The exam and the image for the exam can be downloaded by pasting into your web browser address (URL) space:

ftp://ftp/hellerm/EEE221/MidtermExam

or, by going to EEE221 on my web site, click on MIDTERM EXAM. Click on Word Document to open up word with the Midterm Exam

Click on Midterm Exam Image,
1. 20 pt's. An engineer thinks that edges can be found by using the following wopr:

\[
\begin{array}{ccc}
1 & 0 & -1 \\
0 & 0 & 0 \\
-1 & 0 & 1 \\
\end{array}
\]

Image:

\[
\begin{array}{cccccc}
50 & 30 & 10 & 5 & 5 \\
30 & 60 & 35 & 10 & 5 \\
10 & 35 & 60 & 30 & 10 \\
5 & 10 & 30 & 50 & 35 \\
5 & 5 & 10 & 35 & 55 \\
\end{array}
\]

a) Apply this wopr to the given image:

\[
\begin{array}{cccccc}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 70 & 20 & -45 & 0 & 0 \\
0 & 20 & 90 & 35 & 0 & 0 \\
0 & -45 & 35 & 95 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
\end{array}
\]

Threshold = 50

\[
\begin{array}{cccccc}
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
\end{array}
\]

c) What indicates to you that the wopr might be an edge detector? Explain.

It appears from applying the given wopr to the image that it will at detect diagonal edges and corners; however, upon closer inspection, the wopr is not good for finding horizontal and vertical edges. That is, consider pixels in an image with intensity \( h \) as follows:

Vertical edges:

\[
\begin{array}{ccc}
0 & h & h \\
0 & h & h \\
0 & h & h \\
\end{array}
\]

or

\[
\begin{array}{ccc}
0 & 0 & h \\
0 & 0 & h \\
0 & 0 & h \\
\end{array}
\]

wopr results in \( p^* = 0 \)
### Horizontal edges:

<table>
<thead>
<tr>
<th>h</th>
<th>h</th>
<th>h</th>
<th>h</th>
<th>h</th>
<th>h</th>
</tr>
</thead>
<tbody>
<tr>
<td>h</td>
<td>h</td>
<td>h</td>
<td>h</td>
<td>h</td>
<td>h</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

or

<table>
<thead>
<tr>
<th>h</th>
<th>h</th>
<th>h</th>
<th>h</th>
<th>h</th>
<th>h</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

wopr results in $p^* = 0$

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

wopr results in $p^* = 0$

However, this wopr will also find corners:

<table>
<thead>
<tr>
<th>h</th>
<th>h</th>
<th>0</th>
<th>h</th>
<th>h</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>h</td>
<td>h</td>
<td>0</td>
<td>h</td>
<td>h</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

or

<table>
<thead>
<tr>
<th>h</th>
<th>h</th>
<th>0</th>
<th>h</th>
<th>h</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>h</td>
<td>h</td>
<td>0</td>
<td>h</td>
<td>h</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

wopr results in $p^* = +h$

<table>
<thead>
<tr>
<th>0</th>
<th>h</th>
<th>h</th>
<th>0</th>
<th>h</th>
<th>h</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>h</td>
<td>h</td>
<td>0</td>
<td>h</td>
<td>h</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

or

<table>
<thead>
<tr>
<th>0</th>
<th>h</th>
<th>h</th>
<th>0</th>
<th>h</th>
<th>h</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>h</td>
<td>h</td>
<td>0</td>
<td>h</td>
<td>h</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

wopr results in $p^* = -h$

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

or

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

wopr results in $p^* = +h$

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

or

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

wopr results in $p^* = -h$

A 45 degree diagonal with + slope yields $p^* = +2h$ and 45 degree diagonal with – slope yields $p^* = -2h$. Other diagonals would yield + or – h.
Following is an example of applying the wopr to an object:

```matlab
% MidTermS03P1.m
clear all
format compact
getbini
wopr=[1 0 -1
      0 0 0
    -1 0 1];
iedge=winopr(iclean,wopr);
iedge=abs(iedge);
figure(1)
clf
histplot(iedge);
ibinedge=threshld(iedge,0.5);
figure(2)
clf
subplot(1,2,1)
shows(iclean)
subplot(1,2,2)
shows(ibinedge);
```
2. 20 pt's. Equalize the following image to improve the contrast of the object(s) of interest. Show all of the steps used to obtain the equalized image.

Image:

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>50</td>
<td>30</td>
<td>10</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>30</td>
<td>60</td>
<td>35</td>
<td>10</td>
<td>5</td>
</tr>
<tr>
<td>10</td>
<td>35</td>
<td>60</td>
<td>30</td>
<td>10</td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>30</td>
<td>50</td>
<td>35</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>10</td>
<td>35</td>
<td>55</td>
</tr>
</tbody>
</table>

Contrast enhanced image:

