EXPERIMENT 5

Aim : To Study and Practice Blurring and smoothing of the Image using different type of and different sized filters.

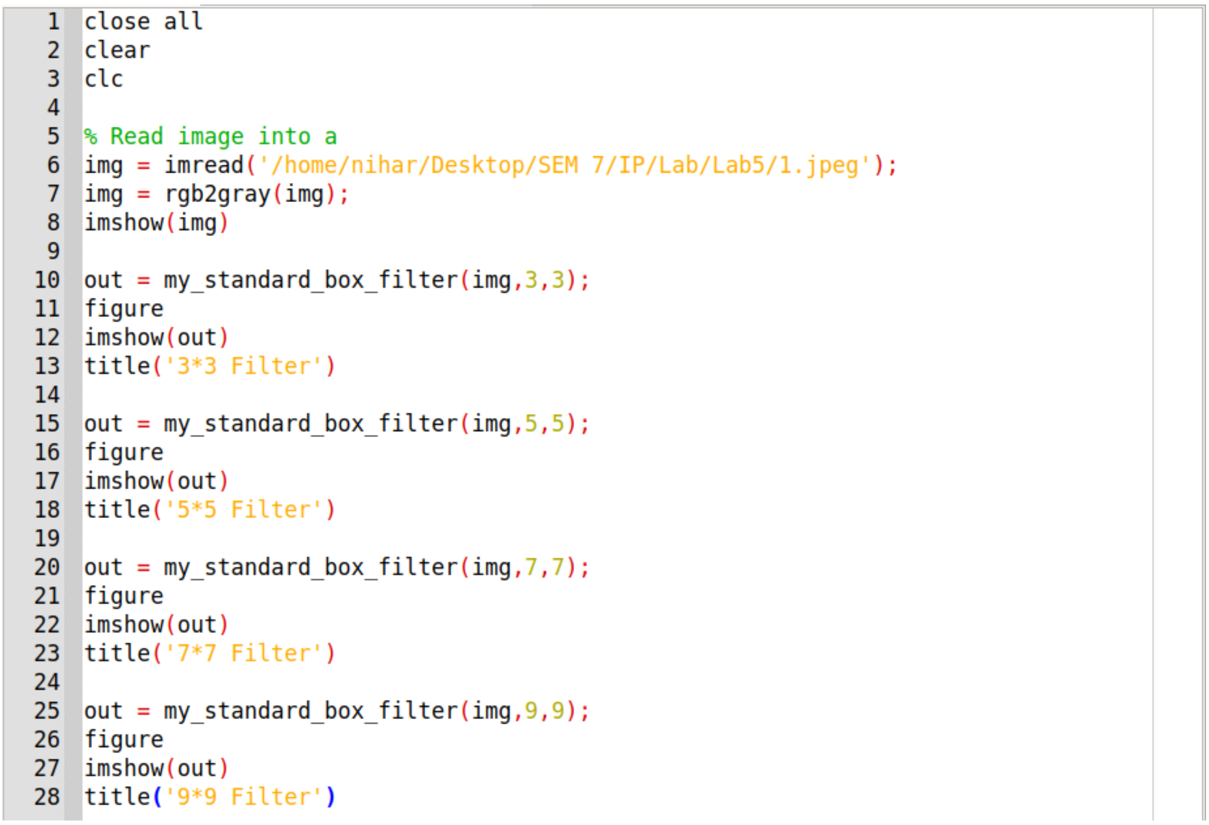
❖ Exercises :

1. **Take any of your gray scale photo and blur it with standard box filter of size 3x3, 5x5 ,7x7 and 9x9. Comment on amount of blurring and filter size. Assume padding of zeros.**

→ **Solution :-**

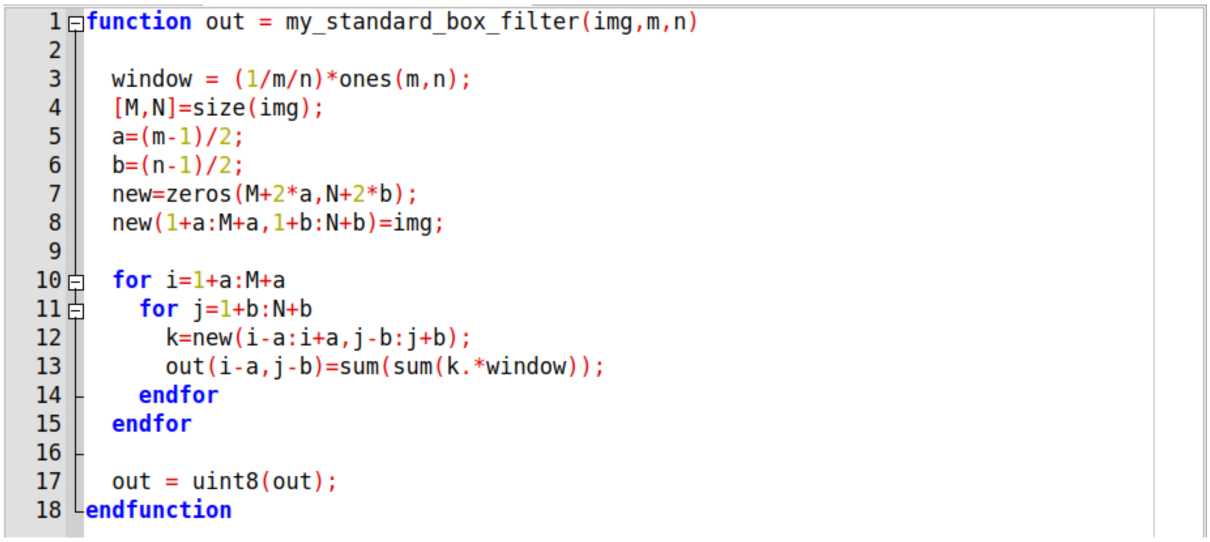
- As filter size increases blurring will increases.
- As filter size increases pixel will be affected by more surrounding pixels so blurring will increases.

