Spatial filtering

Created by:

Asaf Kamer

Stav Sabag

Table of Contents


```
clearvars;
clear all ;
close all ;
clc ;
```

check

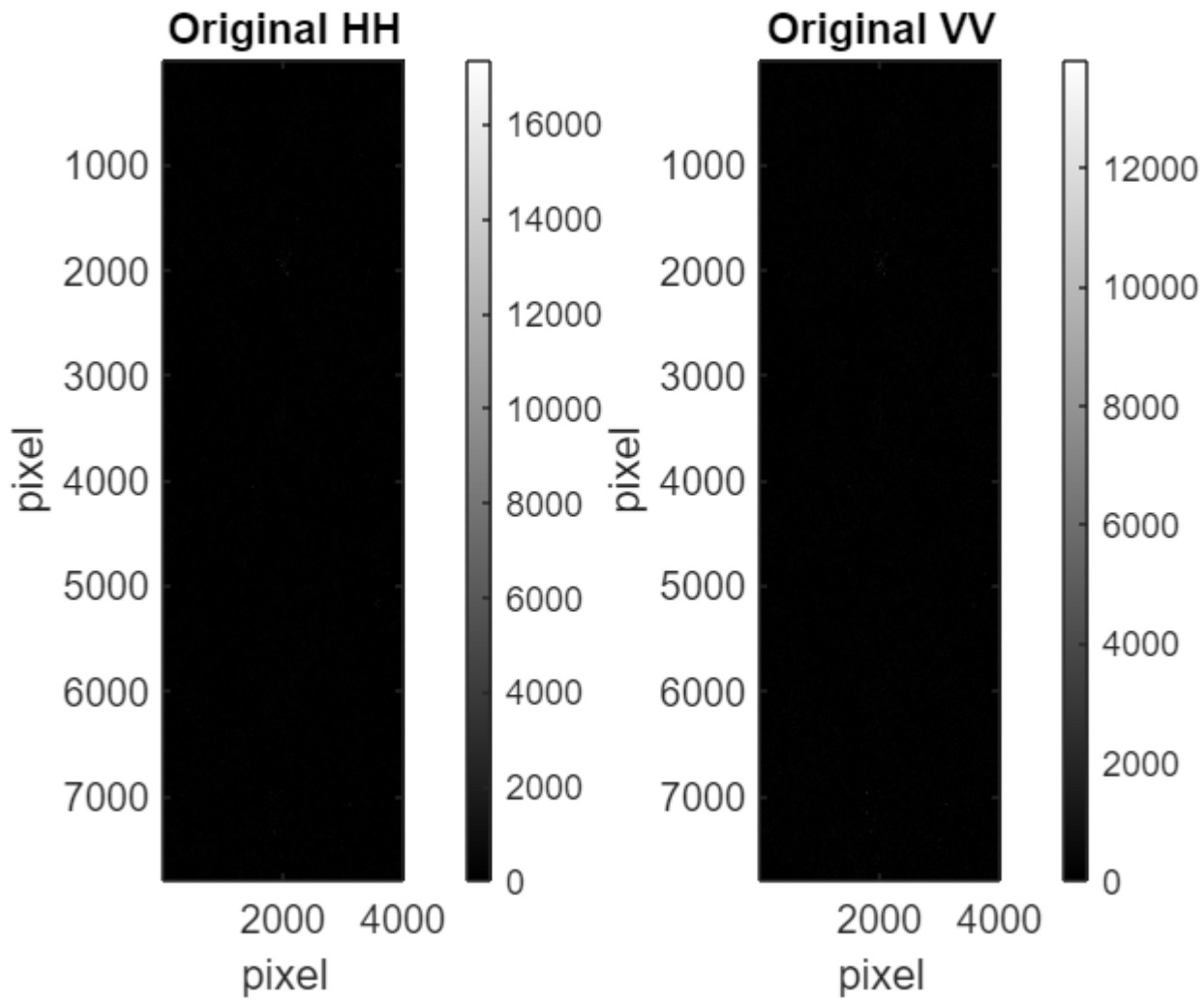
```
```

Loading satellite images

```
% Specify the filename of the COSAR file
filename = 'IMAGE_HH_SRA_spot_048.cos';

% Call the readCosFile function to read the data and information from the file
[HH_image, info] = readCosFile(filename,1);


% Specify the filename of the COSAR file
filename = 'IMAGE_VV_SRA_spot_048.cos';

% Call the readCosFile function to read the data and information from the file
[VV_image, info] = readCosFile(filename,1);

% Display some basic information about the data and the file
disp(['File ' filename ' contains ' num2str(size(VV_image,2)) ' range lines and ' ...
    num2str(size(VV_image,3)) ' azimuth lines.'])
```

```
File IMAGE_VV_SRA_spot_048.cos contains 7808 range lines and 4018 azimuth lines.
```

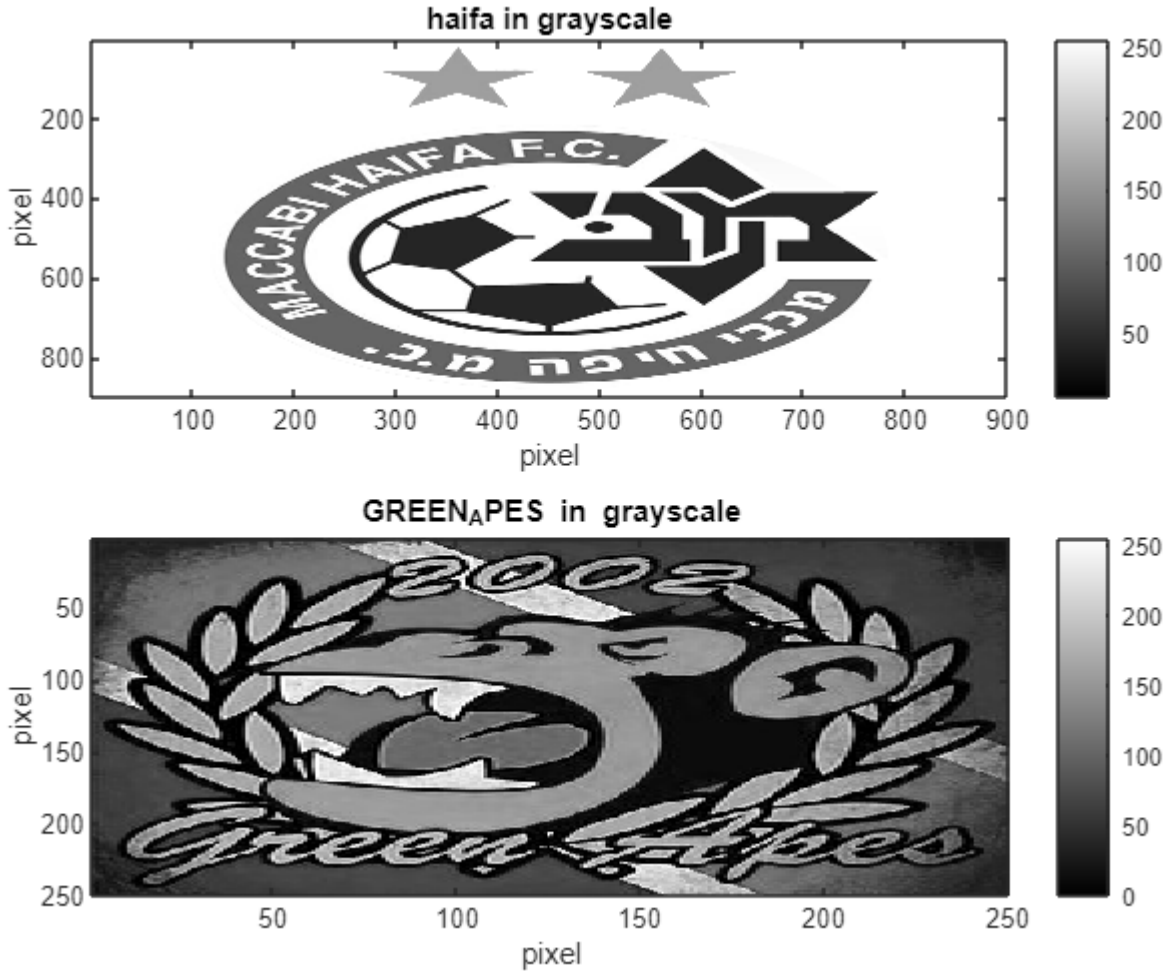
```
% Combine the real and imaginary parts of the first image and the second into a single complex
abs_HH_image = (double(HH_image(1,:,:)).^2 + double(HH_image(2,:,:)).^2).^0.5 ;
abs_HH_image = uint16(squeeze(abs_HH_image)) ;
% Compute the magnitude of each complex number in the matrix


abs_VV_image = (double(VV_image(1,:,:)).^2 + double(VV_image(2,:,:)).^2).^0.5 ;
abs_VV_image = uint16(squeeze(abs_VV_image)) ;

tle1 = 'Original HH' ;
tle2 = 'Original VV' ;
figure ;
subplot(1,2,1)
imagesc(abs_HH_image)
colormap("gray");
colorbar;
title(tle1,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
subplot(1,2,2)
imagesc(abs_VV_image)
colormap("gray");
colorbar;
title(tle2,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```



Loading Camera's images

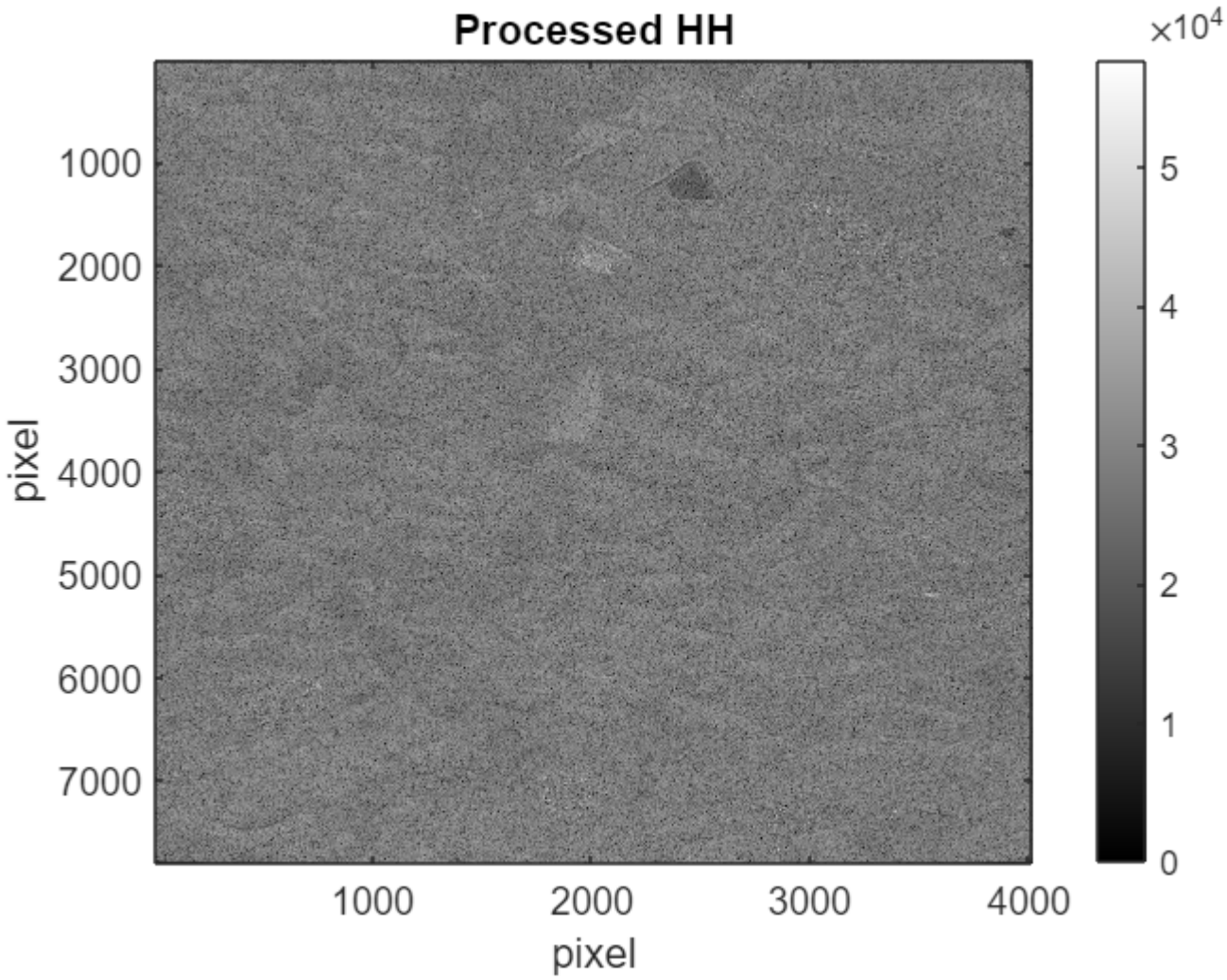
```
% Read the images
haifa = imread('haifa.png');
GREEN_APES = imread('GREEN_APES_2002_ICON.jpg');

