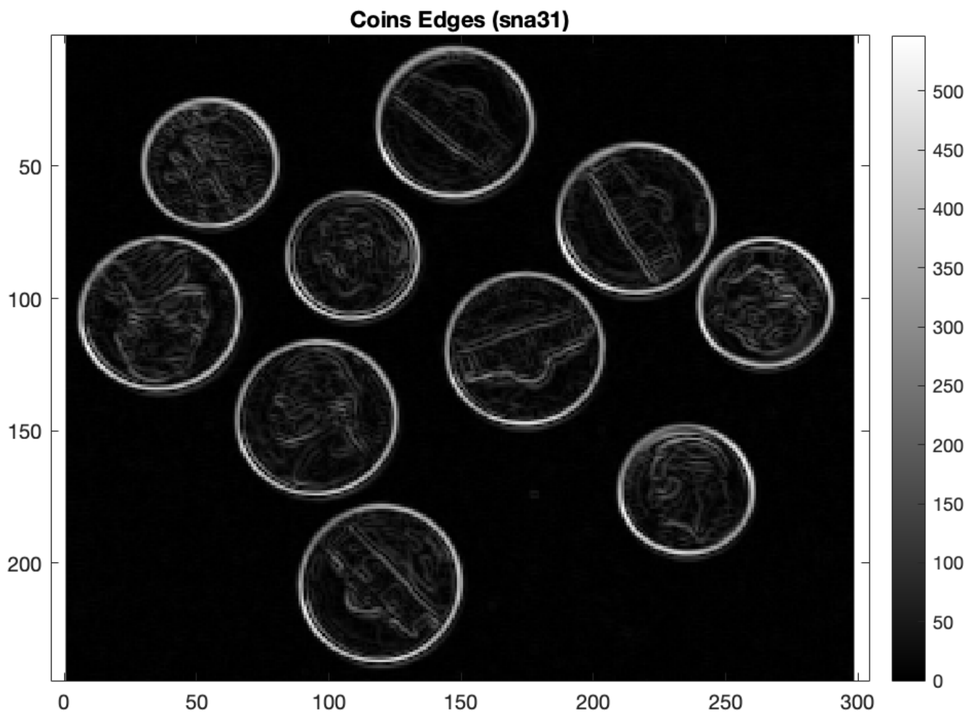
9.7 Exercise 7

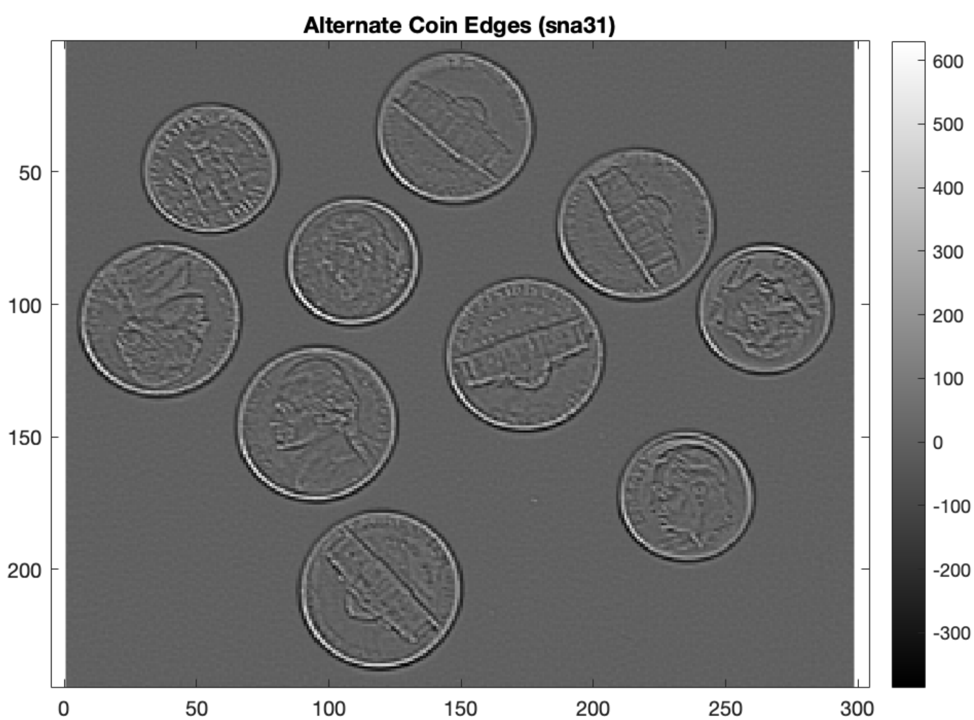




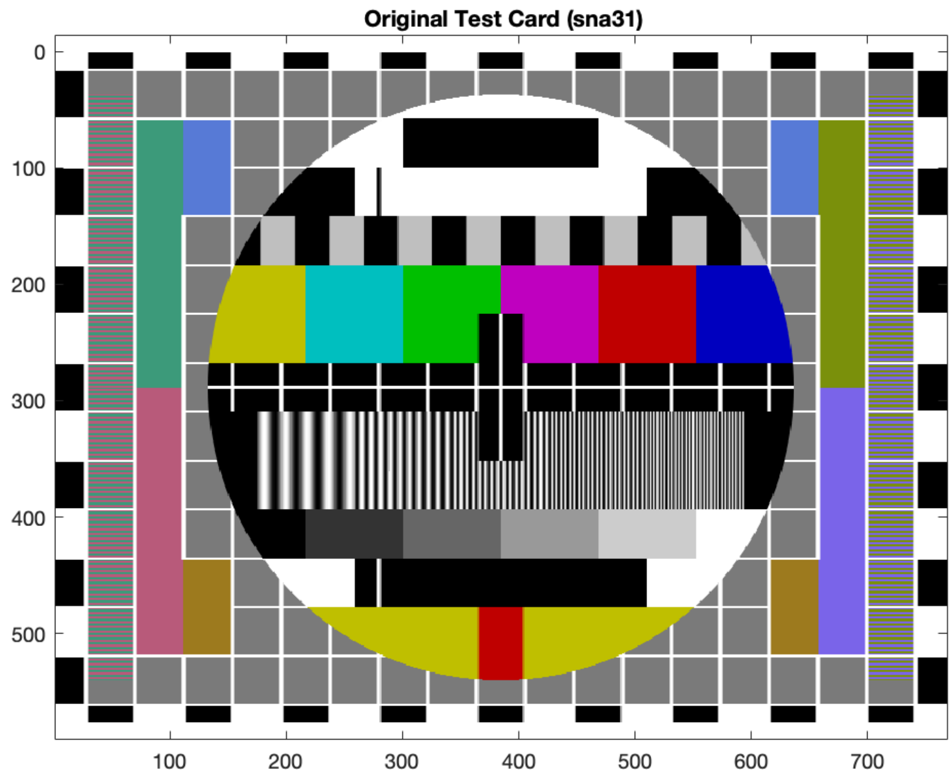

Coin Horizontal Edges (sna31)
50
100
150
200
0
50
100
150
200
250
300
400
300
200
100
0
-100
-200
-300
-400