Code :



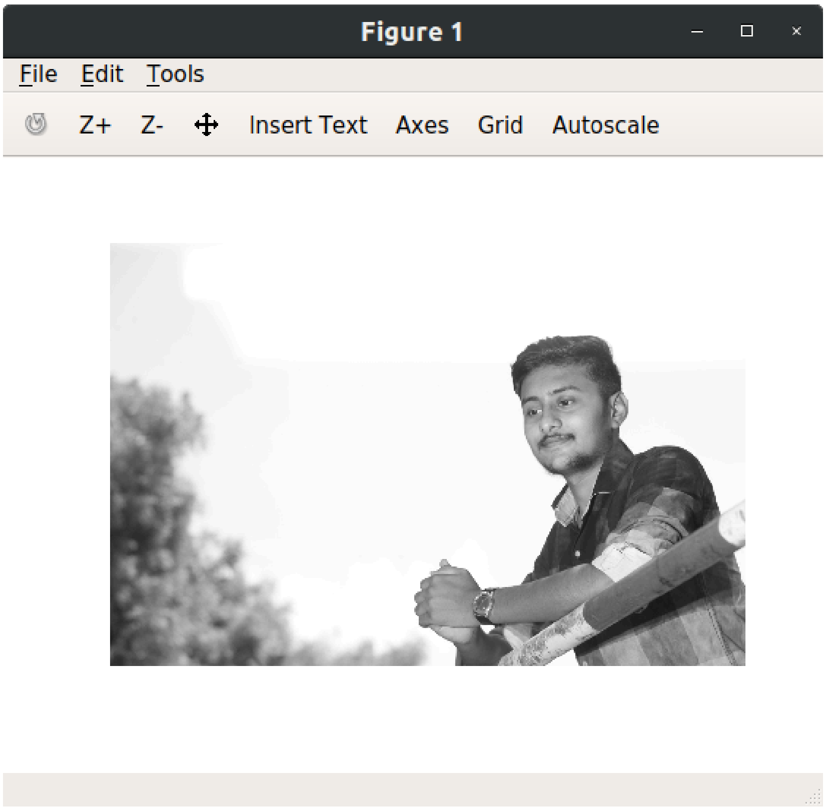
```
close all
clear
clc

% Read image into a
img = imread('/home/nihar/Desktop/SEM 7/IP/Lab/Lab5/1.jpeg');
img = rgb2gray(img);
imshow(img)

out = my_standard_box_filter(img,3,3);
figure
imshow(out)
title('3*3 Filter')

out = my_standard_box_filter(img,5,5);
figure
imshow(out)
title('5*5 Filter')

out = my_standard_box_filter(img,7,7);
figure
imshow(out)
title('7*7 Filter')

out = my_standard_box_filter(img,9,9);
figure
imshow(out)
title('9*9 Filter')
```

Function Code for Standard Box Filter :