% Convert the images to grayscale
haifa_gray = im2gray(haifa(:,:,2));
GREEN_APES_gray = im2gray(GREEN_APES(:,:,2));

% Display the grayscale images
tle1 = 'haifa in grayscale' ;
tle2 = 'GREEN_APES in grayscale' ;
figure ;
subplot(2,1,1)
imagesc(haifa_gray)
colormap("gray");
colorbar;
title(tle1,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
subplot(2,1,2)
```

```
imagesc(GREEN_APES_gray)
colormap("gray");
colorbar;
title(tle2,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```

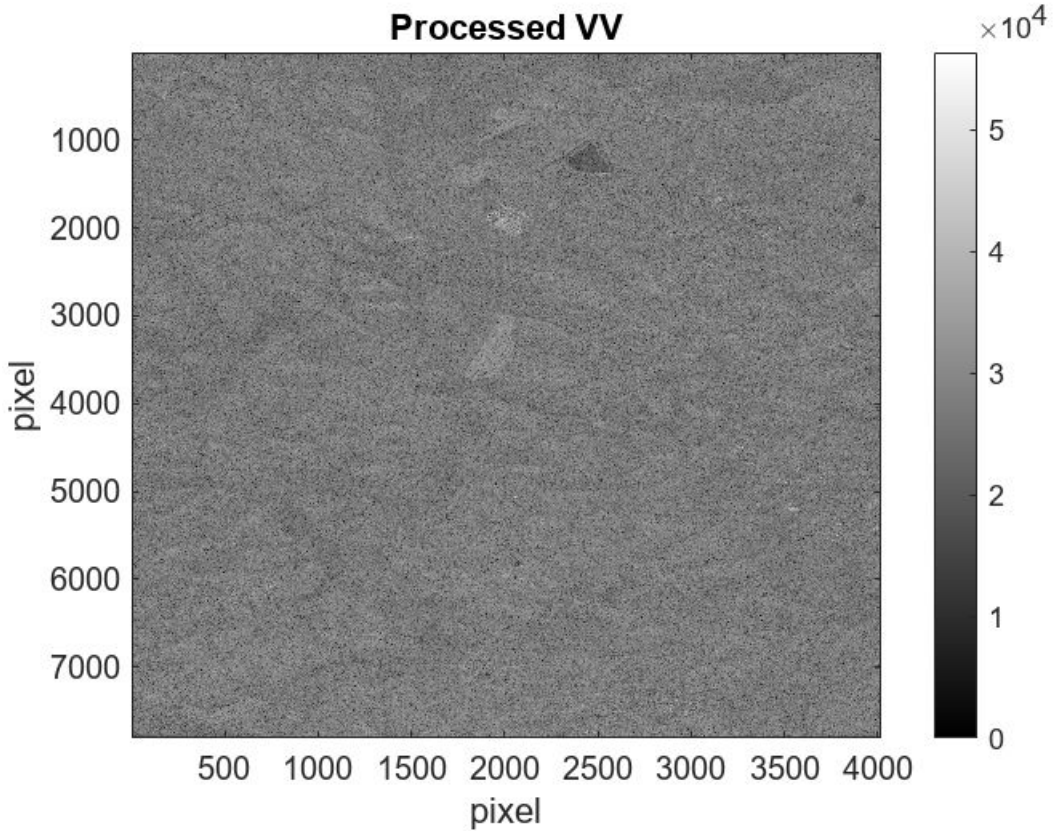


Pixel processing on HH and VV:

```
% Start the timer
tic ;

abs_HH_processed = Logarithmic(abs_HH_image) ;
abs_VV_processed = Logarithmic(abs_VV_image) ;
tle1 = 'Processed HH' ;
tle2 = 'Processed VV' ;
figure ;
imagesc(abs_HH_processed)
colormap("gray");
colorbar;
title(tle1,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```




```
figure ;
imagesc(abs_VV_processed)
colormap("gray");
colorbar;
title(tle2,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```



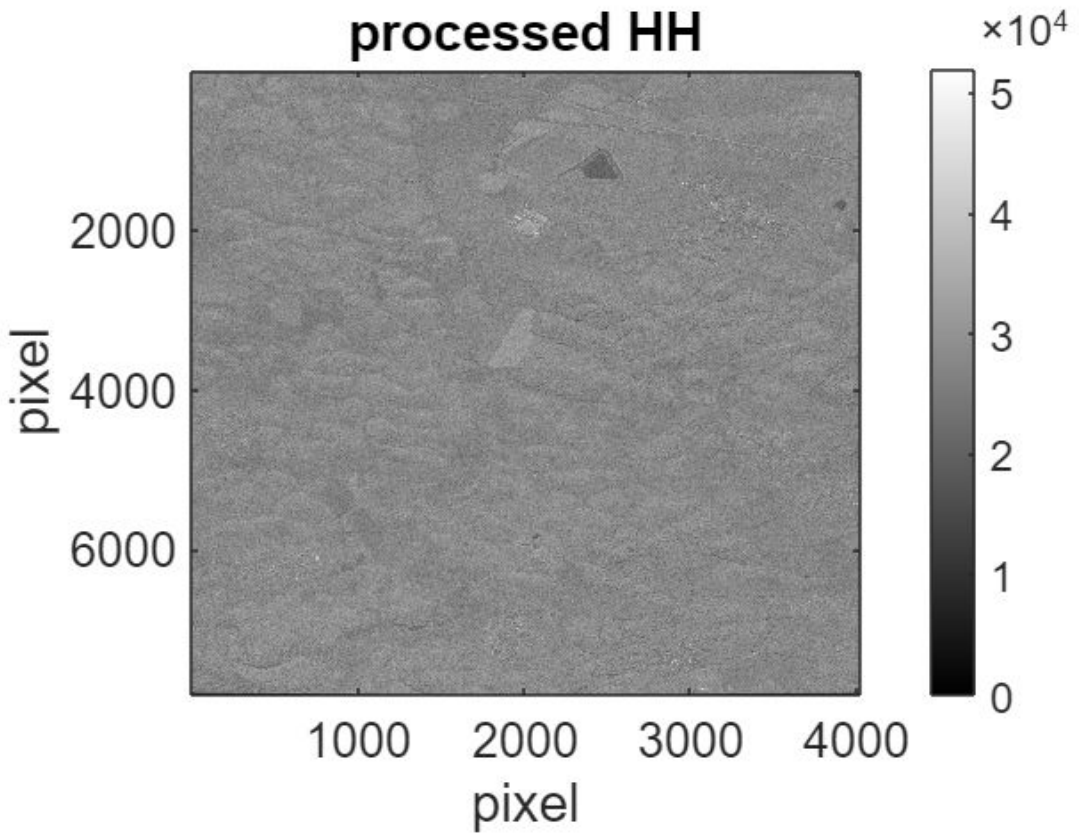

```
elapsed_time = toc;
mins = floor(elapsed_time / 60);
secs = round(mod(elapsed_time, 60));
fprintf('Total elapsed time: %02d:%02d\n', mins, secs);
```

```
Total elapsed time: 00:01
```