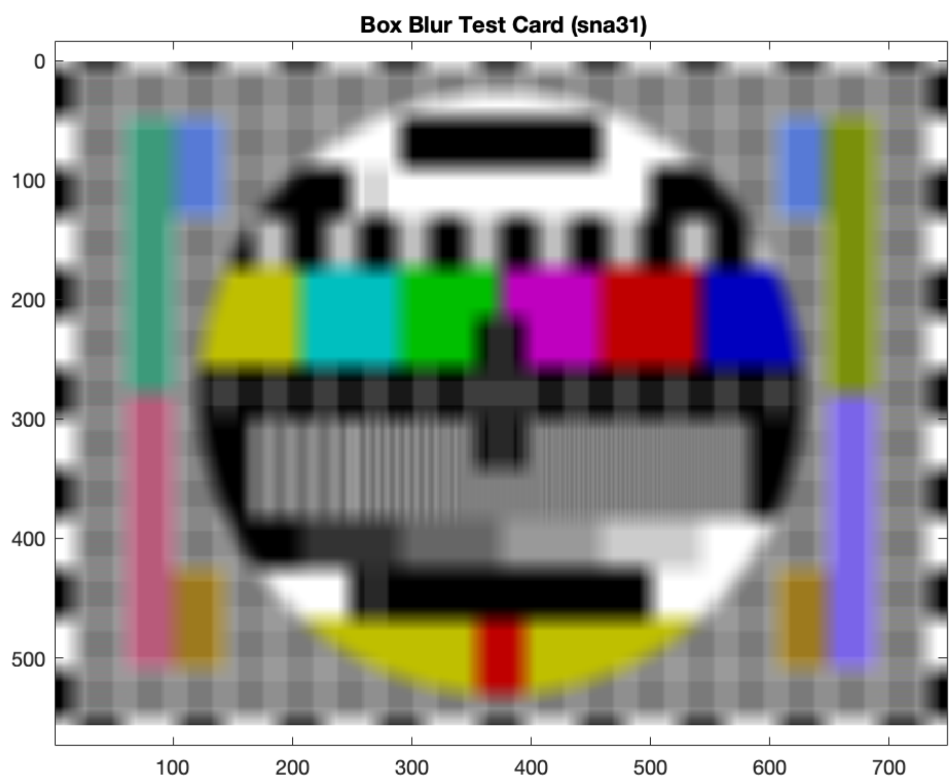

Coins Edges (sna31)
50
100
150
200
0
50
100
150
200
250
300
700
600
500
400
300
200
100
0

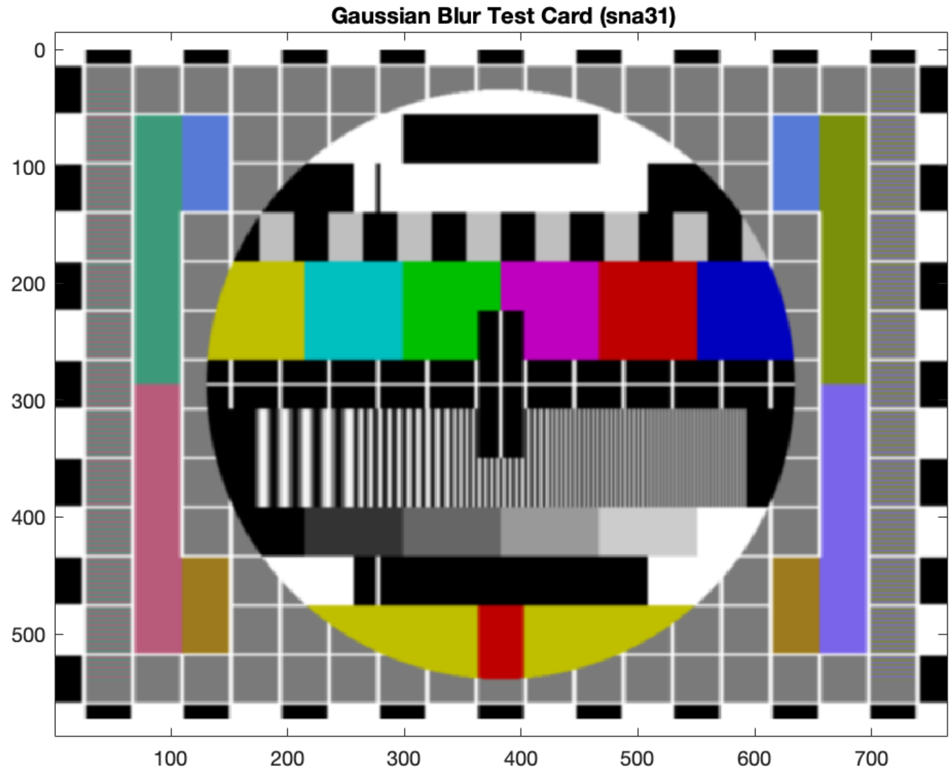

Coins Edges (sna31)
50
100
150
200
0
50
100
150
200
250
300
500
450
400
350
300
250
200
150
100
50
0