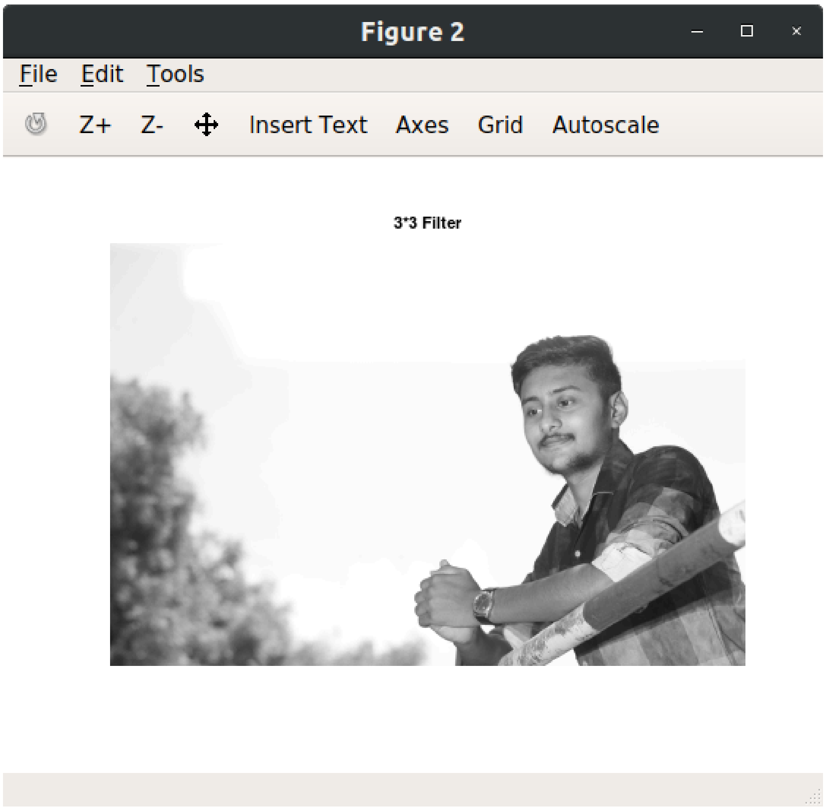


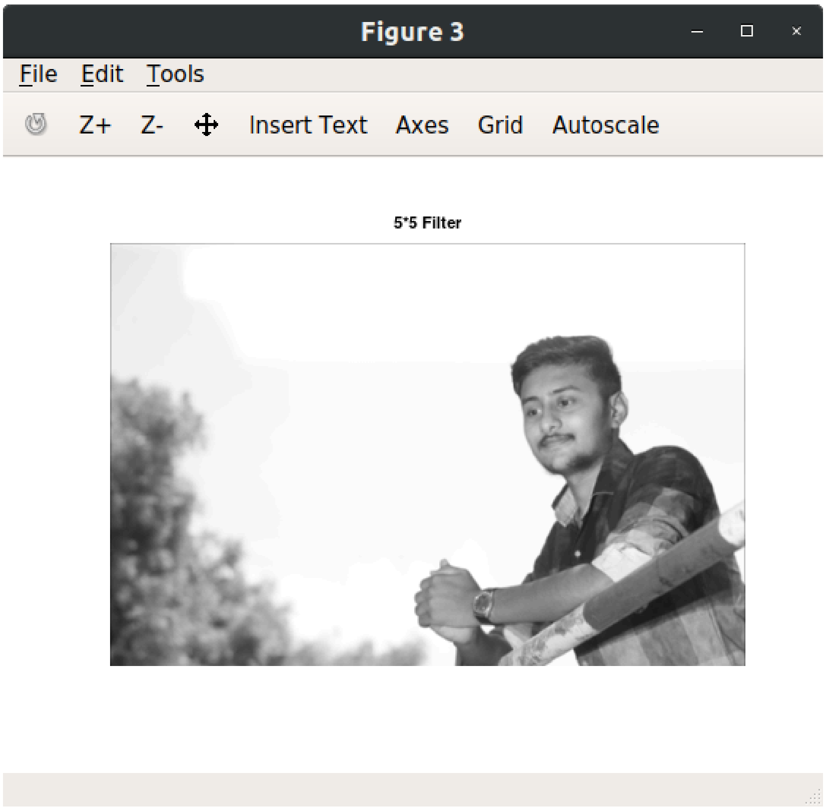
```
function out = my_standard_box_filter(img,m,n)

  window = (1/m/n)*ones(m,n);
  [M,N]=size(img);
  a=(m-1)/2;
  b=(n-1)/2;
  new=zeros(M+2*a,N+2*b);
  new(1+a:M+a,1+b:N+b)=img;

  for i=1+a:M+a
    for j=1+b:N+b
      k=new(i-a:i+a,j-b:j+b);
      out(i-a,j-b)=sum(sum(k.*window));
    endfor
  endfor

  out = uint8(out);
endfunction
```

Input Image :



Output Images :





