Lab #7

Name: Grant Beatty SID: 862037946

EE146 Section (022)

Problem 1

Template (regular) Cross correlation Normalized

(Template intensity)

(Image intensity)

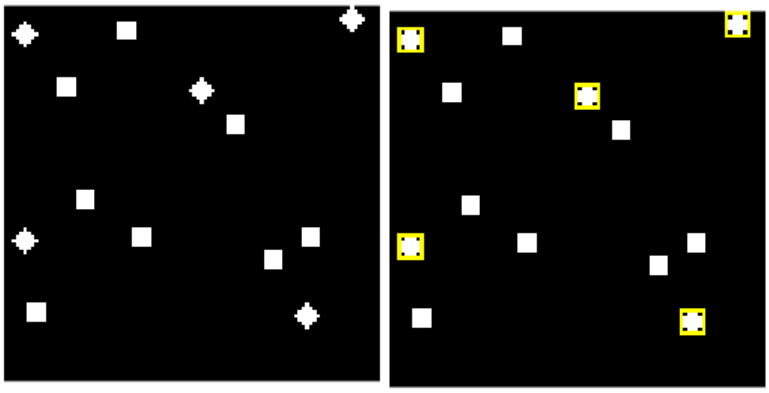
(Image and Template Intensity)

(Rotation)

The correlation template matching fails when there is a sharpness change or orientation change to the image or template. The normalized correlation template matching works for all sharpness changes to the image or template. It fails when the orientation is changed.

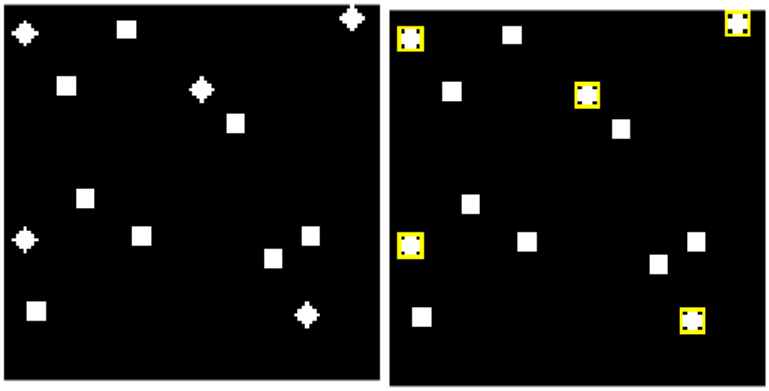
Problem 2

Normalized Cross correlation



```
normalized cross-correlation time: 2.262330e-02
```

Chamfer



```
ChamferMatch time: 6.374120e-02
```

The CPU time for the normalized cross-correlation is a little less than half of the CPU time for the ChamferMatch. The normalized cross-correlation is faster but under certain circumstances it can produce false positives when scanning binary images. Although slower, the ChamferMatch is more accurate for binary images in general.

Lab #7 Code

```
%lab 7 matching / distance transform

clc
clear all
close all
```

Problem 1 - templete matching

a)

```
I = imread('cameraman.tif');
t = I(60:85,130:170 ); % this is your template image
imshow(t,"InitialMagnification",500)
```

```
image=double(I);
temp=double(t);

% write your code here
% functions: imadjust(), imrotate(), xcorr2(), normxcorr2()

% step 1 - get cross correlation / normalized cross correlation

crr=xcorr2(image,temp);                    %cross cross correlation
ncrr=normxcorr2(t,I);                      %normalized cross correlation

crr=crr(26:height(crr),41:width(crr));     %shifting correlation
ncrr=ncrr(26:height(ncrr),41:width(ncrr)); %shifting normalized correlation

% step 2 - find the peak value location in the correlation matrix
minval=abs(crr-sum(temp.^2,"all"));

[xpeak,ypeak]=find(minval==min(minval(:))); %correlation peak
[nxpeak,nypeak]=find(ncrr==max(ncrr(:)));    %normalized correlation peak
```

```
% step 3 -  use function size() to get the size of template image

[R C]=size(t);                                  %size

shape=[(ypeak) (xpeak) C R];                    %correlation shape and position
nshape=[(nypeak) (nxpeak) C R];                 %normalized correlation shape and
position


% step 4 - use insertShape() to display the matched box

I1=insertShape(I,'Rectangle',shape,"LineWidth",2);
imshow(I1)
```

```
I2=insertShape(I,'Rectangle',nshape,"LineWidth",2);
imshow(I2)
```