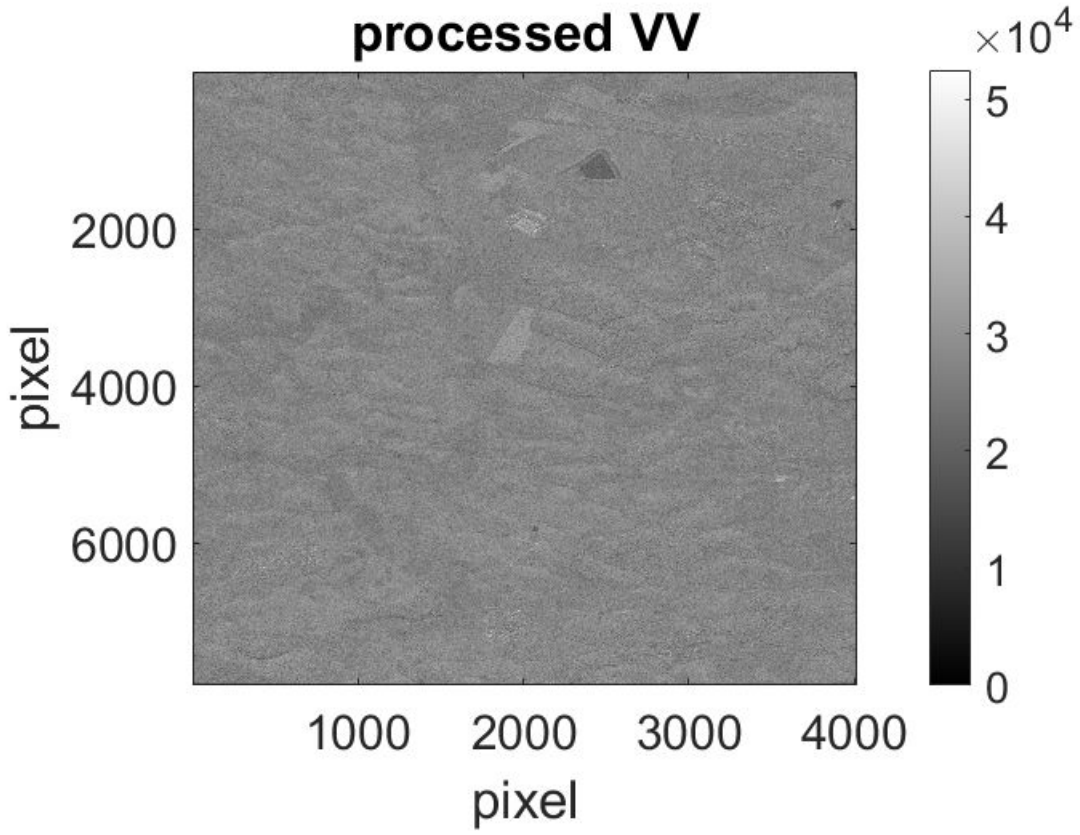
Median filter 3×3

```
% Start the timer
tic ;


fx = 3 ;
fy = 3 ;
processed_HH = median(abs_HH_processed,fx,fy) ;
processed_VV = median(abs_VV_processed,fx,fy) ;
processed_haifa = median(haifa_gray,fx,fy) ;
processed_GREEN_APES = median(GREEN_APES_gray,fx,fy) ;


tle1 = 'processed HH' ;
tle2 = 'processed VV' ;
tle3 = 'processed haifa' ;
tle4 = 'processed GREEN_APES' ;
figure ;
```

```
imagesc(processed_HH)
colormap("gray");
colorbar;
title(tle1,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```

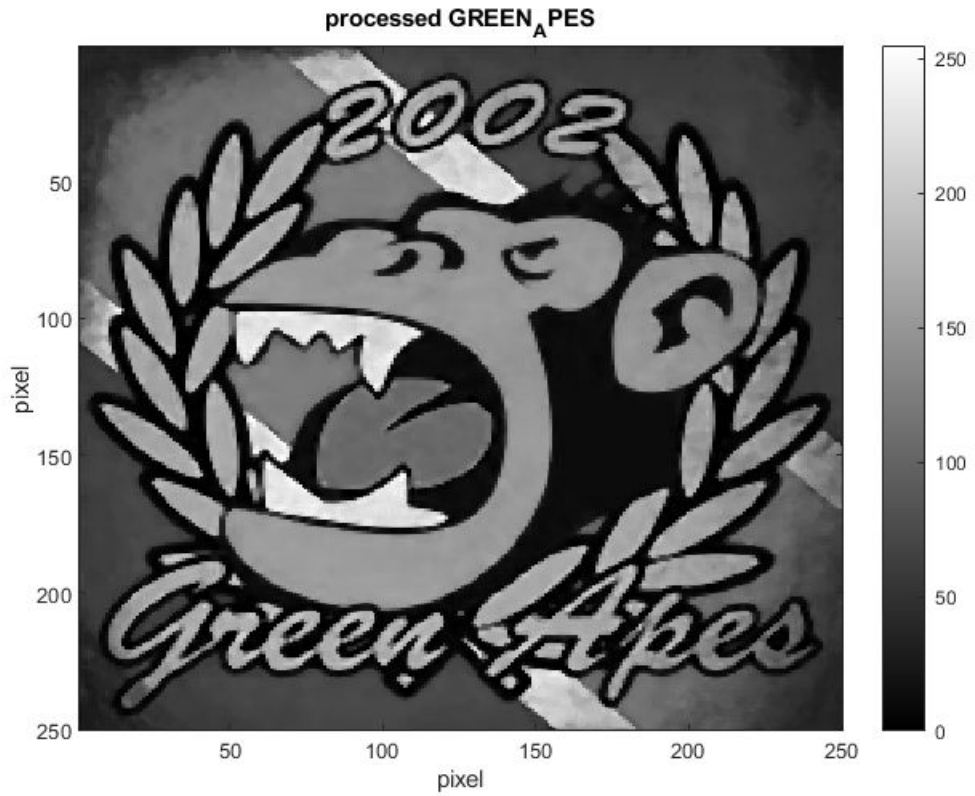



```
figure ;
imagesc(processed_VV)
colormap("gray");
colorbar;
title(tle2,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```




```
figure ;
imagesc(processed_haifa)
colormap("gray");
colorbar;
title(tle3,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```


```
figure ;
imagesc(processed_GREEN_APES)
colormap("gray");
colorbar;
title(tle4,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```




```
elapsed_time = toc;
mins = floor(elapsed_time / 60);
secs = round(mod(elapsed_time, 60));
fprintf('Total elapsed time: %02d:%02d\n', mins, secs);
```

```
Total elapsed time: 04:59
```

Std & Mean

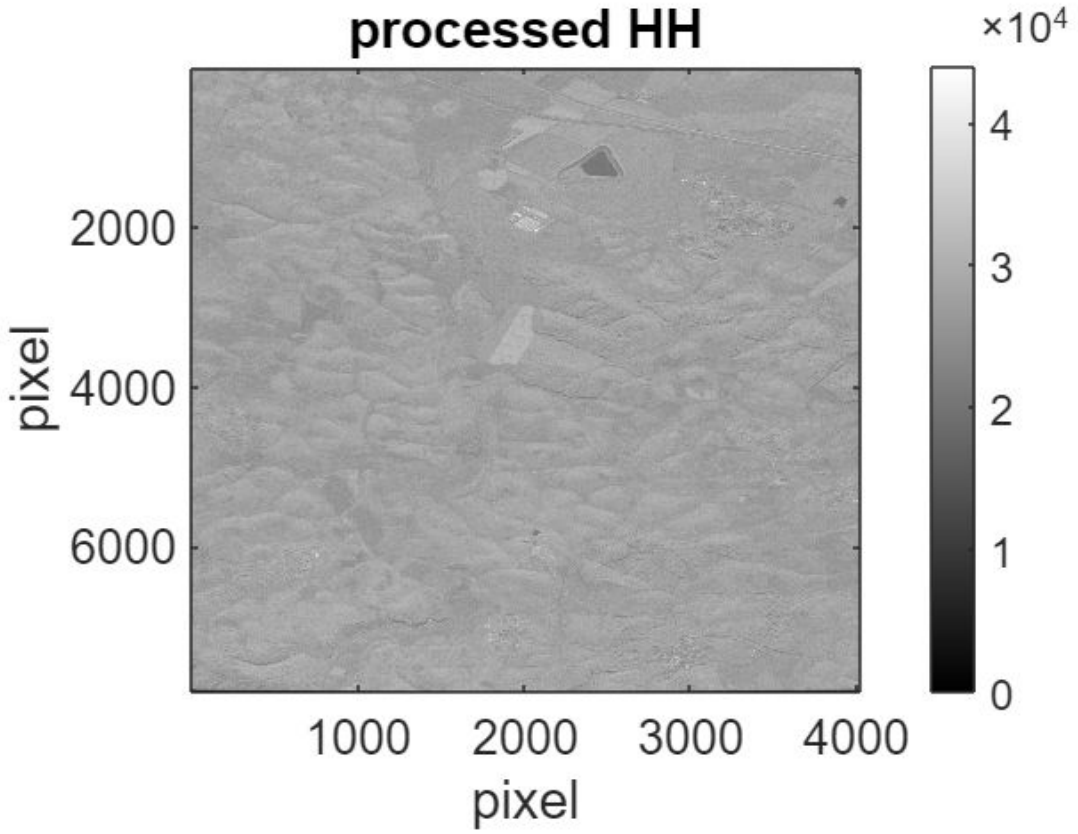
```
image_names = {'HH';'VV';'haifa';'GREEN_APES'};
print_image_stats(image_names, abs_HH_processed, abs_VV_processed, haifa_gray, GREEN_APES_gray,
```

Image name	Mean original	Std original	Mean processed	Std processed
{'HH' }	27306	4119.2	27806	2385.8
{'VV' }	26942	4098.9	27442	2361.9
{'haifa' }	215.24	74.046	215.37	73.842
{'GREEN_APES'}	89.968	67.881	87.89	64.26