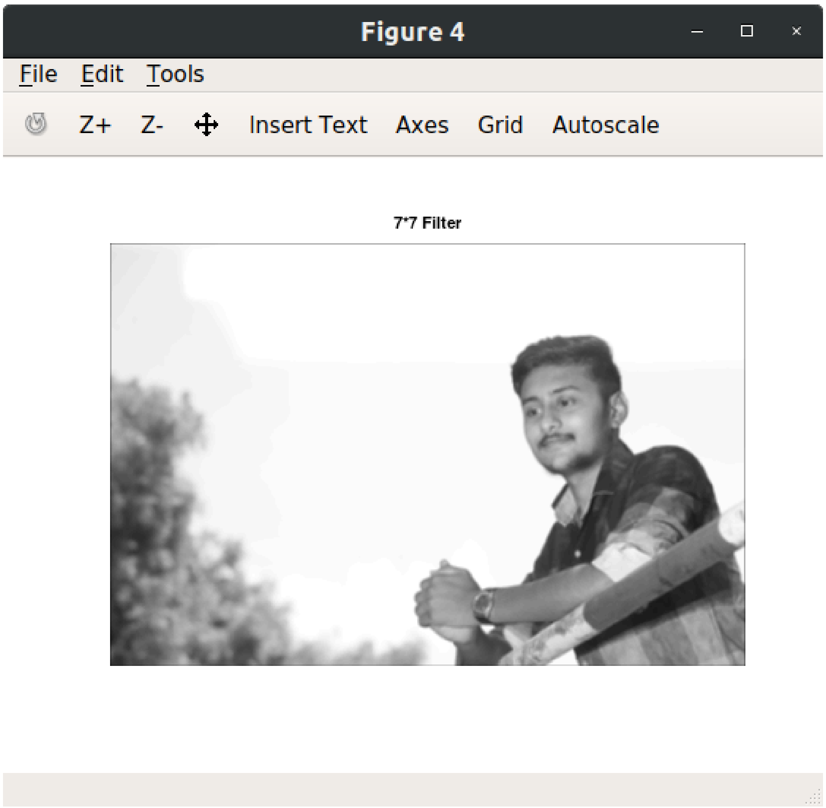


Figure 4
File
Edit
Tools
Z+
Z-
Insert Text
Axes
Grid
Autoscale
7*7 Filter

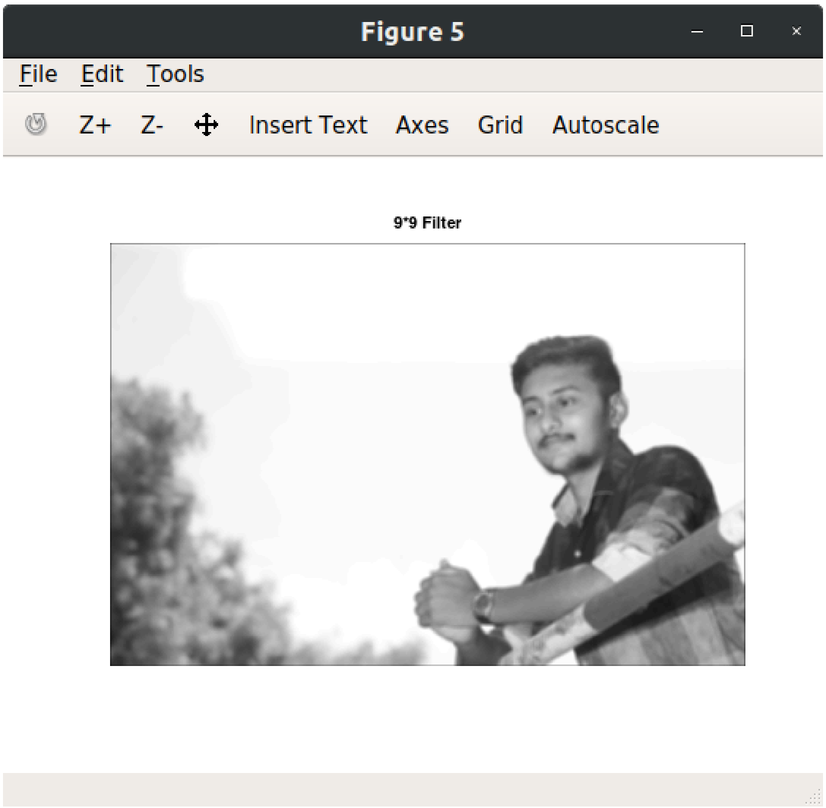


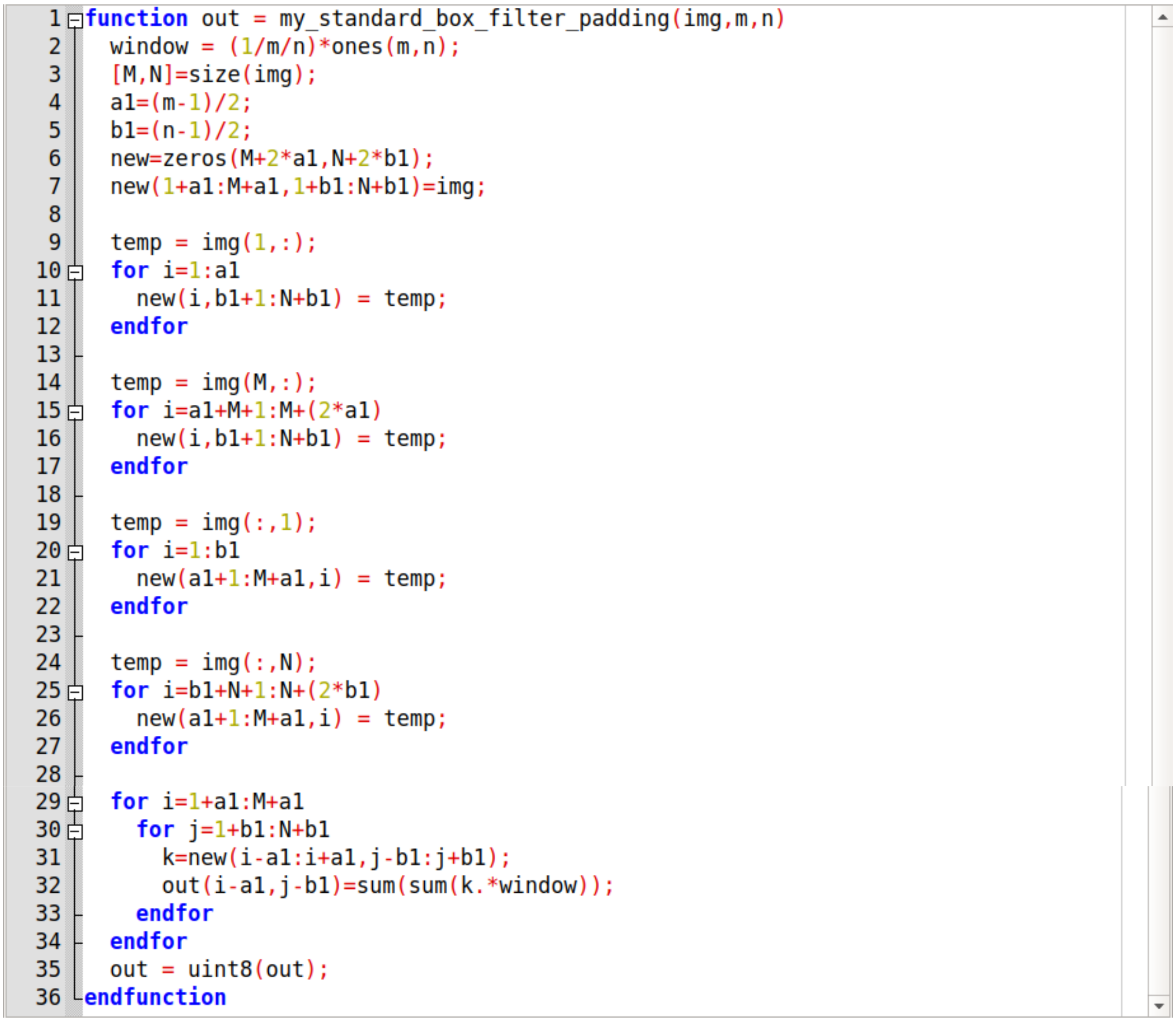
Figure 5
File
Edit
Tools
Z+
Z-
Insert Text
Axes
Grid
Autoscale
9*9 Filter

2. **Observe border of image for results in (a). Justify the reason for dark borders. Comment on thickness of the border and filter size. Suggest a way to solve the issue. Implement your suggestion and show the code and results.**