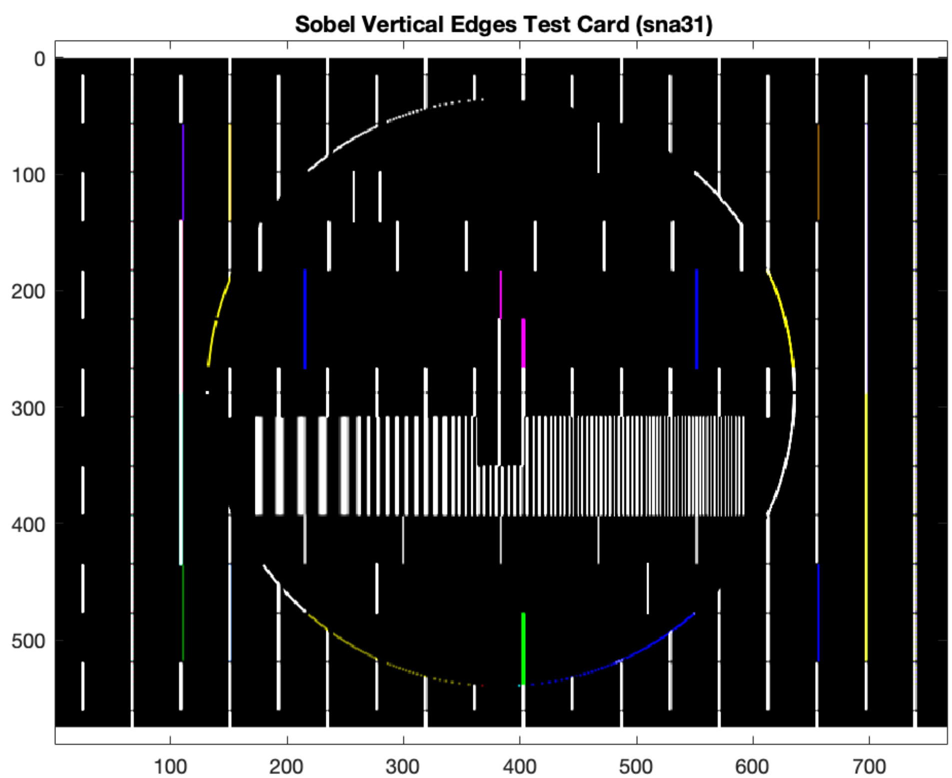

Alternate Coin Edges (sna31)
50
100
150
200
0
50
100
150
200
250
300
600
500
400
300
200
100
0
-100
-200
-300