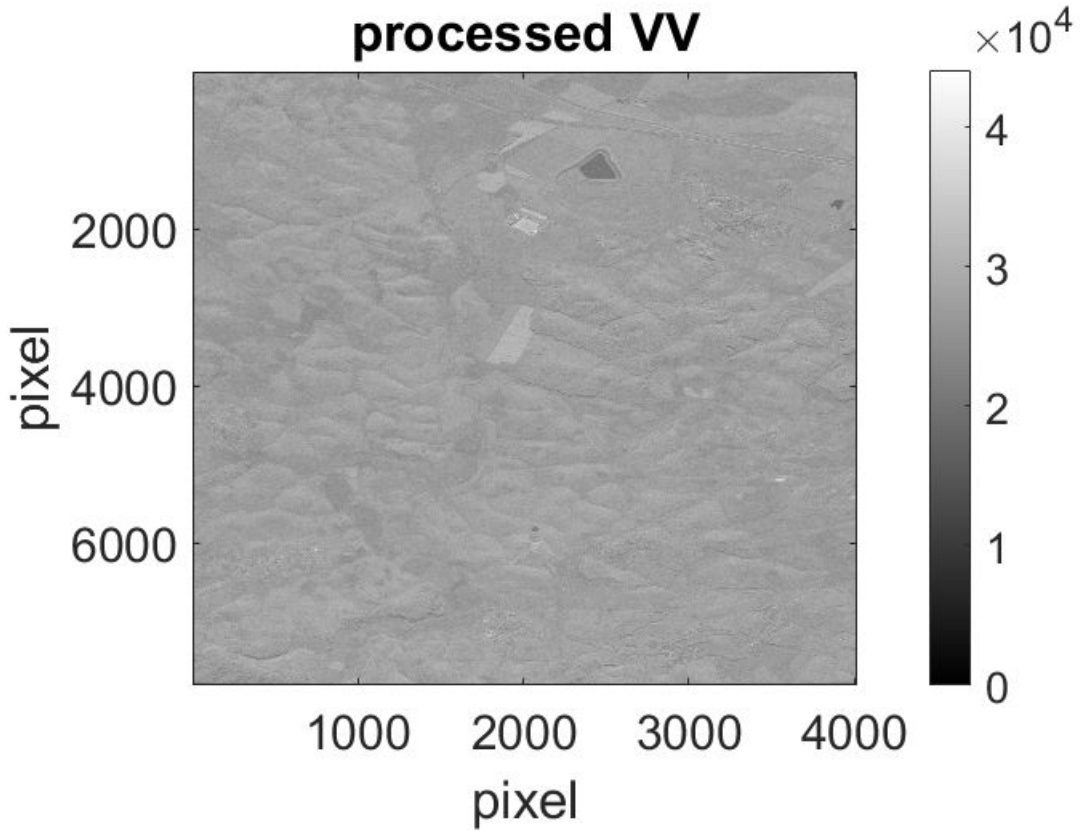
Median filter 9×9

```
% Start the timer
tic ;

fx = 9 ;
fy = 9 ;
```

```
processed_HH = median(abs_HH_processed,fx,fy) ;
processed_VV = median(abs_VV_processed,fx,fy) ;
processed_haifa = median(haifa_gray,fx,fy) ;
processed_GREEN_APES = median(GREEN_APES_gray,fx,fy) ;


figure ;
imagesc(processed_HH)
colormap("gray");
colorbar;
title(tle1,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```

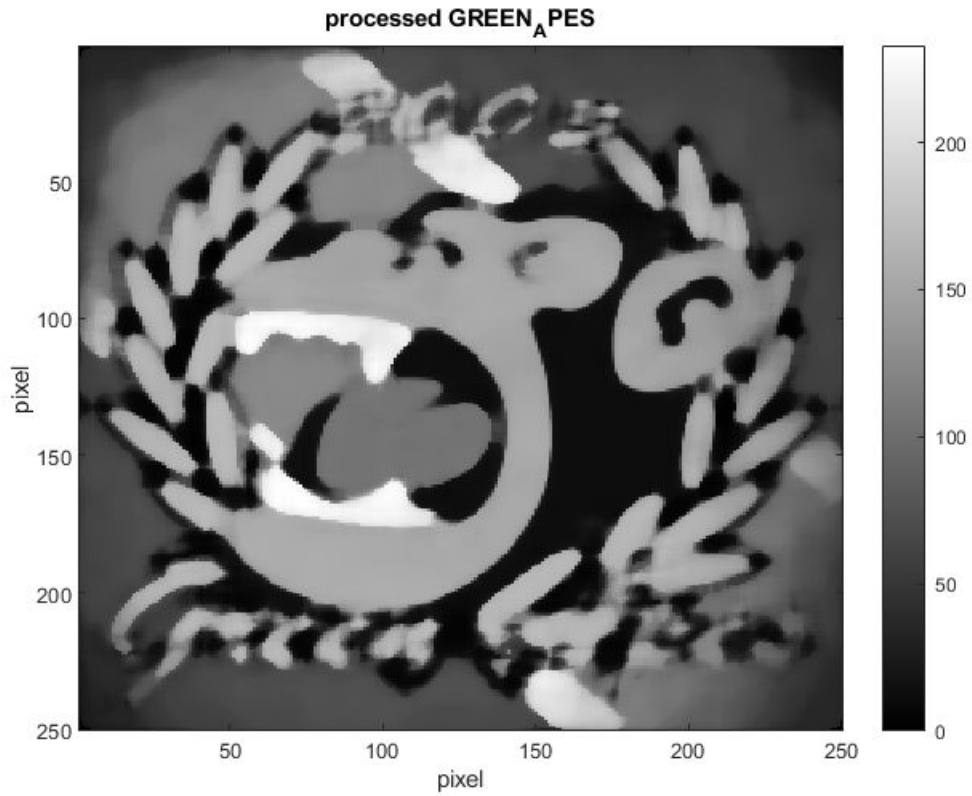



```
figure ;
imagesc(processed_VV)
colormap("gray");
colorbar;
title(tle2,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```




```
figure ;
imagesc(processed_haifa)
colormap("gray");
colorbar;
title(tle3,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```


```
figure ;
imagesc(processed_GREEN_APES)
colormap("gray");
colorbar;
title(tle4,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```




```
elapsed_time = toc;
mins = floor(elapsed_time / 60);
secs = round(mod(elapsed_time, 60));
fprintf('Total elapsed time: %02d:%02d\n', mins, secs);
```

```
Total elapsed time: 06:50
```

Std & Mean

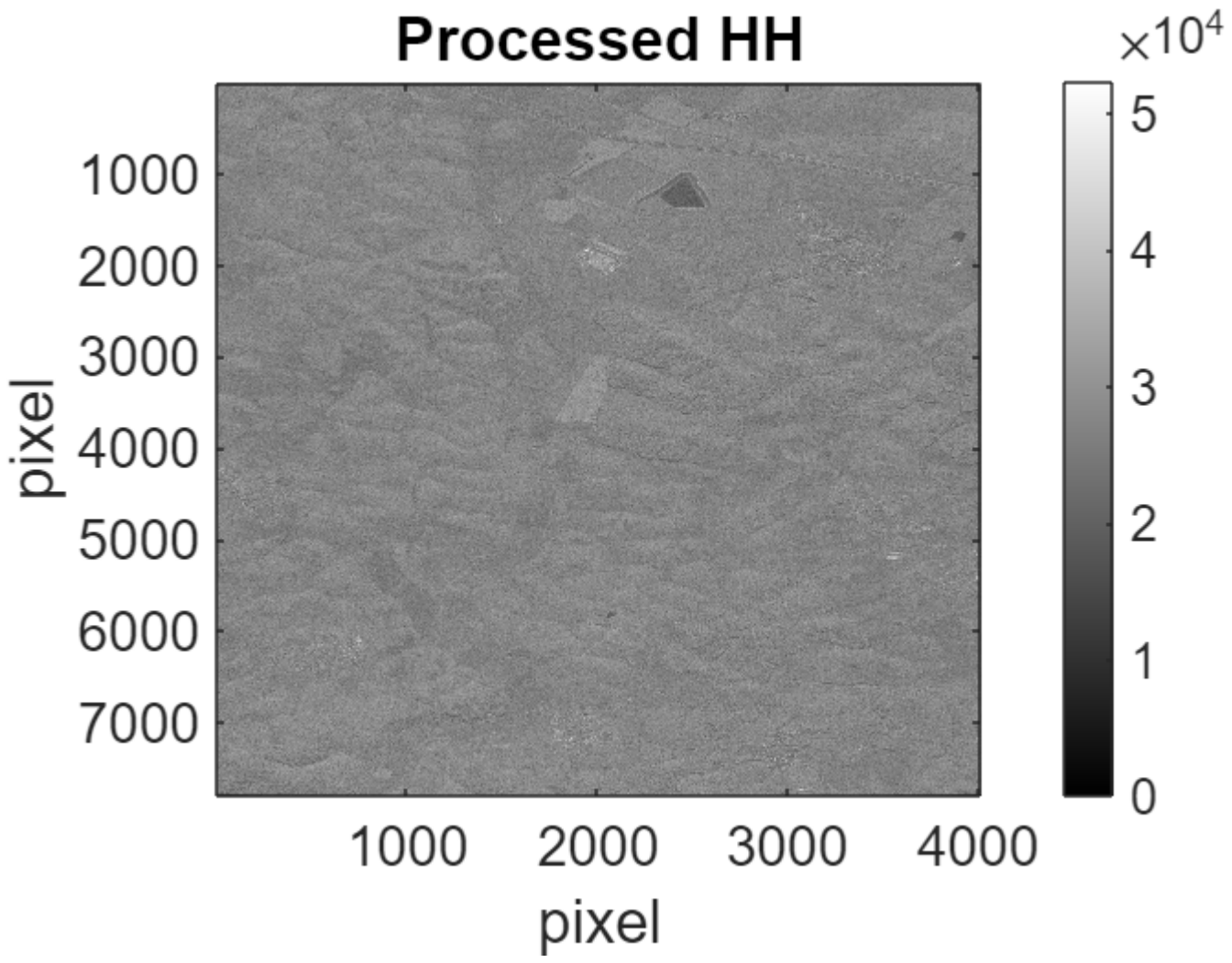
```
image_names = {'HH';'VV';'haifa';'GREEN_APES'};
print_image_stats(image_names, abs_HH_processed, abs_VV_processed, haifa_gray, GREEN_APES_gray,
```

Image name	Mean original	Std original	Mean processed	Std processed
{'HH' }	27306	4119.2	27835	1507.1
{'VV' }	26942	4098.9	27473	1482
{'haifa' }	215.24	74.046	215.56	73.354
{'GREEN_APES'}	89.968	67.881	90.852	56.481

