

# Assignment3

*by* Nikhil Samrat Gunda

---

**Submission date:** 02-Sep-2018 06:20PM (UTC+0800)

**Submission ID:** 995898762

**File name:** report.txt (15.49K)

**Word count:** 1521

**Character count:** 13709

```
%% MyMainScript

close all;

tic;

%% Your code here

im = load('./data/boat');

im = im.imageOrig;

im = mat2gray(im);

sigmas = 1;

sigmaw = 2;

k = 0.04;

% sigma for smoothing = 1

% sigma for weights in harris matrix = 2

% K = 0.04 empirical constant in corner-measure
```

```
I = myHarrisCornerDetector(im,sigmas,sigmaw,k);
```

```
toc;
```

```
function I = myHarrisCornerDetector(im,sgs,sgw,k)
```

```
G1 = fspecial('gaussian', [3*sgs 3*sgs] ,sgs);
```

```
I1 = imfilter(im, G1);
```

```
[x,y] = gradient(I1);
```

```
imshow(mat2gray(x))
```

```
title('x-gradient image')
```

```
figure
```

```
imshow(mat2gray(y))
```

```
title('y-gradient image')
```

```
xx = x.*x;
```

```
xy = x.*y;
```

```
yy = y.*y;
```

```
G = fspecial('gaussian', [3*sgw 3*sgw],sgw);
```

```
xx = imfilter(xx, G);
```

```
xy = imfilter(xy, G);
```

```
yy = imfilter(yy, G);
```

```
for i=1:size(l1,1)
```

```
    for j=1:size(l1,2)
```

```
        z = [xx(i,j),xy(i,j);xy(i,j),yy(i,j)];
```

```
        A(i,j) = min(eig(z));
```

```
        B(i,j) = max(eig(z));
```

```
        C(i,j) = det(z) - k*trace(z)*trace(z);
```

```
        if(C(i,j)*100000>1)
```

```
            C(i,j) = 1;
```

```
        else

            C(i,j) = 0;

        end

    end

end

end

my_display(A,'min Eigen-Valued Image');

my_display(B,'max Eigen-Valued Image');

my_display(C,'Harris Response Image');

I = C;

function my_display(s1,res)

    myNumOfColors = 256;

    myColorScale = zeros(myNumOfColors,3);
```

```

myColorScale(:,1) = 0:1/(myNumOfColors-1):1;% ,[0:1/(myNumOfColors-
1):1] ,[0:1/(myNumOfColors-1):1] ];

myColorScale(:,2) = 0:1/(myNumOfColors-1):1;

myColorScale(:,3) = 0:1/(myNumOfColors-1):1;

%

if size(s1,3)==1

% colormap jet;

colormap (myColorScale);

figure

imshow(mat2gray(s1)),colorbar

axis on;

title(res);

end

```

```
daspect ([1 1 1]);
```

```
axis tight;
```

```
function my_display(s1,res)
```

```
myNumOfColors = 256;
```

```
myColorScale = zeros(myNumOfColors,3);
```

```
myColorScale(:,1) = 0:1/(myNumOfColors-1):1;% ,[0:1/(myNumOfColors-  
1):1] ,[0:1/(myNumOfColors-1):1]  ];
```

```
myColorScale(:,2) = 0:1/(myNumOfColors-1):1;
```

```
myColorScale(:,3) = 0:1/(myNumOfColors-1):1;
```

```
figure();
```

```
imagesc((s1));
```

```
axis on;
```

```
title(res);
```

```
%   imwrite(mat2gray(s1),res);

    if size(s1,3)==1

%       colormap jet;

        colormap (myColorScale);

        colorbar;

    end

    daspect ([1 1 1]);

    axis tight;

%% MyMainScript

tic;

%% Mean Shift Segmantation for baboonColor

%%% Tuned Sigma_Spatial = 8

%%% Tuned Sigma_Intensity = 0.16 (equivalent to 41 in 0-255 intensity space)
```



```
%%% NumberofNeighbours Considered = 100

%%% Number of Iterations = 20

img_raw = imread('../data/baboonColor.png');

img_smooth = imgaussfilt(mat2gray(img_raw),1);

[M,N,C] = size(img_smooth);

img = img_smooth(1:2:M,1:2:N,:);

my_display(img,'orig.png');

noOfNeighbours=200;

sigma_S = 8;

sigma_I = 0.16;

max_iter = 20;

% res= myMeanShiftSegmentation(img,t,sigma_I,sigma_S,max_iter,noOfNeighbours);

my_display(res(:,1:3),'Segemnted imgae');
```

```

toc;

function res_img =

myMeanShiftSegmentation(img,t,sigma_l,sigma_x,n_iter,noOfNeighbours)

[M,N,C] = size(img);

temp_img = zeros(M,N,C+2).*1.0;

temp_img(:,1:C) = img;

temp1_img = zeros(M*N,5);

for i = 1:1:M

    for j = 1:1:N

        temp_img(i,j,4) = i;

        temp_img(i,j,5) = j;

        temp1_img((i-1)*N+j,:) = temp_img(i,j,:);

    end