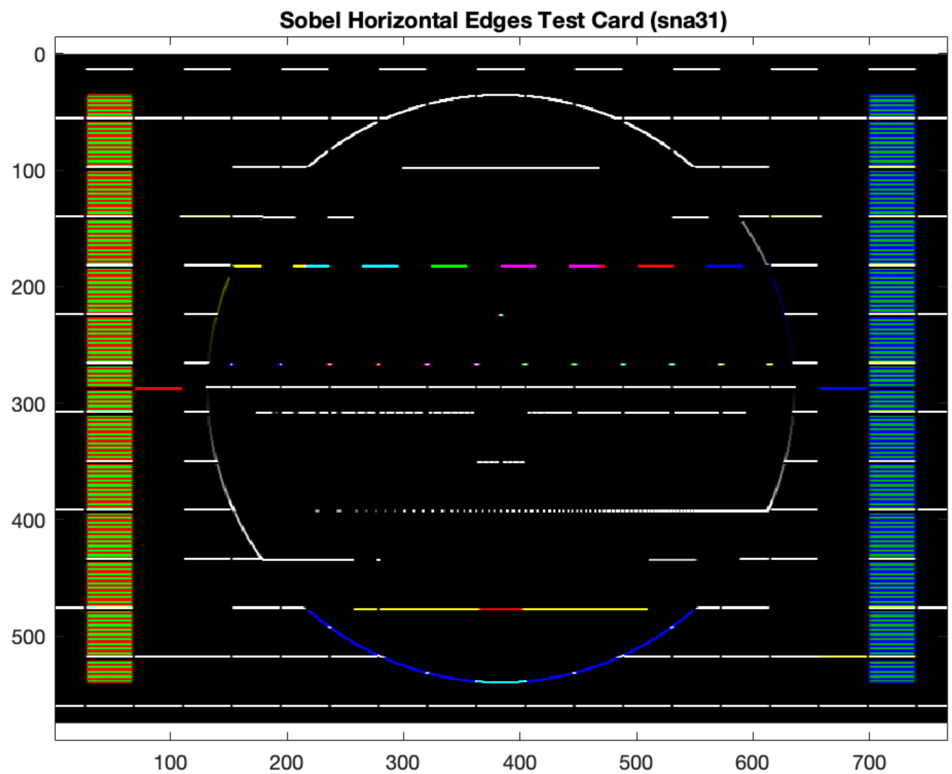
9.8 Exercise 8



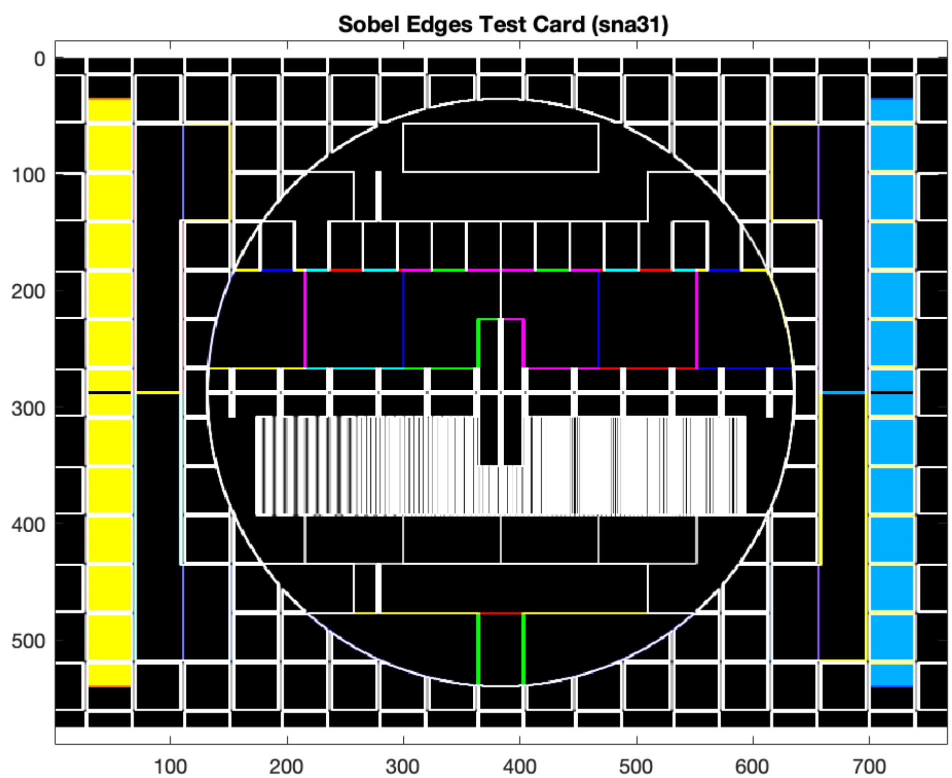



Gaussian Blur Test Card (sna31)
0
100
200
300
400
500
100
200
300
400
500
600
700


Sobel Vertical Edges Test Card (sna31)
0
100
200
300
400
500
100
200
300
400
500
600
700



Sobel Horizontal Edges Test Card (sna31)
0
100
200
300
400
500
100
200
300
400
500
600
700



Sobel Edges Test Card (sna31)
0
100
200
300
400
500
100
200
300
400
500
600
700

10 Codes

10.1 Exercise 1

Code 1

```
rad = 100;
del = 10;
[x, y] = meshgrid((-3*rad-del):(3*rad+del));
[rows, cols] = size(x);
dist = @(x, y, xc, yc) sqrt((x-xc).^2+(y-yc).^2);

p=0.005;


venn_img = zeros(rows, cols, 3);
venn_img(:,:,1) = 1 ./(1+(p.*(dist(x, y, rad.*cos(0), rad.*sin(0)))));
venn_img(:,:,2) = 1 ./(1+(p.*(dist(x, y, rad.*cos(2*pi/3), rad.*sin(2*pi/3)))));
venn_img(:,:,3) = 1 ./(1+(p.*(dist(x, y, rad.*cos(4*pi/3), rad.*sin(4*pi/3)))));

figure(1); clf
image(venn_img)
axis equal
```