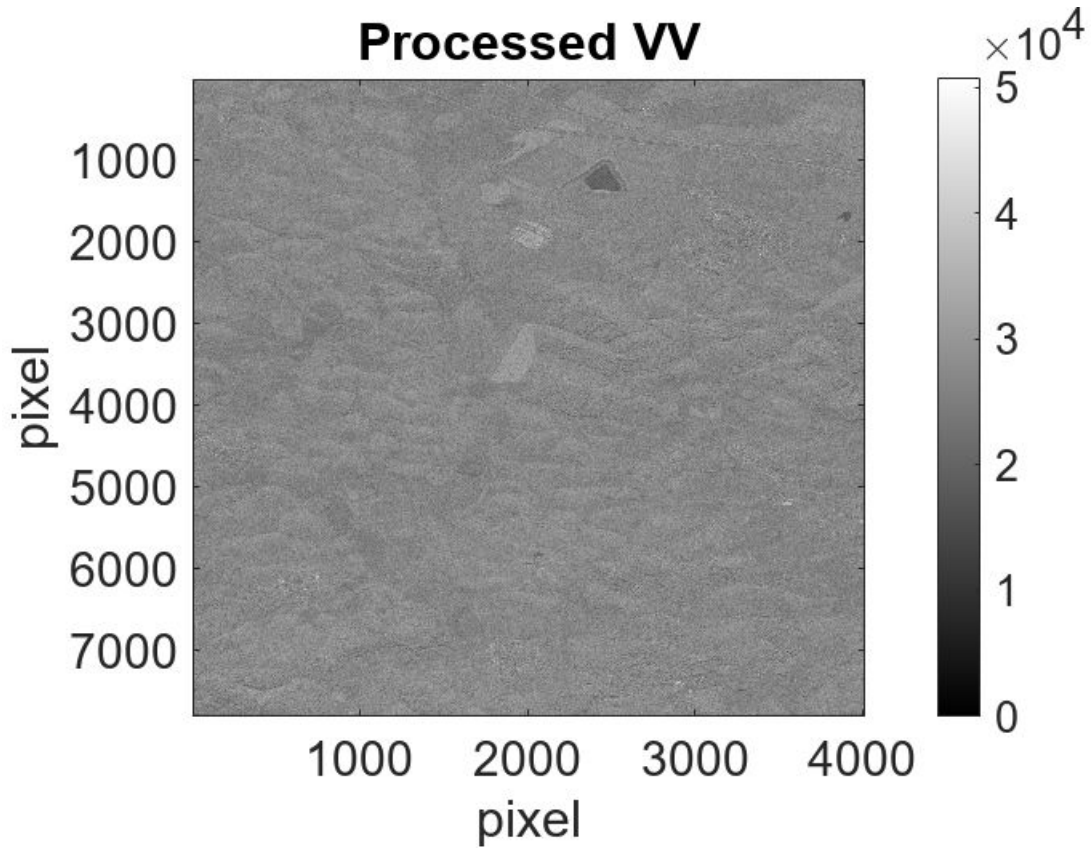
gaussian filter 3×3

```
% Start the timer
tic ;

fx = 3 ;
```

```
fy = 3;
var_x = sqrt(5) ;
var_y = sqrt(5) ;
processed_HH = gaussian_filter(abs_HH_processed, fx, fy, var_x, var_y) ;
processed_VV = gaussian_filter(abs_VV_processed, fx, fy, var_x, var_y) ;
processed_haifa = gaussian_filter(haifa_gray, fx, fy, var_x, var_y) ;
processed_GREEN_APES = gaussian_filter(GREEN_APES_gray, fx, fy, var_x, var_y) ;


figure ;
imagesc(processed_HH)
colormap("gray");
colorbar;
title(tle1,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```




```
figure ;
imagesc(processed_VV)
colormap("gray");
colorbar;
title(tle2,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```




```
figure ;
imagesc(processed_haifa)
colormap("gray");
colorbar;
```


```
title(tle3,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```

```
Unrecognized function or variable
'tle3'.
```

```
figure ;
imagesc(processed_GREEN_APES)
colormap("gray");
colorbar;
title(tle4,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
elapsed_time = toc;
mins = floor(elapsed_time / 60);
secs = round(mod(elapsed_time, 60));
fprintf('Total elapsed time: %02d:%02d\n', mins, secs);
```

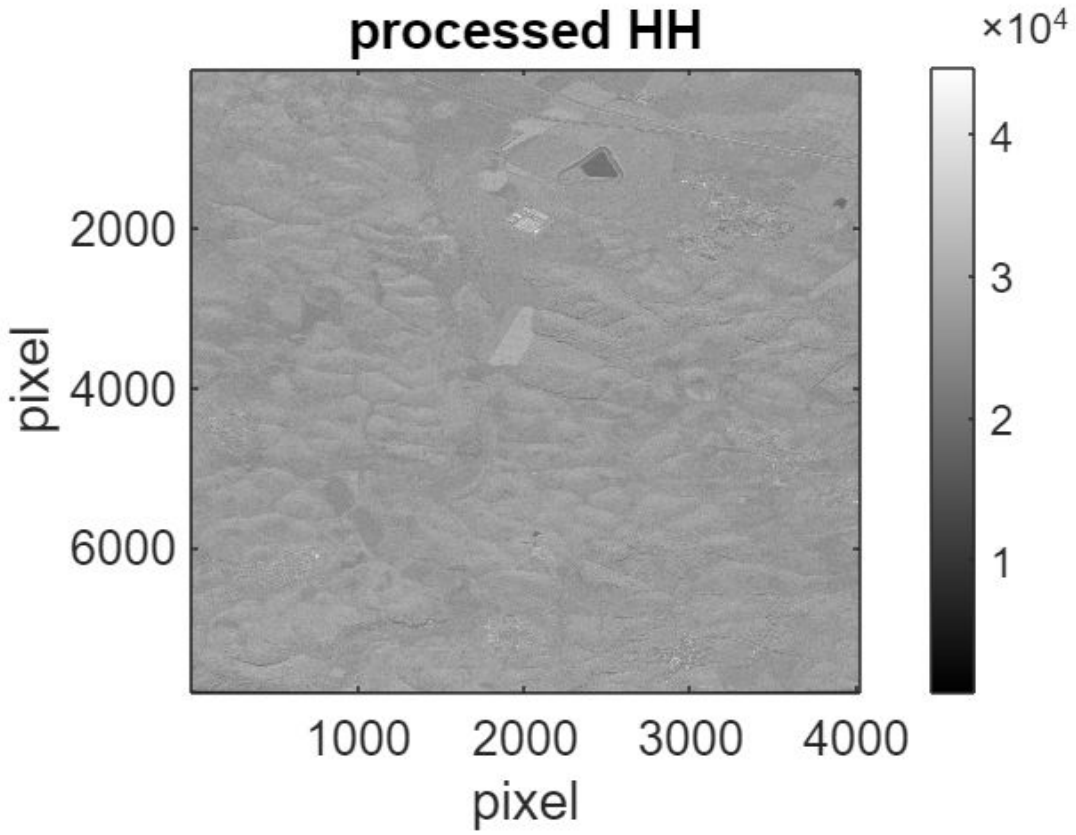
Std & Mean

```
image_names = {'HH';'VV';'haifa';'GREEN_APES'};
print_image_stats(image_names, abs_HH_processed, abs_VV_processed, haifa_gray, GREEN_APES_gray,
```

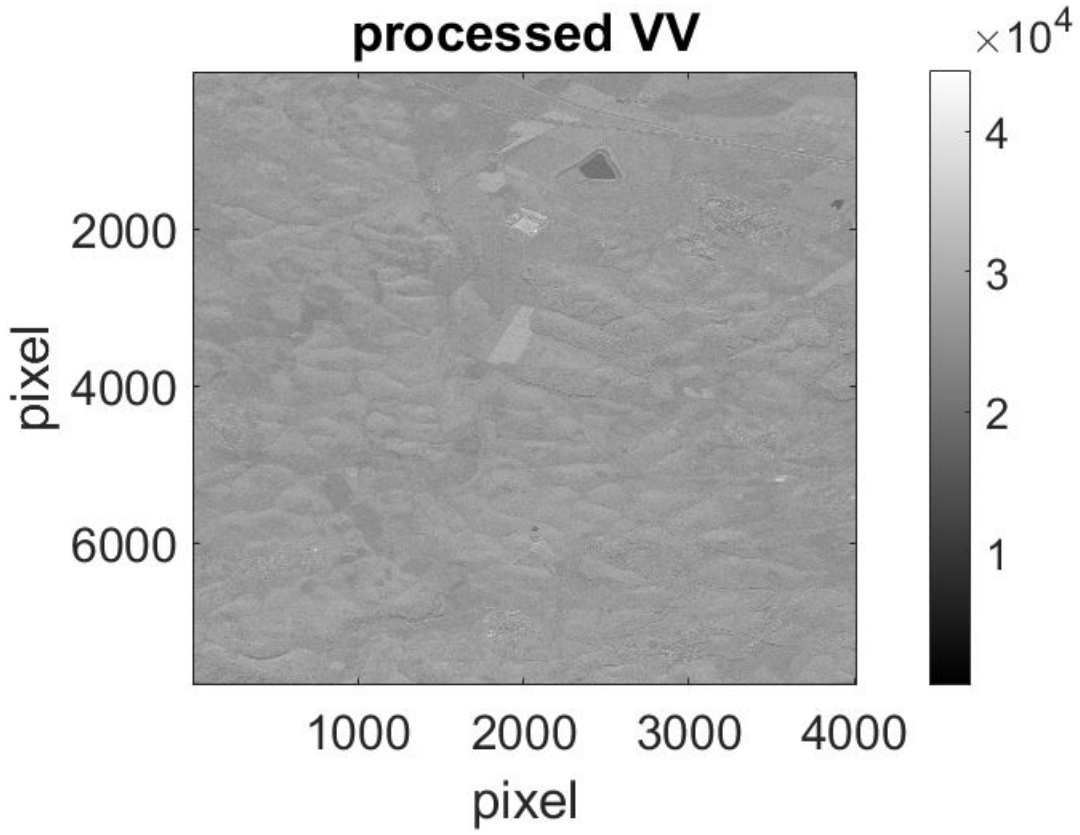
Image name	Mean original	Std original	Mean processed	Std processed
{'HH' }	27306	4119.2	27300	2315.5
{'VV' }	26942	4098.9	26936	2289.9
{'haifa' }	215.24	74.046	214.88	73.195
{'GREEN_APES'}	89.968	67.881	89.64	58.19

gaussian filter 9×9

```
% Start the timer
tic ;

fx = 9 ;
fy = 9;
var_x = sqrt(8) ;
var_y = sqrt(8) ;
processed_HH = gaussian_filter(abs_HH_processed, fx, fy, var_x, var_y) ;
processed_VV = gaussian_filter(abs_VV_processed, fx, fy, var_x, var_y) ;
processed_haifa = gaussian_filter(haifa_gray, fx, fy, var_x, var_y) ;
processed_GREEN_APES = gaussian_filter(GREEN_APES_gray, fx, fy, var_x, var_y) ;


figure ;
imagesc(processed_HH)
colormap("gray");
colorbar;
title(tle1,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```