→ **Solution :-**

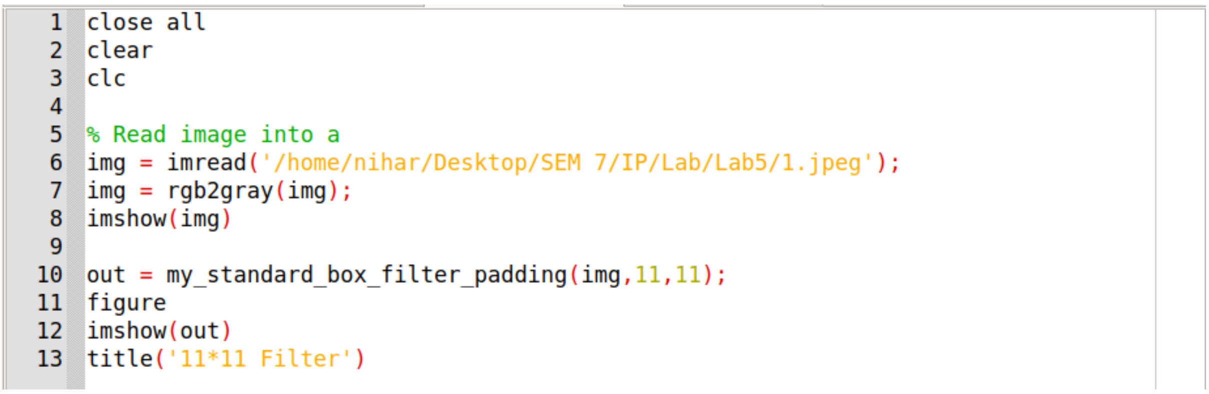
- Dark borders in the image because of padding with 0**.**
- Thickness of border will increases as filter size increases.
- We can solve this error by padding with same values as border has Instead of 0.

Function Code for Standard Box Filter padding with same as border values :



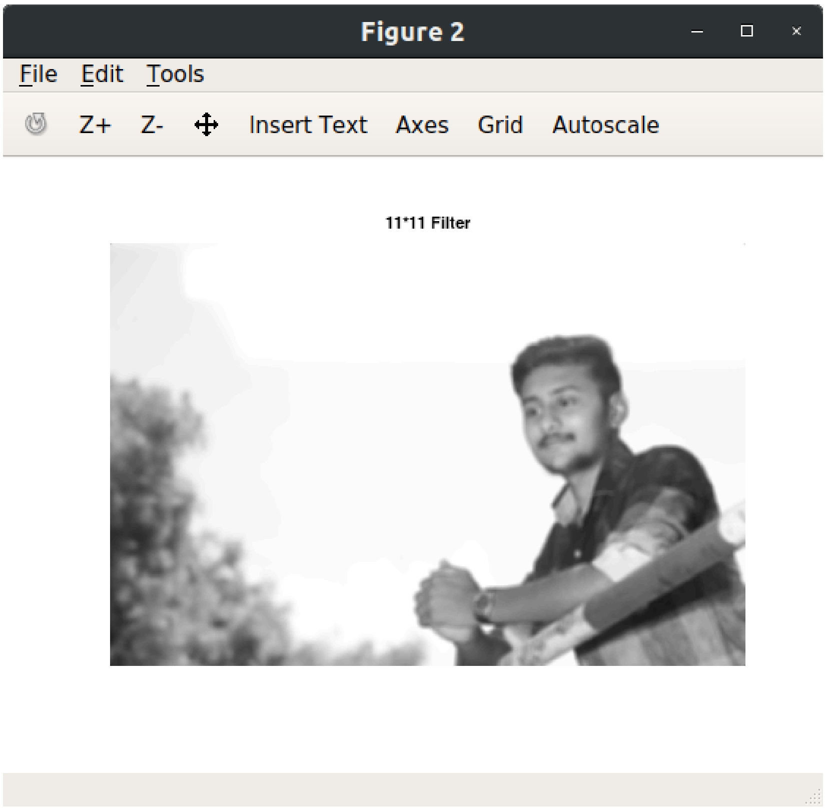
```
function out = my_standard_box_filter_padding(img,m,n)
  window = (1/m/n)*ones(m,n);
  [M,N]=size(img);
  a1=(m-1)/2;
  b1=(n-1)/2;
  new=zeros(M+2*a1,N+2*b1);
  new(1+a1:M+a1,1+b1:N+b1)=img;

  temp = img(1,:);
  for i=1:a1
    new(i,b1+1:N+b1) = temp;
  endfor

  temp = img(M,:);
  for i=a1+M+1:M+(2*a1)
    new(i,b1+1:N+b1) = temp;
  endfor

  temp = img(:,1);
  for i=1:b1
    new(a1+1:M+a1,i) = temp;
  endfor

  temp = img(:,N);
  for i=b1+N+1:N+(2*b1)
    new(a1+1:M+a1,i) = temp;
  endfor

  for i=1+a1:M+a1
    for j=1+b1:N+b1
      k=new(i-a1:i+a1,j-b1:j+b1);
      out(i-a1,j-b1)=sum(sum(k.*window));
    endfor
  endfor
  out = uint8(out);
endfunction
```

Code :



```
close all
clear
clc

% Read image into a
img = imread('/home/nihar/Desktop/SEM 7/IP/Lab/Lab5/1.jpeg');
img = rgb2gray(img);
imshow(img)

out = my_standard_box_filter_padding(img,11,11);
figure
imshow(out)
title('11*11 Filter')
```

Output Image :

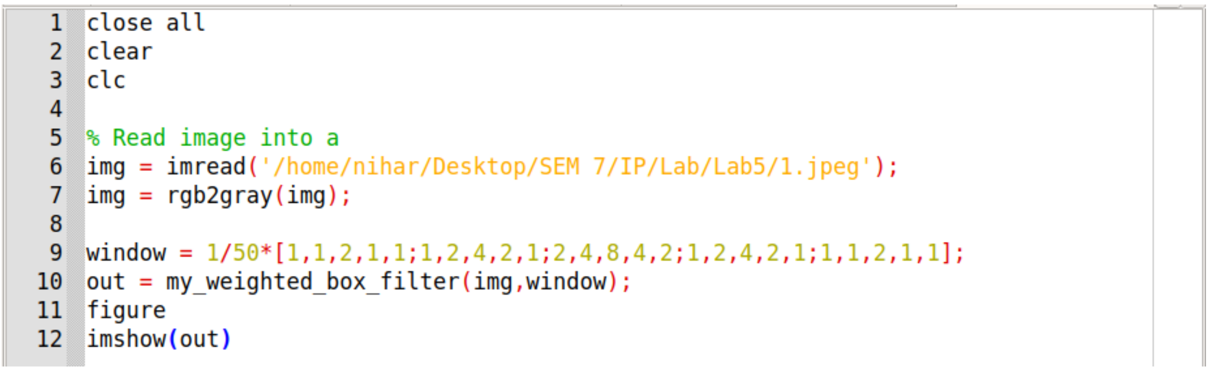


3. Take any of your gray scale photo and blur it with weighted average filter. Compare amount of blurring with the standard box filter of the same size.