10.2 Exercise 2

Code 2

```
x = rand(200, 200, 3);


figure(1); clf
image(x)
axis equal
```

10.3 Exercise 3

Code 3

```
tc = linspace(0, 1, 101);
xc = humps(tc);
td = linspace(0, 1, 11);
xd = humps(td);

h = [0.5  0.5];
y2 = conv(xd, h, 'same');

h5= [0.2  0.2  0.2  0.2  0.2];
y5 = conv(xd, h5, 'same');

figure (1); clf
plot(tc, xc, 'b-')
hold on

plot(td, xd, 'bo')
hold on

plot(td, y2, 'r-o')
hold on

plot(td, y5, 'g-o')
hold off
```

10.4 Exercise 4

Code 4

```
tc = linspace(0, 1, 101);
xc = humps(tc);
deltatc = tc(2)-tc(1);
td = linspace(0, 1, 51);
xd = humps(td);
deltatd = td(2)-td(1);

h = [1,0,-1]/2/deltatd;
approxhumps= conv(xd, h, 'same');

figure (1); clf
plot(tc, xc, 'b-')
hold on
plot(td, xd, 'bo')
hold off
title('Values')

figure (2); clf
twopointdiff = diff(xc)/deltatc;
twopointdiff(end+1)=twopointdiff(end);
plot(tc, twopointdiff , 'b-')
hold on
plot(tc(1:10:end), twopointdiff(1:10:end), 'mo')
hold on
plot(td, approxhumps, 'mo')
hold off
title('Derivative Approximation (51 points) (sna31)')
```

10.5 Exercise 5

Code 5

```
x = imread('coins.png');
h = ones(50, 2)/10^2;
y = conv2(x, h, 'same');
figure (1); clf
image(x)
axis equal; colormap gray; colorbar
title('Original')

figure (2); clf
image(y)
axis equal; colormap gray; colorbar
title('50x2 Blur(sna31)')
```

10.6 Exercise 6

Code 6

```
x = imread('coins.png');

h = ones(10, 10)/10^2;
y = conv2(x, h, 'valid');
figure (1); clf
image(x)
axis equal; colormap gray; colorbar
title('Original')
figure (2); clf
image(y)
axis equal; colormap gray; colorbar
title('Valid 10x10 Blur(sna31)')
```

10.7 Exercise 7

Code 7

```
x = imread('coins.png');

h = 1/256.*[1, 4, 6, 4, 1; 4, 16, 24, 16, 4; 6, 24, 36, 24, 6; 4, 16, 24, 16, 4; 1, 4, 6, 4,1
h_x = [-1 0 1; -1 0 1; -1 0 1];
h_y =[-1 -1 -1; 0 0 0; 1 1 1];
h_new =[-1 -1 -1; -1 8 -1; -1 -1 -1];

CoinsEdgeY = conv2(x,h_y, 'valid');
CoinsEdgeX = conv2(x,h_x, 'valid');
Coin_New = conv2(x, h_new, 'valid');

t = sqrt((CoinsEdgeY).^2 + (CoinsEdgeX).^2);
figure (1); clf
image(x)
axis equal; colormap gray; colorbar
title('Original')
figure (2); clf
imagesc(Coin_New)
axis equal; colormap gray; colorbar
title('Alternate Coin Edges (sna31)')
```

10.8 Exercise 8

Code 8

```
x = imread('TestCard.png');

h_box_blur = ones(21, 21)/441;

h_gauss = 1/256.*[1, 4, 6, 4, 1; 4, 16, 24, 16, 4; 6, 24, 36, 24, 6; 4, 16, 24, 16, 4; 1, 4,
h_sobel_x = [1 0 -1; 2 0 -2; 1 0 -1];
h_sobel_y =[1 2 1; 0 0 0; -1 -2 -1];
figure (1); clf
imagesc(x)
axis equal;
title(' Original Test Card (sna31)')

y1(:,:,1)= conv2(x(:,:,1), h_sobel_y, 'valid');
y1(:,:,2)= conv2(x(:,:,2), h_sobel_y, 'valid');
y1(:,:,3)= conv2(x(:,:,3), h_sobel_y, 'valid');

y2(:,:,1)= conv2(x(:,:,1), h_sobel_x, 'valid');
y2(:,:,2)= conv2(x(:,:,2), h_sobel_x, 'valid');
y2(:,:,3)= conv2(x(:,:,3), h_sobel_x, 'valid');
t = sqrt((y1).^2 + (y2).^2);

figure (2); clf
imagesc(uint8(t))
axis equal;
title(' Sobel Edges Test Card (sna31)')
```