```

```

end

temp1_img(:,1:3) = temp1_img(:,1:3)./(sigma_l*sqrt(2));

temp1_img(:,4:5) = temp1_img(:,4:5)./(sigma_x*sqrt(2));

for k=1:n_iter

    tic;

    [IDX, D] = knnsearch(temp1_img, temp1_img, 'k', noOfNeighbours);

    for i=1:M*N

        weights = exp(-(D(i,:).^2));

        for j=1:5

            v = (sum(sum(temp1_img(IDX(i,:),j).*weights))/sum(sum(weights)));

            u = temp1_img(i,j);

            temp1_img(i,j) = temp1_img(i,j) +

            t*((sum(sum(temp1_img(IDX(i,:),j).*weights))/sum(sum(weights)))-temp1_img(i,j));

```

```

        z = temp1_img(i,j);

    end

end

display(strcat('iter',num2str(k),' complete'));

toc;

end

for i = 1:1:M

    for j = 1:1:N

        res_img(i,j,1:3) = (temp1_img((i-1)*N+j,1:3)).*(sigma_l*sqrt(2));

        res_img(i,j,4:5) = (temp1_img((i-1)*N+j,4:5)).*(sigma_x*sqrt(2));

    end

end

end

<!DOCTYPE html

```

PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

<html><head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8">

<!--

This HTML was auto-generated from MATLAB code.

To make changes, update the MATLAB code and republish this document.

--><title>myMainScript</title><meta name="generator" content="MATLAB  
8.6"><link rel="schema.DC" href="http://purl.org/dc/elements/1.1/"><meta  
name="DC.date" content="2018-09-02"><meta name="DC.source"  
content="myMainScript.m"><style type="text/css">