→ Solution :-

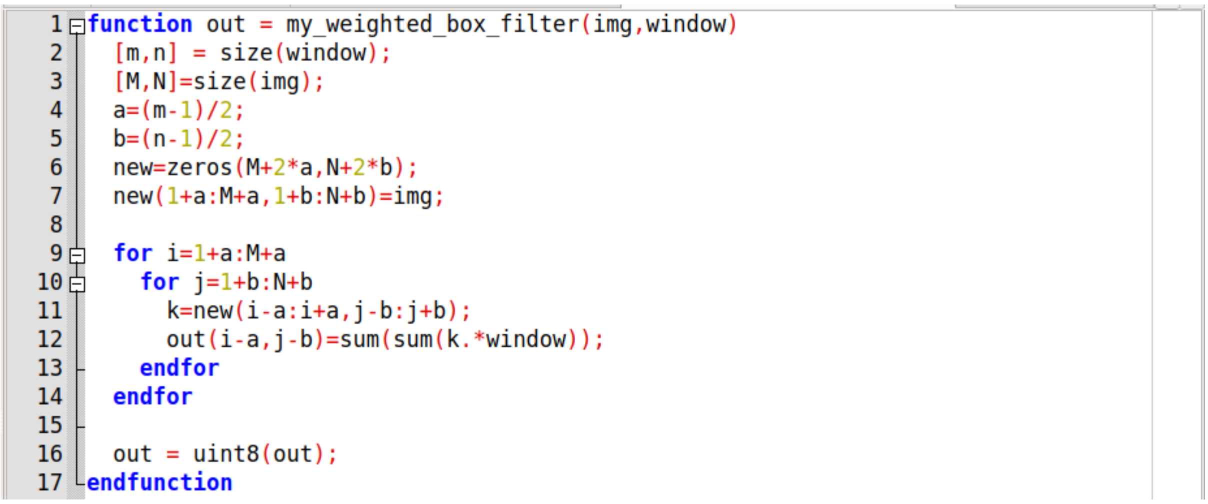
- In Weighted Box Filter blurring will be more than compare to the same size of Standard Box Filter.

Code :



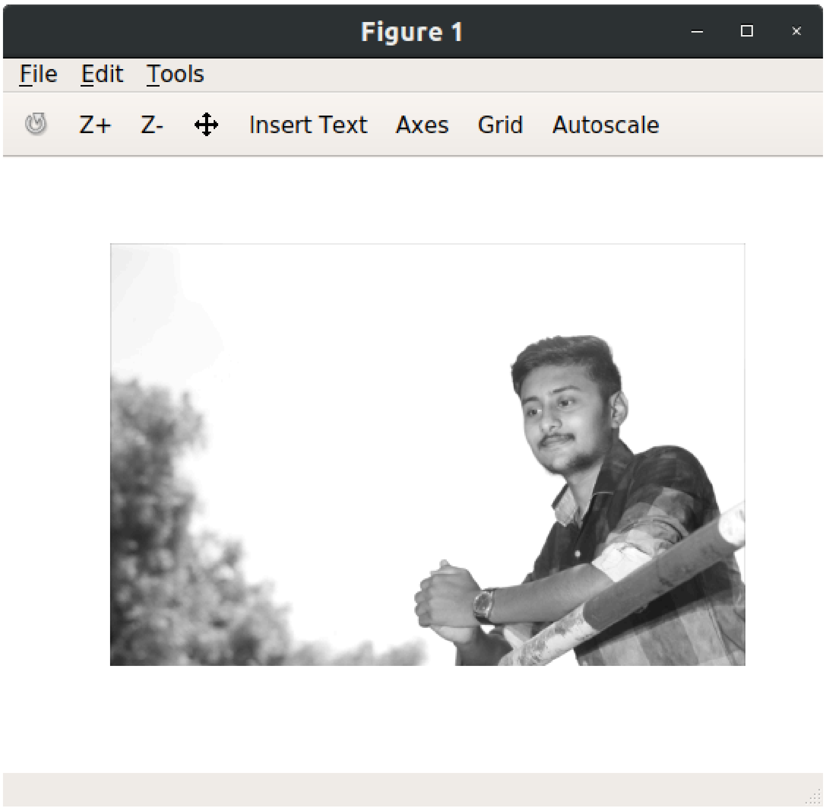
```
close all
clear
clc

% Read image into a
img = imread('/home/nihar/Desktop/SEM 7/IP/Lab/Lab5/1.jpeg');
img = rgb2gray(img);

window = 1/50*[1,1,2,1,1;1,2,4,2,1;2,4,8,4,2;1,2,4,2,1;1,1,2,1,1];
out = my_weighted_box_filter(img,window);
figure
imshow(out)
```