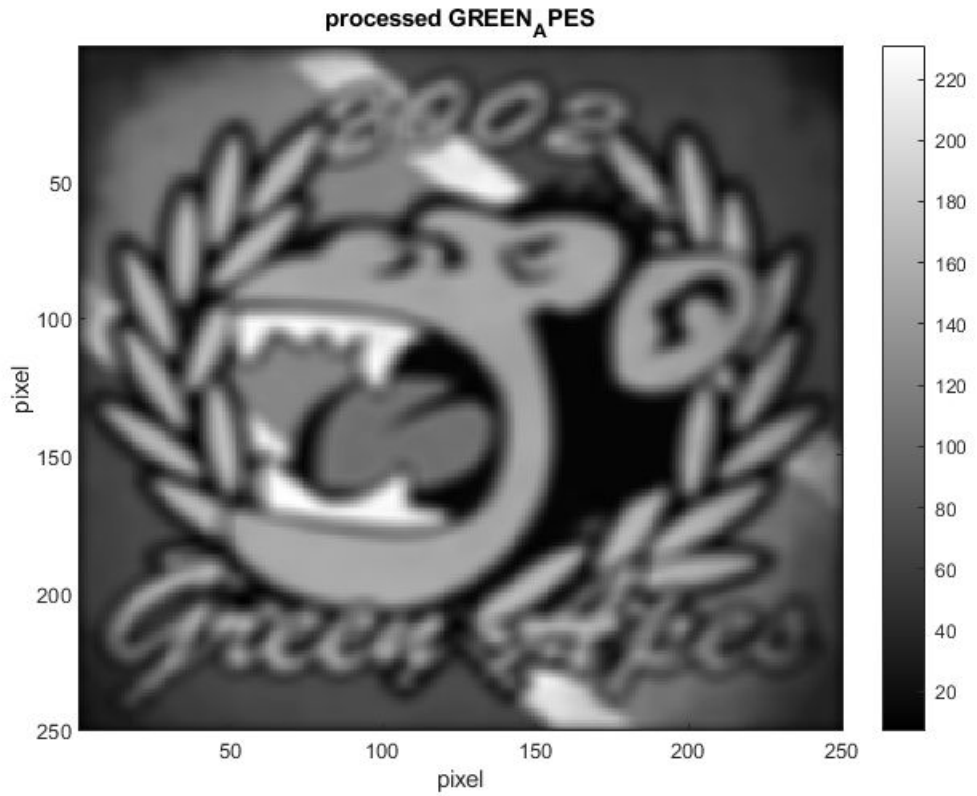



```
figure ;
imagesc(processed_VV)
colormap("gray");
colorbar;
title(tle2,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```




```
figure ;
imagesc(processed_haifa)
colormap("gray");
colorbar;
title(tle3,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```


```
figure ;
imagesc(processed_GREEN_APES)
colormap("gray");
colorbar;
title(tle4,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```




```
elapsed_time = toc;
mins = floor(elapsed_time / 60);
secs = round(mod(elapsed_time, 60));
fprintf('Total elapsed time: %02d:%02d\n', mins, secs);
```

```
Total elapsed time: 00:03
```

Std & Mean

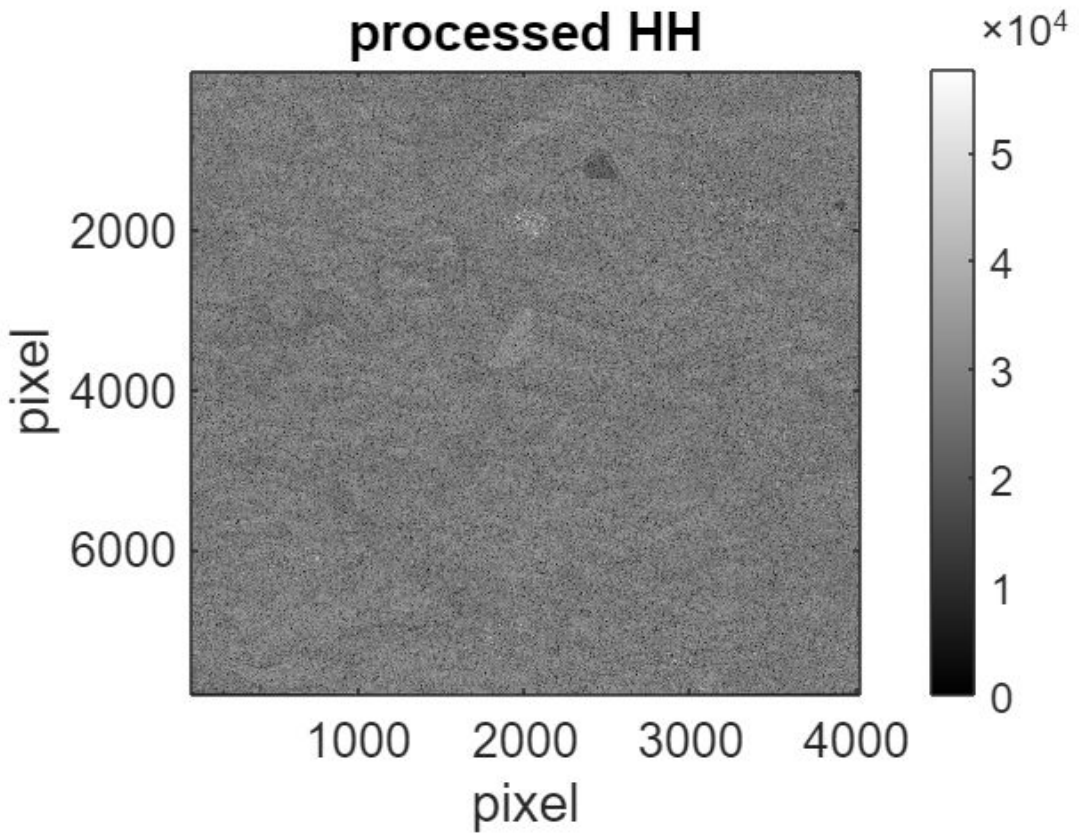
```
image_names = {'HH';'VV';'haifa';'GREEN_APES'};
print_image_stats(image_names, abs_HH_processed, abs_VV_processed, haifa_gray, GREEN_APES_gray,
```

Image name	Mean original	Std original	Mean processed	Std processed
{'HH' }	27306	4119.2	27289	1573.9
{'VV' }	26942	4098.9	26925	1545.3
{'haifa' }	215.24	74.046	214.23	70.442
{'GREEN_APES'}	89.968	67.881	89.087	44.829

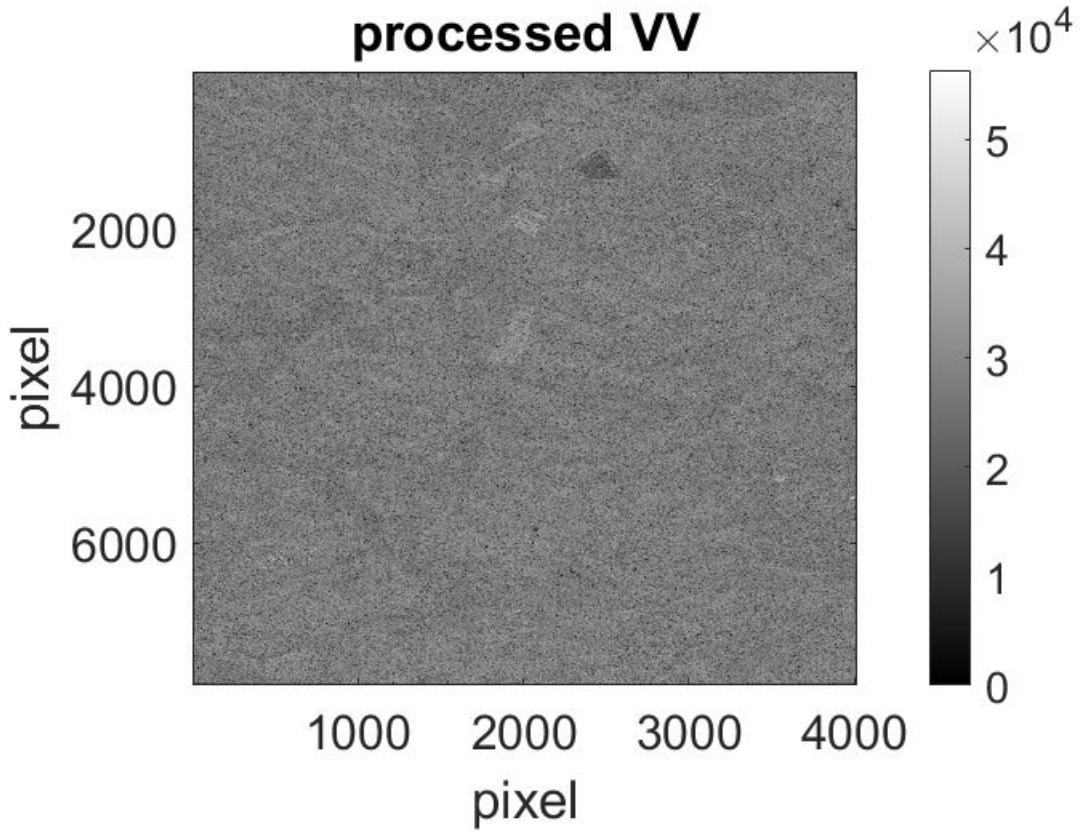
bilateral filter 3×3

```
% Start the timer
tic ;

fx = 3 ;
```

```
fy = 3 ;
spatial_var = sqrt(10) ;
radiometric_var = sqrt(12) ;


processed_HH = bilateral_filter(abs_HH_processed, fx, fy, spatial_var, radiometric_var) ;
processed_VV = bilateral_filter(abs_VV_processed, fx, fy, spatial_var, radiometric_var) ;
processed_haifa = bilateral_filter(haifa_gray, fx, fy, spatial_var, radiometric_var) ;
processed_GREEN_APES = bilateral_filter(GREEN_APES_gray, fx, fy, spatial_var, radiometric_var)

figure ;
imagesc(processed_HH)
colormap("gray");
colorbar;
title(tle1,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```