html,body,div,span,applet,object,iframe,h1,h2,h3,h4,h5,h6,p,blockquote,pre,a,abbr,acron  
ym,address,big,cite,code,del,dfn,em,font,img,ins,kbd,q,s,samp,small,strike,strong,sub,sup  
,tt,var,b,u,i,center,dl,dt,dd,ol,ul,li,fieldset,form,label,legend,table,caption,tbody,tfoot,thea  
d,tr,th,td{margin:0;padding:0;border:0;outline:0;font-size:100%;vertical-

```
align:baseline;background:transparent}body{line-height:1}ol,ul{list-  
style:none}blockquote,q{quotes:none}blockquote:before,blockquote:after,q:before,q:afte  
r{content:"";content:none}:focus{outline:0}ins{text-decoration:none}del{text-  
decoration:line-through}table{border-collapse:collapse;border-spacing:0}  
  
html { min-height:100%; margin-bottom:1px; }  
  
html body { height:100%; margin:0px; font-family:Arial, Helvetica, sans-serif; font-  
size:10px; color:#000; line-height:140%; background:#fff none; overflow-y:scroll; }  
  
html body td { vertical-align:top; text-align:left; }  
  
h1 { padding:0px; margin:0px 0px 25px; font-family:Arial, Helvetica, sans-serif; font-  
size:1.5em; color:#d55000; line-height:100%; font-weight:normal; }  
  
h2 { padding:0px; margin:0px 0px 8px; font-family:Arial, Helvetica, sans-serif; font-  
size:1.2em; color:#000; font-weight:bold; line-height:140%; border-bottom:1px solid  
#d6d4d4; display:block; }
```

```
h3 { padding:0px; margin:0px 0px 5px; font-family:Arial, Helvetica, sans-serif; font-size:1.1em; color:#000; font-weight:bold; line-height:140%; }
```

```
a { color:#005fce; text-decoration:none; }
```

```
a:hover { color:#005fce; text-decoration:underline; }
```

```
a:visited { color:#004aa0; text-decoration:none; }
```

```
p { padding:0px; margin:0px 0px 20px; }
```

```
img { padding:0px; margin:0px 0px 20px; border:none; }
```

```
p img, pre img, tt img, li img, h1 img, h2 img { margin-bottom:0px; }
```

```
ul { padding:0px; margin:0px 0px 20px 23px; list-style:square; }
```

```
ul li { padding:0px; margin:0px 0px 7px 0px; }
```

```
ul li ul { padding:5px 0px 0px; margin:0px 0px 7px 23px; }
```

```
ul li ol li { list-style:decimal; }
```

```
ol { padding:0px; margin:0px 0px 20px 0px; list-style:decimal; }
```

```
ol li { padding:0px; margin:0px 0px 7px 23px; list-style-type:decimal; }
```

```
ol li ol { padding:5px 0px 0px; margin:0px 0px 7px 0px; }
```

```
ol li ol li { list-style-type:lower-alpha; }
```

```
ol li ul { padding-top:7px; }
```

```
ol li ul li { list-style:square; }
```

```
.content { font-size:1.2em; line-height:140%; padding: 20px; }
```

```
pre, code { font-size:12px; }
```

```
tt { font-size: 1.2em; }
```

```
pre { margin:0px 0px 20px; }
```

```
pre.codeinput { padding:10px; border:1px solid #d3d3d3; background:#f7f7f7; }
```

```
pre.codeoutput { padding:10px 11px; margin:0px 0px 20px; color:#4c4c4c; }
```

```
pre.error { color:red; }
```

```
@media print { pre.codeinput, pre.codeoutput { word-wrap:break-word; width:100%; } }
```



```
span.keyword { color:#0000FF }
```

```
span.comment { color:#228B22 }
```

```
span.string { color:#A020F0 }
```

```
span.undermstring { color:#B20000 }
```

```
span.syscmd { color:#B28C00 }
```

```
.footer { width:auto; padding:10px 0px; margin:25px 0px 0px; border-top:1px dotted
```

```
#878787; font-size:0.8em; line-height:140%; font-style:italic; color:#878787; text-
```

```
align:left; float:none; }
```

```
.footer p { margin:0px; }
```

```
.footer a { color:#878787; }
```

```
.footer a:hover { color:#878787; text-decoration:underline; }
```

```
.footer a:visited { color:#878787; }
```

```
table th { padding:7px 5px; text-align:left; vertical-align:middle; border: 1px solid
```

```
#d6d4d4; font-weight:bold; }
```

```
table td { padding:7px 5px; text-align:left; vertical-align:top; border:1px solid #d6d4d4; }
```

```
</style> </head> <body> <div class="content"> <h2>Contents</h2> <div> <ul> <li> <a
```

```
href="#1">MyMainScript</a> </li> <li> <a href="#2">Your code
```

```
here</a> </li> </ul> </div> <h2>MyMainScript<a name="1"> </a> </h2> <pre
```

```
class="codeinput">close <span class="string">all</span>;
```

```
tic;
```

```
</pre> <h2>Your code here<a name="2"> </a> </h2> <pre class="codeinput">im =
```

```
load(<span class="string">'../data/boat'</span>);
```

```
im = im.imageOrig;
```

```
im = mat2gray(im);
```

```
sigmas = 1;
```

```
sigmaw = 2;
```

```
k = 0.04;
```

```
<span class="comment">% sigma for smoothing = 1</span>
```

```
<span class="comment">% sigma for weights in harris matrix = 2</span>
```

```
<span class="comment">% K = 0.04 empirical constant in corner-measure</span>
```

```
I = myHarrisCornerDetector(im,sigmas,sigmaw,k);
```

```
toc;
```

```
</pre> <pre class="codeoutput">Elapsed time is 3.956546 seconds.
```

```
</pre>      <p class="footer"><br><a
```

```
href="http://www.mathworks.com/products/matlab/">Published with MATLAB®
```

```
R2015b</a><br></p></div><!--
```

```
##### SOURCE BEGIN #####
```

```
%% MyMainScript
```

```
close all;
```

```
tic;
```

```
%% Your code here
```

```
im = load('./data/boat');
```

```
im = im.imageOrig;
```

```
im = mat2gray(im);
```

```
sigmas = 1;
```

```
sigmaw = 2;
```

```
k = 0.04;
```

```
% sigma for smoothing = 1
```

```
% sigma for weights in harris matrix = 2
```

% K = 0.04 empirical constant in corner-measure

I = myHarrisCornerDetector(im,sigmas,sigmaw,k);

toc;

##### SOURCE END #####

--></body></html>

<!DOCTYPE html

PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">

<html><head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8">

<!--

This HTML was auto-generated from MATLAB code.