b)

```
I = imread('cameraman.tif');
t = I(60:85,130:170 ); % this is your template image
t=imadjust(t);
imshow(t,"InitialMagnification",500)
```

```
image=double(I);
temp=double(t);

% step 1 - get cross correlation / normalized cross correlation

crr=xcorr2(image,temp);                      %cross cross correlation
ncrr=normxcorr2(t,I);                        %normalized cross correlation

crr=crr(26:height(crr),41:width(crr));       %shifting correlation
ncrr=ncrr(26:height(ncrr),41:width(ncrr));   %shifting normalized correlation

% step 2 - find the peak value location in the correlation matrix
minval=abs(crr-sum(temp.^2,"all"));
```

```
 [xpeak,ypeak]=find(minval==min(minval(:))); %correlation peak
 xpeak=sort(xpeak);
 ypeak=sort(ypeak);
 [nxpeak,nypeak]=find(ncrr==max(ncrr(:)));    %normalized correlation peak

 % step 3 -  use function size() to get the size of template image

 [R C]=size(t);                               %size

 shape=[(ypeak) (xpeak) C R];                 %correlation shape and position
 nshape=[(nypeak) (nxpeak) C R];              %normalized correlation shape and
position


 % step 4 - use insertShape() to display the matched box

 I3=insertShape(I,'Rectangle',shape,"LineWidth",2);
 imshow(I3)
```

```
 I4=insertShape(I,'Rectangle',nshape,"LineWidth",2);
 imshow(I4)
```

c)

```
I = imread('cameraman.tif');
t = I(60:85,130:170 ); % this is your template image
I=imadjust(I);
imshow(t,"InitialMagnification",500)
```

```
image=double(I);
temp=double(t);

% step 1 - get cross correlation / normalized cross correlation

crr=xcorr2(image,temp);                     %cross cross correlation
ncrr=normxcorr2(t,I);                       %normalized cross correlation

crr=crr(26:height(crr),41:width(crr));      %shifting correlation
ncrr=ncrr(26:height(ncrr),41:width(ncrr));  %shifting normalized correlation

% step 2 - find the peak value location in the correlation matrix
minval=abs(crr-sum(temp.^2,"all"));
```

```
 [xpeak,ypeak]=find(minval==min(minval(:))); %correlation peak
 xpeak=sort(xpeak);
 ypeak=sort(ypeak);
 [nxpeak,nypeak]=find(ncrr==max(ncrr(:)));    %normalized correlation peak

 % step 3 -  use function size() to get the size of template image

 [R C]=size(t);                               %size

 shape=[(ypeak) (xpeak) C R];                 %correlation shape and position
 nshape=[(nypeak) (nxpeak) C R];              %normalized correlation shape and
position


 % step 4 - use insertShape() to display the matched box

 I3=insertShape(I,'Rectangle',shape,"LineWidth",2);
 imshow(I3)
```

```
 I4=insertShape(I,'Rectangle',nshape,"LineWidth",2);
 imshow(I4)
```

d)

```
I = imread('cameraman.tif');
t = I(60:85,130:170 ); % this is your template image
I=imadjust(I);
t=imadjust(t);
imshow(t,"InitialMagnification",500)
```

```
image=double(I);
temp=double(t);

% step 1 - get cross correlation / normalized cross correlation

crr=xcorr2(image,temp);                    %cross cross correlation
ncrr=normxcorr2(t,I);                      %normalized cross correlation

crr=crr(26:height(crr),41:width(crr));     %shifting correlation
ncrr=ncrr(26:height(ncrr),41:width(ncrr)); %shifting normalized correlation

% step 2 - find the peak value location in the correlation matrix
```

```
 minval=abs(crr-sum(temp.^2,"all"));

 [xpeak,ypeak]=find(minval==min(minval(:))); %correlation peak
 xpeak=sort(xpeak);
 ypeak=sort(ypeak);
 [nxpeak,nypeak]=find(ncrr==max(ncrr(:)));    %normalized correlation peak

 % step 3 -  use function size() to get the size of template image

 [R C]=size(t);                                %size

 shape=[(ypeak) (xpeak) C R];                  %correlation shape and position
 nshape=[(nypeak) (nxpeak) C R];               %normalized correlation shape and
position


 % step 4 - use insertShape() to display the matched box

 I3=insertShape(I,'Rectangle',shape,"LineWidth",2);
 imshow(I3)
```

```
 I4=insertShape(I,'Rectangle',nshape,"LineWidth",2);
 imshow(I4)
```

e)

```
I = imread('cameraman.tif');
t = I(60:85,130:170 ); % this is your template image
t=imrotate(t,90);

imshow(t,"InitialMagnification",500)
```

```
image=double(I);
temp=double(t);

% step 1 - get cross correlation / normalized cross correlation

crr=xcorr2(image,temp);                    %cross cross correlation
ncrr=normxcorr2(t,I);                      %normalized cross correlation

crr=crr(26:height(crr),41:width(crr));     %shifting correlation
```

```
 ncrr=ncrr(26:height(ncrr),41:width(ncrr));  %shifting normalized correlation

 % step 2 - find the peak value location in the correlation matrix
 minval=abs(crr-sum(temp.^2,"all"));

 [xpeak,ypeak]=find(minval==min(minval(:))); %correlation peak
 xpeak=sort(xpeak);
 ypeak=sort(ypeak);
 [nxpeak,nypeak]=find(ncrr==max(ncrr(:)));    %normalized correlation peak

 % step 3 -  use function size() to get the size of template image

 [R C]=size(t);                                %size

 shape=[(ypeak) (xpeak) C R];                  %correlation shape and position
 nshape=[(nypeak) (nxpeak) C R];               %normalized correlation shape and
position


 % step 4 - use insertShape() to display the matched box

 I3=insertShape(I,'Rectangle',shape,"LineWidth",2);
 imshow(I3)
```

```
 I4=insertShape(I,'Rectangle',nshape,"LineWidth",2);
 imshow(I4)
```

The correlation template matching fails when there is a sharpness change or orientation change to the image or template. The normalized correlation template matching works for all sharpness changes to the image or template. It fails when the orientation is changed.