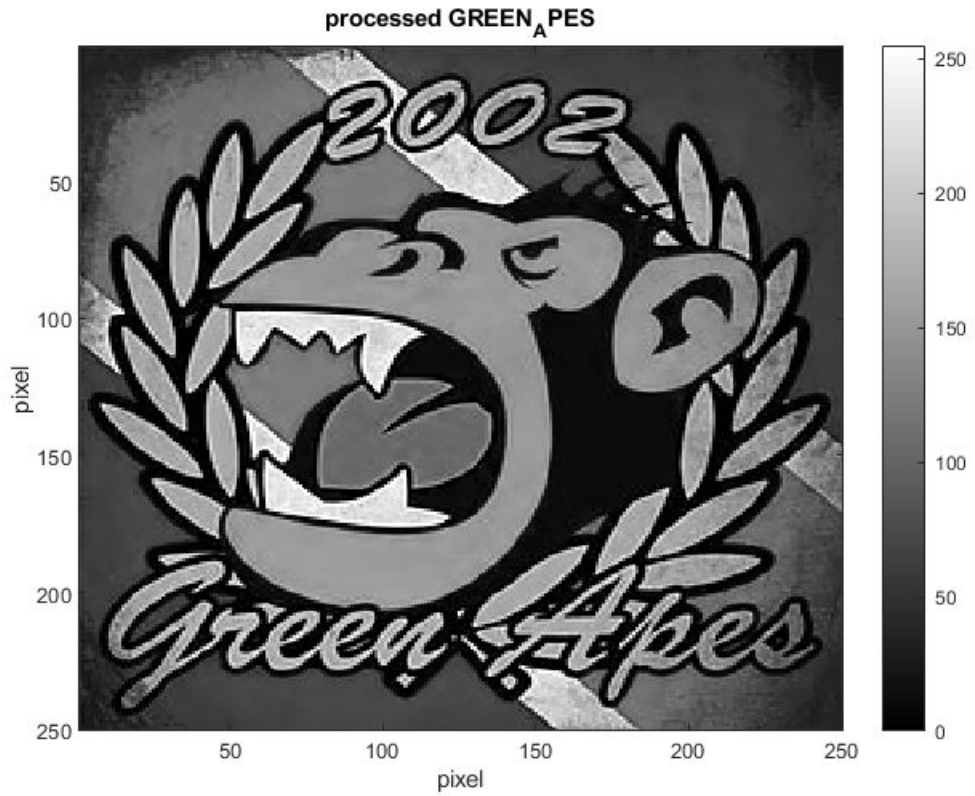



```
figure ;
imagesc(processed_VV)
colormap("gray");
colorbar;
title(tle2,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```




```
figure ;
imagesc(processed_haifa)
colormap("gray");
colorbar;
title(tle3,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```


```
figure ;
imagesc(processed_GREEN_APES)
colormap("gray");
colorbar;
title(tle4,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```




```
elapsed_time = toc;
mins = floor(elapsed_time / 60);
secs = round(mod(elapsed_time, 60));
fprintf('Total elapsed time: %02d:%02d\n', mins, secs);
```

```
Total elapsed time: 02:48
```

Std & Mean

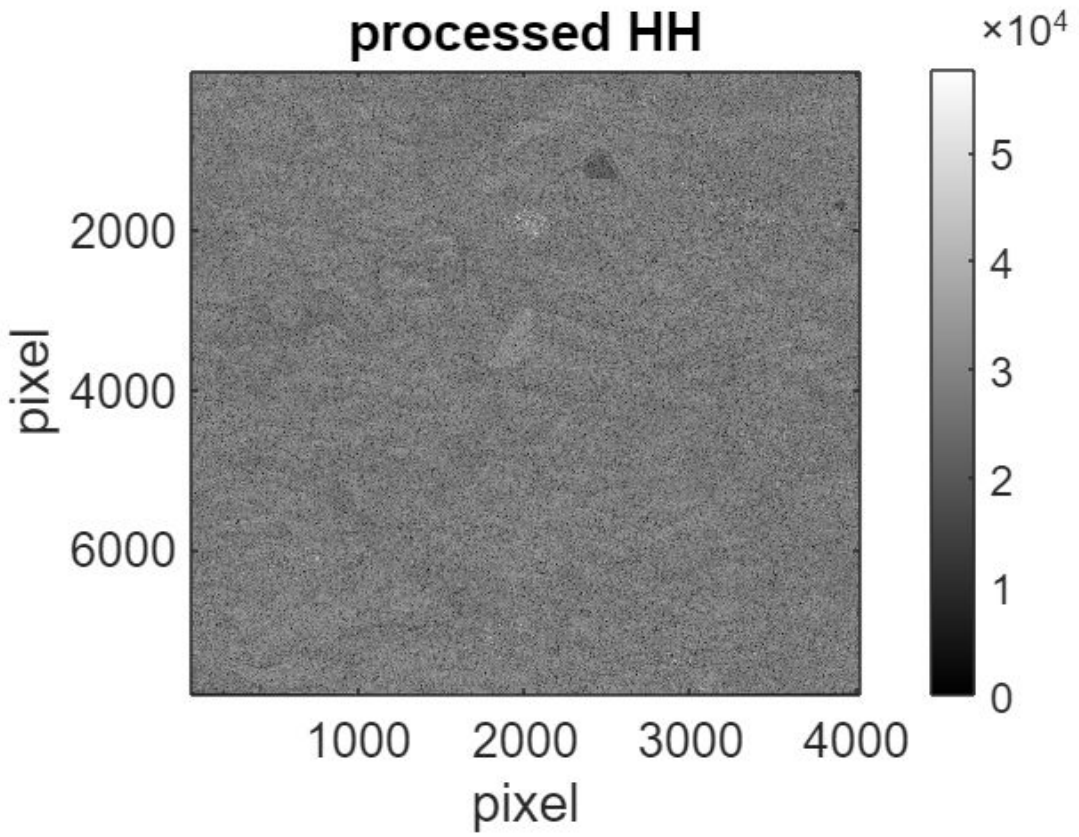
```
image_names = {'HH';'VV';'haifa';'GREEN_APES'};
print_image_stats(image_names, abs_HH_processed, abs_VV_processed, haifa_gray, GREEN_APES_gray,
```

Image name	Mean original	Std original	Mean processed	Std processed
{'HH' }	27306	4119.2	27306	4119.2
{'VV' }	26942	4098.9	26942	4098.9
{'haifa' }	215.24	74.046	215.26	74.046
{'GREEN_APES'}	89.968	67.881	89.95	67.871

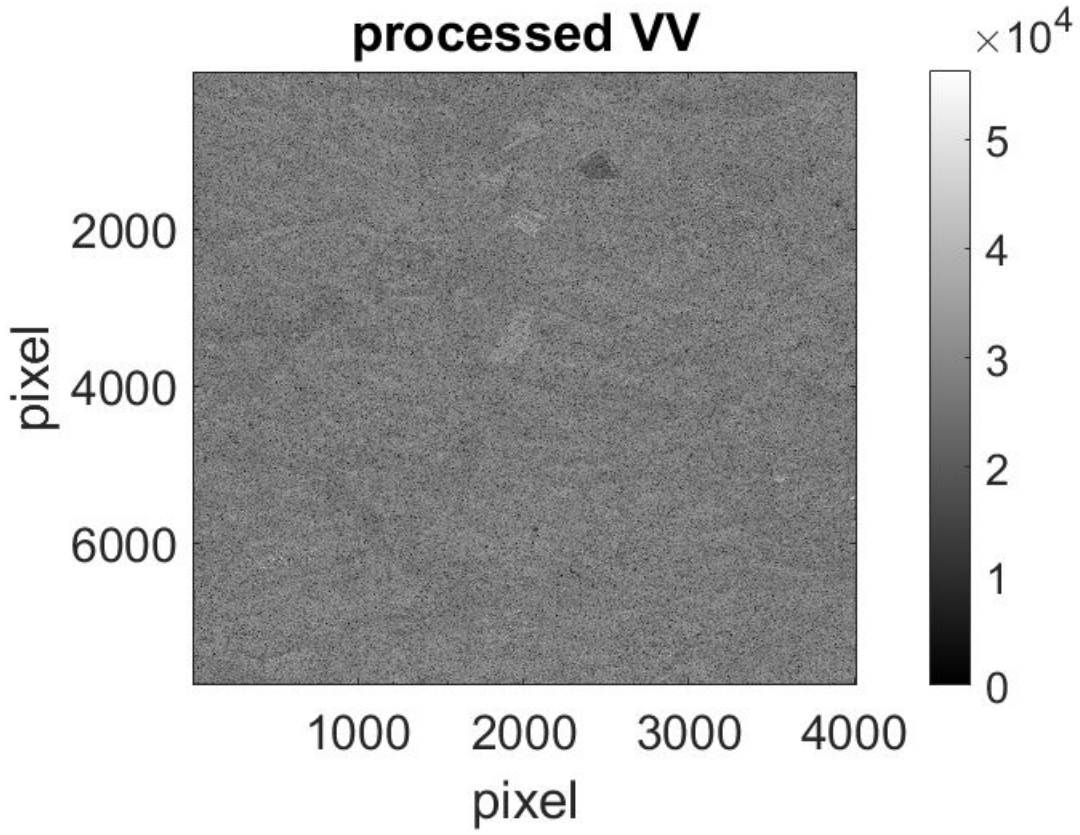
bilateral filter 9×9

```
% Start the timer
tic ;

fx = 9 ;
```

```
fy = 9 ;
spatial_var = sqrt(10) ;
radiometric_var = sqrt(12) ;


processed_HH = bilateral_filter(abs_HH_processed, fx, fy, spatial_var, radiometric_var) ;
processed_VV = bilateral_filter(abs_VV_processed, fx, fy, spatial_var, radiometric_var) ;
processed_haifa = bilateral_filter(haifa_gray, fx, fy, spatial_var, radiometric_var) ;
processed_GREEN_APES = bilateral_filter(GREEN_APES_gray, fx, fy, spatial_var, radiometric_var)

figure ;
imagesc(processed_HH)
colormap("gray");
colorbar;
title(tle1,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```