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>53</td>
<td>38</td>
<td>29</td>
<td>14</td>
<td>14</td>
</tr>
<tr>
<td>38</td>
<td>60</td>
<td>48</td>
<td>29</td>
<td>14</td>
</tr>
<tr>
<td>29</td>
<td>48</td>
<td>60</td>
<td>38</td>
<td>29</td>
</tr>
<tr>
<td>14</td>
<td>29</td>
<td>38</td>
<td>53</td>
<td>48</td>
</tr>
<tr>
<td>14</td>
<td>14</td>
<td>29</td>
<td>48</td>
<td>55</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Histogram</th>
<th>p</th>
<th>h(p)</th>
<th>q</th>
<th>q(i)=q(i-1)+h(p(i)*c</th>
<th>q integer</th>
</tr>
</thead>
<tbody>
<tr>
<td>p0</td>
<td>0</td>
<td>0</td>
<td>q0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>p1</td>
<td>5</td>
<td>6</td>
<td>q1</td>
<td>14.4</td>
<td>14</td>
</tr>
<tr>
<td>p2</td>
<td>10</td>
<td>6</td>
<td>q2</td>
<td>28.8</td>
<td>29</td>
</tr>
<tr>
<td>p3</td>
<td>15</td>
<td>0</td>
<td>q3</td>
<td>28.8</td>
<td>29</td>
</tr>
<tr>
<td>p4</td>
<td>20</td>
<td>0</td>
<td>q4</td>
<td>28.8</td>
<td>29</td>
</tr>
<tr>
<td>p5</td>
<td>25</td>
<td>0</td>
<td>q5</td>
<td>28.8</td>
<td>29</td>
</tr>
<tr>
<td>p6</td>
<td>30</td>
<td>4</td>
<td>q6</td>
<td>38.4</td>
<td>38</td>
</tr>
<tr>
<td>p7</td>
<td>35</td>
<td>4</td>
<td>q7</td>
<td>48</td>
<td>48</td>
</tr>
<tr>
<td>p8</td>
<td>40</td>
<td>0</td>
<td>q8</td>
<td>48</td>
<td>48</td>
</tr>
<tr>
<td>p9</td>
<td>45</td>
<td>0</td>
<td>q9</td>
<td>48</td>
<td>48</td>
</tr>
<tr>
<td>p10</td>
<td>50</td>
<td>2</td>
<td>q10</td>
<td>52.8</td>
<td>53</td>
</tr>
<tr>
<td>p11</td>
<td>55</td>
<td>1</td>
<td>q11</td>
<td>55.2</td>
<td>55</td>
</tr>
<tr>
<td>p12</td>
<td>60</td>
<td>2</td>
<td>q12</td>
<td>60</td>
<td>60</td>
</tr>
</tbody>
</table>

\[5 \times 5 = 25\]
\[M = 61\text{ Intensities}\]
\[n = 25\]
\[c = (M-1)/n = 2.400000\]
3. 20 pt's. Fit a straight line through the given points using the Hough Transform. Use 2 unit x 2 unit accumulator cells.

\[ y = ax + b \]

Parameter Space

1. \((6,2)\)
   \[ b = -6a + 2 \]
2. \((3,3)\)
   \[ b = -3a + 3 \]
3. \((2,5)\)
   \[ b = -2a + 5 \]

Selected \((2, 10)\)

At least 4 possibilities:

- \(a \& b\)
- \(-6\)
- \(-2.8\)
- \(-210\) selected

\[ y = \text{slope} \cdot x + \text{intercept} = -2x + 10 \]
4. 20 pt's. Construct the $H_{LP}(u,v)$ matrix representing the following ideal cube filter for low pass filtering a 2D Fourier Transformed 9x9 image. The transformed image has been shifted so that $(u,v) = (0,0)$ is in the center of the matrix.

$$H_{LP}(u,v):
\begin{bmatrix}
-4 & -3 & -2 & -1 & 0 & 1 & 2 & 3 & 4 \\
-4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-3 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
-2 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
-1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
 2 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
 3 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \\
 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}$$
5. 20 pts. The following problems are based on the image ix.jpg. Put ix.jpg into a directory that MATLAB has a path to. Use the following MATLAB command to load ix into MATLAB:

```
>>ix=double(imread('ix.jpg','jpg'));
```

You can also past this command at the beginning of a script (.m) file.

a) Use a linear stretch to improve the contrast and to expose the moon. Use only the show function to display your results. Sketch the linear stretch below:

![Linear stretch diagram](image)

b) Find a binary image such that the moon (what can be seen of it) can be chain coded. List the steps and specific values (woprs, thresholds, stretch parameters, etc.) that you used below and chain code the moon using an 8 way code.

Perimeter of moon = **252**

Steps used to obtain an image such that the moon can be chain coded:

1. Histogram ix (given image) to select range of stretch to be used.

2. ix min and max intensities are 0 and 24

3. Stretch ix \( p \) (0 to 24) maps to \( q \) (0 to 225) – see above linear stretch.

4. Try different global thresholds – 57 was selected – ixbin = thresholded binary image.

5. Started chain code 8 way at coordinates \( x = 23 \) and \( y = 42 \) of ixbin where 0,0 is the upper left corner and \( y \) is the horizontal axis.

6. Perimeter of moon = length of chain code = 252.
% eee221midtermP5.m

clear all

format

format compact

ix=double(imread('ix.jpg','jpg'));

figure(1)

clf

histplot(ix);

figure(2)

clf

show(ix)

ixs=stretch(ix);  % Stretchs between 0 and 24

figure(3)

clf

show(ixs)

figure(4)

clf

histplot(ixs);

ixbin=threshld(ixs,57);  % Threshold at 57

figure(5)

clf

shows(ixbin)

%compacti(ixbin);

chaincode=ccode8(ixbin,23,42);

perimeter=length(chaincode)