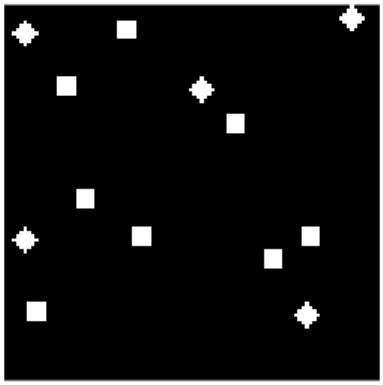
Problem 2 - binary image matching

```
BW  = zeros(100);
b = strel( 'diamond',3);
b = b.Neighborhood;
R = b;

x = [3, 3, 50, 78, 90 ];
y = [5, 60, 20, 80, 1 ];
for i = 1 : length( x )
    BW(y(i):y(i)+6,x(i):x(i)+6) = b;
end

b = strel( 'square',5);
b = b.Neighborhood;

x = [31, 35, 80, 7, 60, 15, 20, 70 ];
y = [5, 60, 60, 80, 30, 20, 50, 66 ];
for i = 1 : length( x )
    BW(y(i):y(i)+4,x(i):x(i)+4) = b;
end

% now, BW is your search image, R is your reference image
```

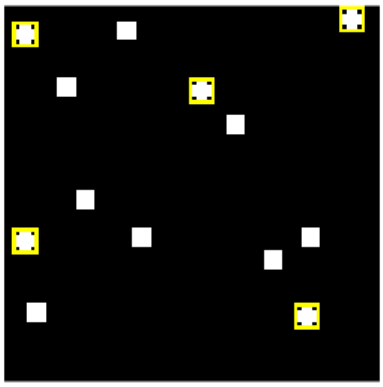
```
% write your code here

% use cross correlation from problem 1 to find matching, record time

tic
ncrr=normxcorr2(R,BW);
ncrr=ncrr(height(R):height(ncrr),width(R):width(ncrr));
t1=toc
```

```
t1 = 0.0226
```

```
[nxpeak,nypeak]=find(ncrr==max(ncrr(:)));
[row col]=size(R);
IX=BW;
for i=1:numel(nxpeak)
vectshape{i}=[(nypeak(i)) (nxpeak(i)) col row];
IX=insertShape(IX,'Rectangle',vectshape{i});
end
imshow(BW,"InitialMagnification",250)
```



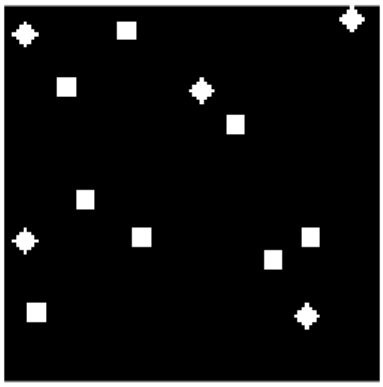
```
imshow(IX,"InitialMagnification",250)
```



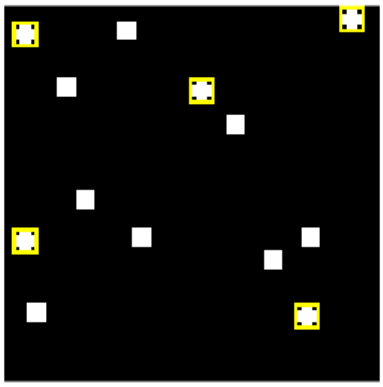
```
% use chamferMatch() to get the distance transform image Q
tic
Q=chamferMatch(BW,R);
t2=toc
```

```
t2 = 0.0637
```

```
[nxpeak,nypeak]=find(Q==min(Q(:)));
[row col]=size(R);
IX=BW;
for i=1:numel(nxpeak)
vectshape1{i}=[(nypeak(i)) (nxpeak(i)) col row];
IX=insertShape(IX,'Rectangle',vectshape1{i});
end
imshow(BW,"InitialMagnification",250)
```



```
imshow(IX,"InitialMagnification",250)
```



```
t1=sprintf('normalized cross-correlation time: %d',t1 );
t2=sprintf('ChamferMatch time: %d',t2);
disp(t1)
```

```
normalized cross-correlation time: 2.262330e-02
```

```
disp(t2)
```

```
ChamferMatch time: 6.374120e-02
```

The CPU time for the normalized cross-correlation is a little less than half of the CPU time for the ChamferMatch. The normalized cross-correlation is faster but under certain circumstances it can produce false positives when scanning binary images. Although

slower, the ChamferMatch is more accurate for binary images in general.