To make changes, update the MATLAB code and republish this document.

```

--> <title>myMainScript</title> <meta name="generator" content="MATLAB
8.6"> <link rel="schema.DC" href="http://purl.org/dc/elements/1.1/"> <meta
name="DC.date" content="2018-09-02"> <meta name="DC.source"
content="myMainScript.m"> <style type="text/css">

html,body,div,span,applet,object,iframe,h1,h2,h3,h4,h5,h6,p,blockquote,pre,a,abbr,acron
ym,address,big,cite,code,del,dfn,em,font,img,ins,kbd,q,s,samp,small,strike,strong,sub,sup
,tt,var,b,u,i,center,dl,dt,dd,ol,ul,li,fieldset,form,label,legend,table,caption,tbody,tfoot,thea
d,tr,th,td{margin:0;padding:0;border:0;outline:0;font-size:100%;vertical-
align:baseline;background:transparent}body{line-height:1}ol,ul{list-
style:none}blockquote,q{quotes:none}blockquote:before,blockquote:after,q:before,q:afte
r{content:"";content:none}.focus{outline:0}ins{text-decoration:none}del{text-
decoration:line-through}table{border-collapse:collapse;border-spacing:0}

html { min-height:100%; margin-bottom:1px; }

html body { height:100%; margin:0px; font-family:Arial, Helvetica, sans-serif; font-
size:10px; color:#000; line-height:140%; background:#fff none; overflow-y:scroll; }

```

```
html body td { vertical-align:top; text-align:left; }
```

```
h1 { padding:0px; margin:0px 0px 25px; font-family:Arial, Helvetica, sans-serif; font-size:1.5em; color:#d55000; line-height:100%; font-weight:normal; }
```

```
h2 { padding:0px; margin:0px 0px 8px; font-family:Arial, Helvetica, sans-serif; font-size:1.2em; color:#000; font-weight:bold; line-height:140%; border-bottom:1px solid #d6d4d4; display:block; }
```

```
h3 { padding:0px; margin:0px 0px 5px; font-family:Arial, Helvetica, sans-serif; font-size:1.1em; color:#000; font-weight:bold; line-height:140%; }
```

```
a { color:#005fce; text-decoration:none; }
```

```
a:hover { color:#005fce; text-decoration:underline; }
```

```
a:visited { color:#004aa0; text-decoration:none; }
```

```
p { padding:0px; margin:0px 0px 20px; }
```

```
img { padding:0px; margin:0px 0px 20px; border:none; }
```

```
p img, pre img, tt img, li img, h1 img, h2 img { margin-bottom:0px; }
```

```
ul { padding:0px; margin:0px 0px 20px 23px; list-style:square; }
```

```
ul li { padding:0px; margin:0px 0px 7px 0px; }
```

```
ul li ul { padding:5px 0px 0px; margin:0px 0px 7px 23px; }
```

```
ul li ol li { list-style:decimal; }
```

```
ol { padding:0px; margin:0px 0px 20px 0px; list-style:decimal; }
```

```
ol li { padding:0px; margin:0px 0px 7px 23px; list-style-type:decimal; }
```

```
ol li ol { padding:5px 0px 0px; margin:0px 0px 7px 0px; }
```

```
ol li ol li { list-style-type:lower-alpha; }
```

```
ol li ul { padding-top:7px; }
```

```
ol li ul li { list-style:square; }
```

```
.content { font-size:1.2em; line-height:140%; padding: 20px; }
```

```
pre, code { font-size:12px; }
```



```
tt { font-size: 1.2em; }
```

```
pre { margin:0px 0px 20px; }
```

```
pre.codeinput { padding:10px; border:1px solid #d3d3d3; background:#f7f7f7; }
```

```
pre.codeoutput { padding:10px 11px; margin:0px 0px 20px; color:#4c4c4c; }
```

```
pre.error { color:red; }
```

```
@media print { pre.codeinput, pre.codeoutput { word-wrap:break-word; width:100%; } }
```

```
span.keyword { color:#0000FF }
```

```
span.comment { color:#228B22 }
```

```
span.string { color:#A020F0 }
```

```
span.undermstring { color:#B20000 }
```

```
span.syscmd { color:#B28C00 }
```

```
.footer { width:auto; padding:10px 0px; margin:25px 0px 0px; border-top:1px dotted  
#878787; font-size:0.8em; line-height:140%; font-style:italic; color:#878787; text-  
align:left; float:none; }
```

```
.footer p { margin:0px; }
```

```
.footer a { color:#878787; }
```

```
.footer a:hover { color:#878787; text-decoration:underline; }
```

```
.footer a:visited { color:#878787; }
```

```
table th { padding:7px 5px; text-align:left; vertical-align:middle; border: 1px solid  
#d6d4d4; font-weight:bold; }
```

```
table td { padding:7px 5px; text-align:left; vertical-align:top; border:1px solid #d6d4d4; }
```

```
</style> </head> <body> <div class="content"> <h2>Contents</h2> <div> <ul> <li> <a  
href="#1">MyMainScript</a> </li> <li> <a href="#2">Mean Shift Segmantation for  
baboonColor</a> </li> <li> <a href="#3">Tuned Sigma_Spatial = 8</a> </li> <li> <a  
href="#4">Tuned Sigma_Intensity = 0.16 (equivalent to 41 in 0-255 intensity
```