Function Code for Weighted Box Filter :



```
function out = my_weighted_box_filter(img,window)
  [m,n] = size(window);
  [M,N]=size(img);
  a=(m-1)/2;
  b=(n-1)/2;
  new=zeros(M+2*a,N+2*b);
  new(1+a:M+a,1+b:N+b)=img;

  for i=1+a:M+a
    for j=1+b:N+b
      k=new(i-a:i+a,j-b:j+b);
      out(i-a,j-b)=sum(sum(k.*window));
    endfor
  endfor

  out = uint8(out);
endfunction
```

Output Image :

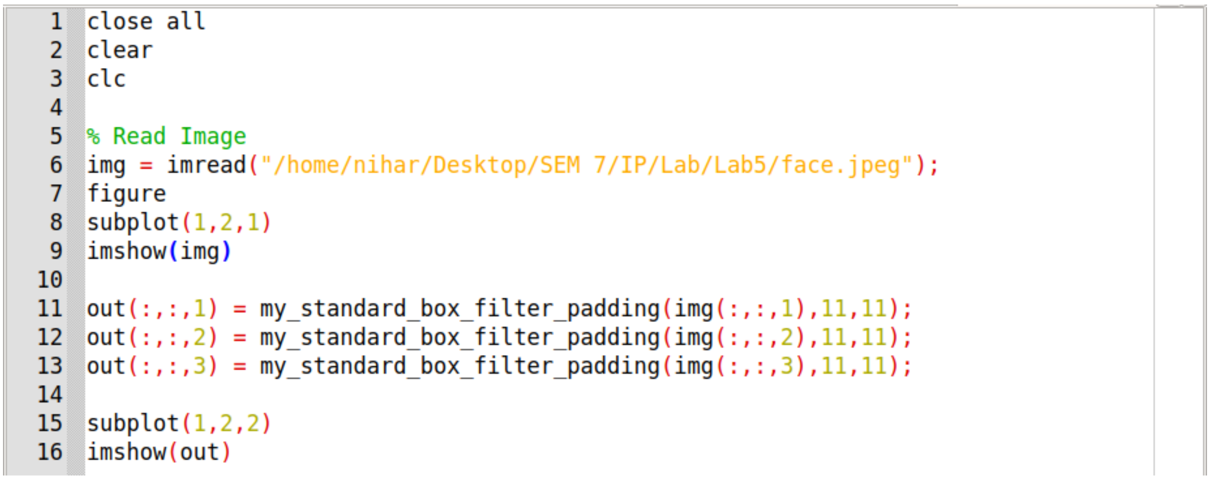


4. **Assume that you are working on some image enhancement application which gives following functionality to user.**
    1) **Anti-aging: Removes the wrinkles on the input face image.**
    2) **Beautify: Removes facial marks.**

   Take any of the color photo of a face and implement any (or both) of the above functionality.

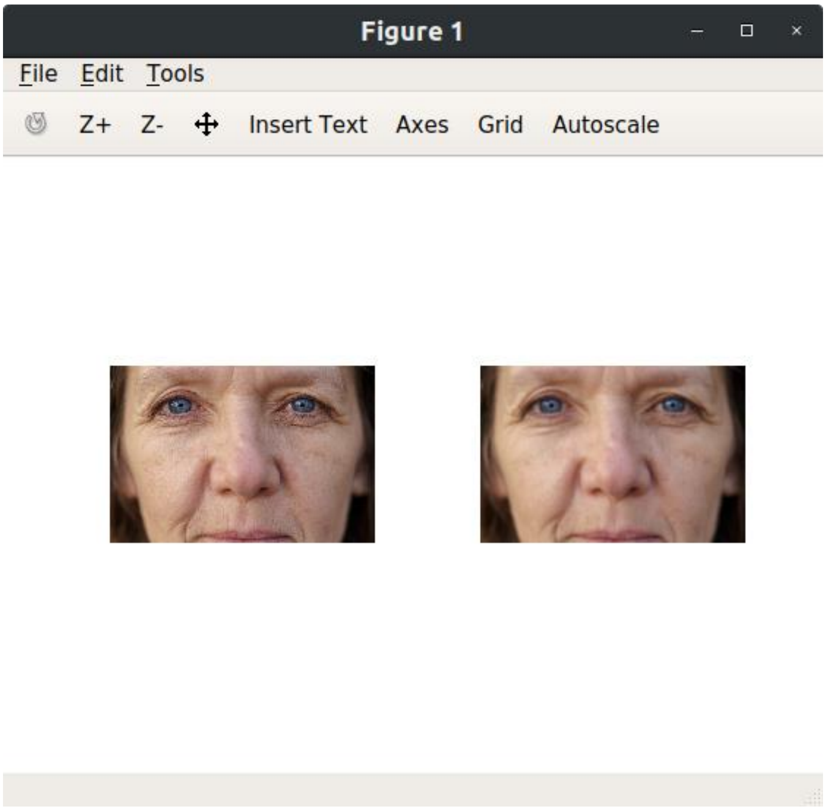
→ **Solution :-**

Code :



```
close all
clear
clc

% Read Image
img = imread("/home/nihar/Desktop/SEM 7/IP/Lab/Lab5/face.jpeg");
figure
subplot(1,2,1)
imshow(img)

out(:,:,1) = my_standard_box_filter_padding(img(:,:,1),11,11);
out(:,:,2) = my_standard_box_filter_padding(img(:,:,2),11,11);
out(:,:,3) = my_standard_box_filter_padding(img(:,:,3),11,11);

subplot(1,2,2)
imshow(out)
```

Output :

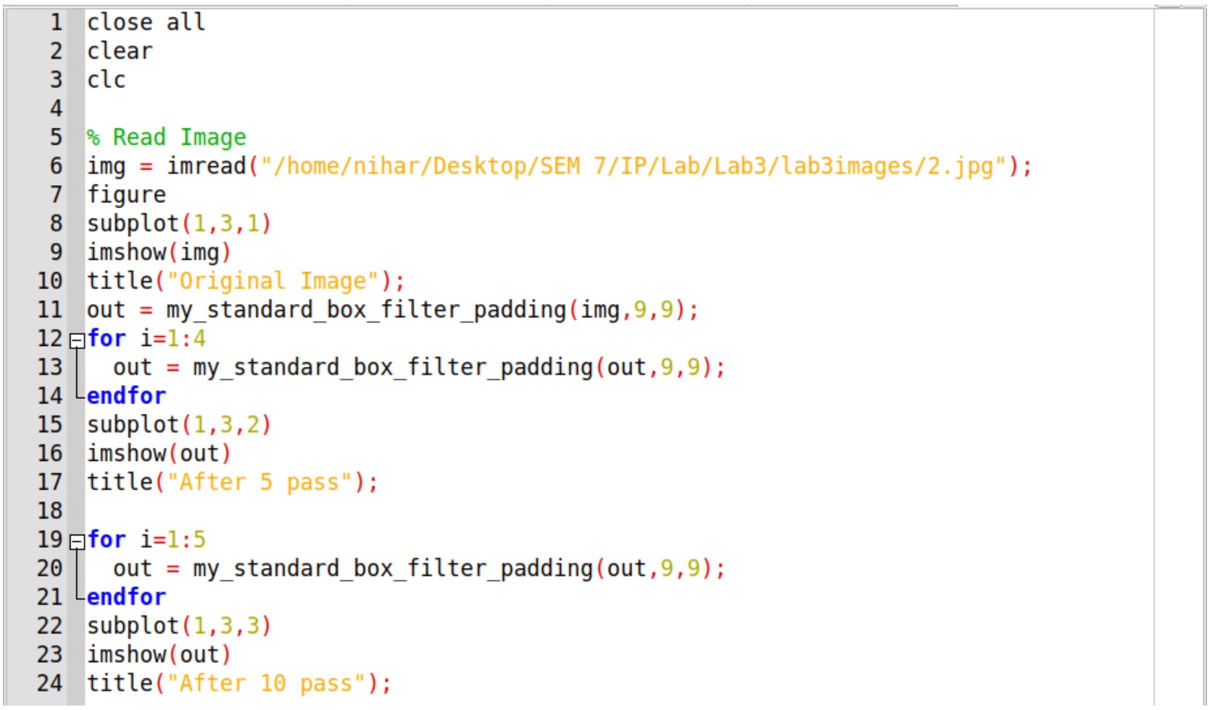


5. **Show the impact of multiple passes of the smoothing filter of same size. Derive your conclusion on image quality and maximum number of passes of filter? What happens if infinite(read very high!) number of passes are applied? Will it change image quality?**

→ **Solution :-**

- As same sized filter applied on the image multiple times than image is getting more blurred.
- If filter is applied infinite times on image than eventually pixel value becomes 0 and image lose its details.

Code :



```
close all
clear
clc

% Read Image
img = imread("/home/nihar/Desktop/SEM 7/IP/Lab/Lab3/lab3images/2.jpg");
figure
subplot(1,3,1)
imshow(img)
title("Original Image");
out = my_standard_box_filter_padding(img,9,9);
for i=1:4
  out = my_standard_box_filter_padding(out,9,9);
endfor
subplot(1,3,2)
imshow(out)
title("After 5 pass");

for i=1:5
  out = my_standard_box_filter_padding(out,9,9);
endfor
subplot(1,3,3)
imshow(out)
title("After 10 pass");
```

Output :

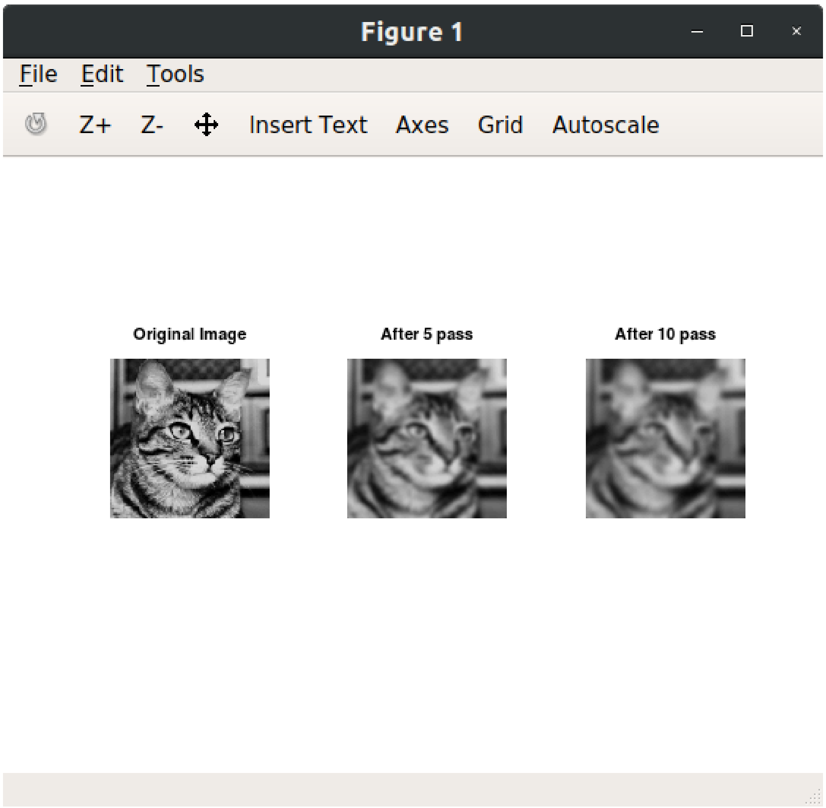


Figure 1
File Edit Tools
Z+ Z- Insert Text Axes Grid Autoscale
Original Image
After 5 pass
After 10 pass