space)</a></li><li><a href="#5">NumberofNeighbours Considered =

100</a></li><li><a href="#6">Number of Iterations =

20</a></li></ul></div><h2>MyMainScript<a name="1"></a></h2><pre

class="codeinput">tic;

</pre><h2>Mean Shift Segmantation for baboonColor<a

name="2"></a></h2><h2>Tuned Sigma\_Spatial = 8<a

name="3"></a></h2><h2>Tuned Sigma\_Intensity = 0.16 (equivalent to 41 in 0-255

intensity space)<a name="4"></a></h2><h2>NumberofNeighbours Considered =

100<a name="5"></a></h2><h2>Number of Iterations = 20<a

name="6"></a></h2><pre class="codeinput">img\_raw = imread(<span

class="string">'../data/baboonColor.png'</span>);

img\_smooth = imgaussfilt(mat2gray(img\_raw),1);

[M,N,C] = size(img\_smooth);

img = img\_smooth(1:2:M,1:2:N,:);

```
my_display(img,<span class="string">'orig.png'</span>);
```

```
noOfNeighbours=200;
```

```
sigma_S = 8;
```

```
sigma_I = 0.16;
```

```
max_iter = 20;
```

```
res= myMeanShiftSegmentation(img,t,sigma_I,sigma_S,max_iter,noOfNeighbours);
```

```
my_display(res(:,1:3),<span class="string">'Segemnted imgae'</span>);
```

```
toc;
```

```
</pre><pre class="codeoutput">Elapsed time is 0.116068 seconds.
```

```
</pre>  <p class="footer"><br><a
```

```
href="http://www.mathworks.com/products/matlab/">Published with MATLAB®;
```

```
R2015b</a><br></p></div><!--
```

```
##### SOURCE BEGIN #####
```

```
%% MyMainScript
```

```
tic;
```

```
%% Mean Shift Segmantation for baboonColor
```

```
%%% Tuned Sigma_Spatial = 8
```

```
%%% Tuned Sigma_Intensity = 0.16 (equivalent to 41 in 0-255 intensity space)
```

```
%%% NumberofNeighbours Considered = 100
```

```
%%% Number of Iterations = 20
```

```
img_raw = imread('../data/baboonColor.png');
```

```
img_smooth = imgaussfilt(mat2gray(img_raw),1);
```

```
[M,N,C] = size(img_smooth);
```

```
img = img_smooth(1:2:M,1:2:N,:);
```

```
my_display(img,'orig.png');
```

```
noOfNeighbours=200;
```

```
sigma_S = 8;
```

```
sigma_I = 0.16;
```

```
max_iter = 20;
```

```
% res= myMeanShiftSegmentation(img,t,sigma_I,sigma_S,max_iter,noOfNeighbours);
```

```
my_display(res(:,:,1:3),'Segemnted imgae');
```

```
toc;
```

```
##### SOURCE END #####
```

```
--> </body> </html>
```

# Assignment3

---

PAGE 1

---

PAGE 2

---

PAGE 3

---

PAGE 4

---

PAGE 5

---

PAGE 6

---

PAGE 7

---

PAGE 8

---

PAGE 9

---

PAGE 10

---

PAGE 11

---

PAGE 12

---

PAGE 13

---

PAGE 14

---

PAGE 15

---

PAGE 16

---

PAGE 17

---

PAGE 18

---

PAGE 19

---

PAGE 20

---

PAGE 21

---

PAGE 22

---

PAGE 23

---

PAGE 24

---

PAGE 25

---

PAGE 26

---

PAGE 27

---

PAGE 28

---

PAGE 29

---