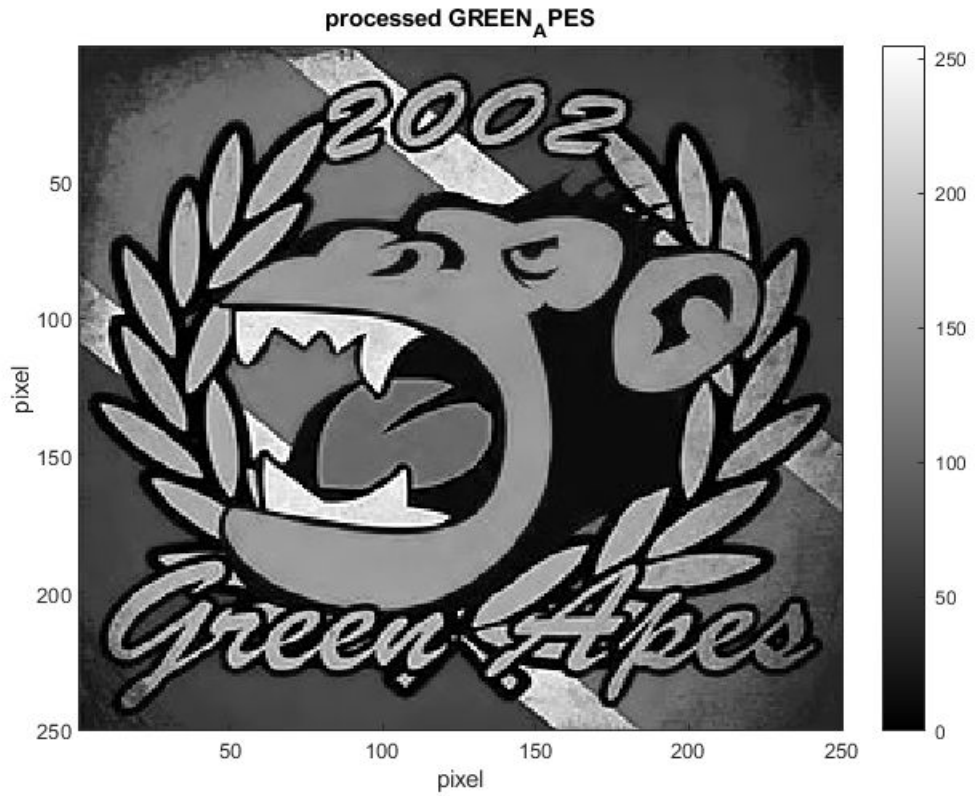



```
figure ;
imagesc(processed_VV)
colormap("gray");
colorbar;
title(tle2,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```




```
figure ;
imagesc(processed_haifa)
colormap("gray");
colorbar;
title(tle3,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```


```
figure ;
imagesc(processed_GREEN_APES)
colormap("gray");
colorbar;
title(tle4,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```




```
elapsed_time = toc;
mins = floor(elapsed_time / 60);
secs = round(mod(elapsed_time, 60));
fprintf('Total elapsed time: %02d:%02d\n', mins, secs);
```

```
Total elapsed time: 03:31
```

Std & Mean

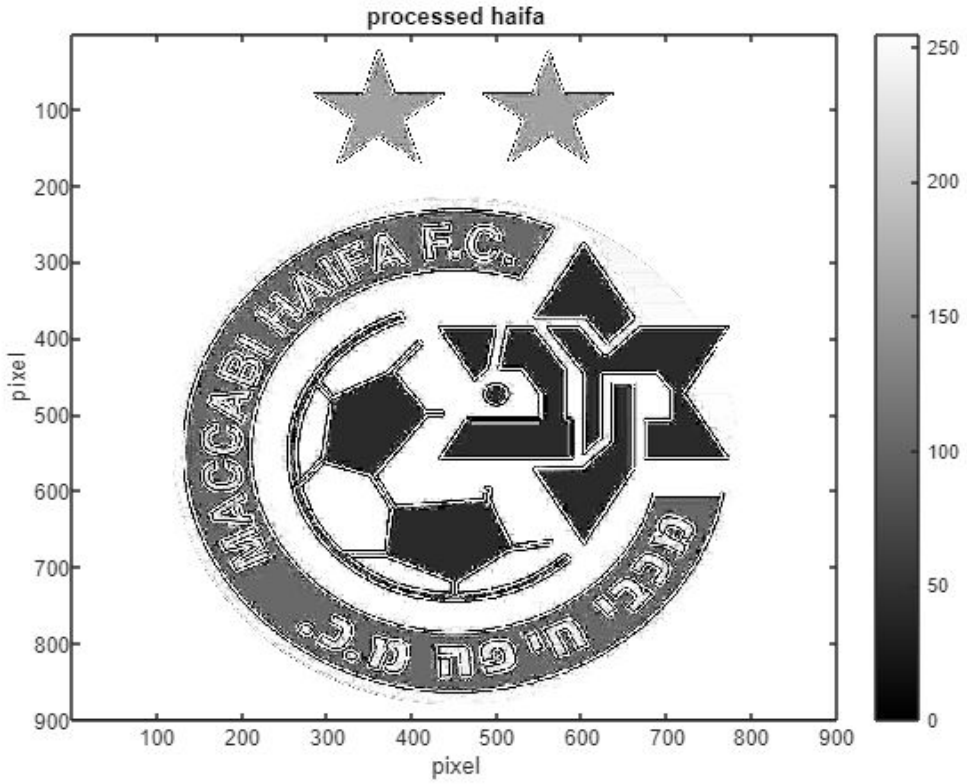
```
image_names = {'HH';'VV';'haifa';'GREEN_APES'};
print_image_stats(image_names, abs_HH_processed, abs_VV_processed, haifa_gray, GREEN_APES_gray,
```

Image name	Mean original	Std original	Mean processed	Std processed
{'HH' }	27306	4119.2	27306	4119.2
{'VV' }	26942	4098.9	26942	4098.9
{'haifa' }	215.24	74.046	215.28	74.036
{'GREEN_APES'}	89.968	67.881	89.937	67.862

Edge highlight

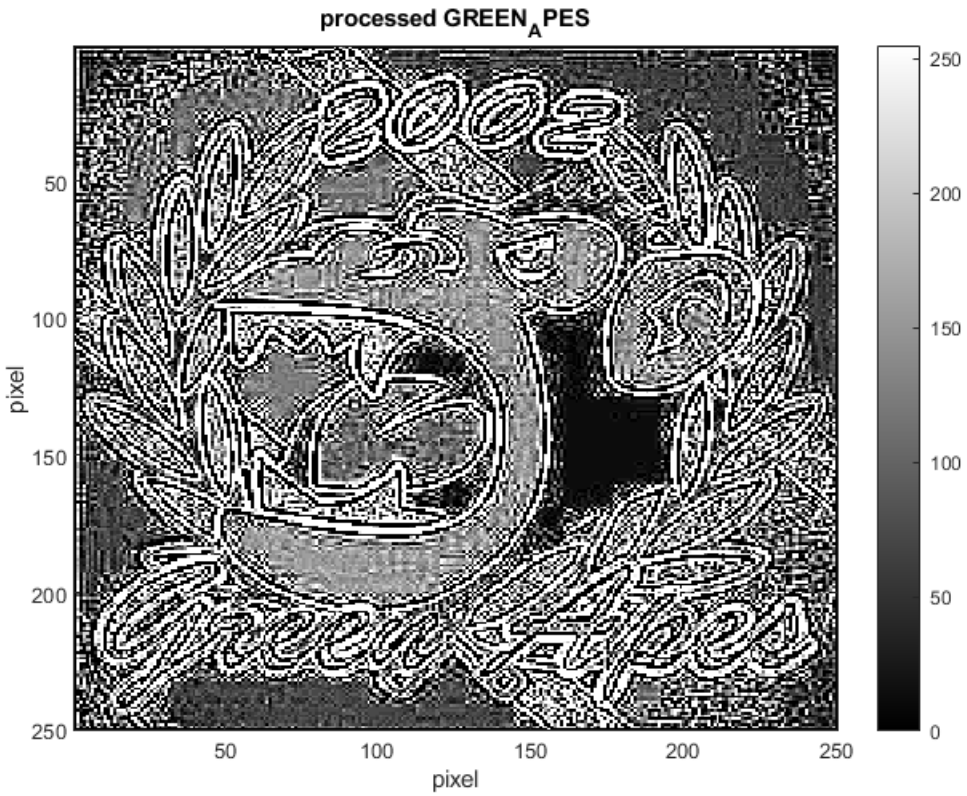
```
% Start the timer
tic ;

c = 10 ;
```

```
[processed_haifa,edge_haifa] = edge_highlight(haifa_gray, c) ;
[processed_GREEN_APES,edge_GREEN_APES] = edge_highlight(GREEN_APES_gray, c) ;


figure ;
imagesc(processed_haifa)
colormap("gray");
colorbar;
title(tle3,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```

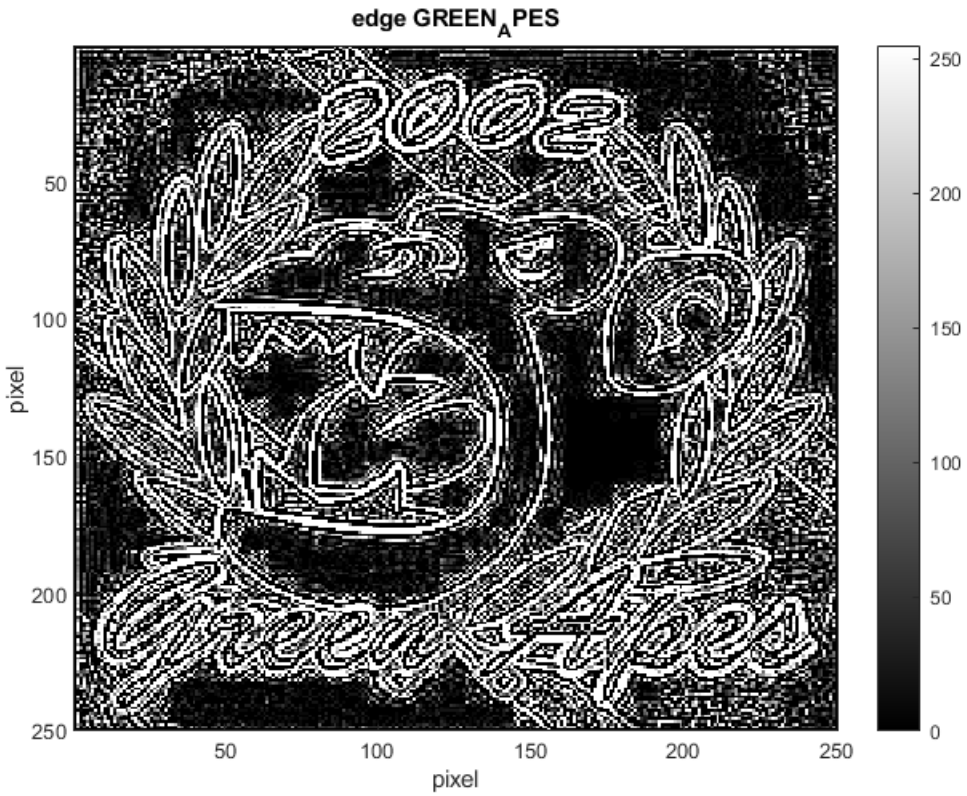



```
figure ;
imagesc(edge_haifa)
colormap("gray");
colorbar;
title('edge haifa','FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",1
```


```
figure ;
imagesc(processed_GREEN_APES)
colormap("gray");
colorbar;
title(tle4,'FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",14); set(
```



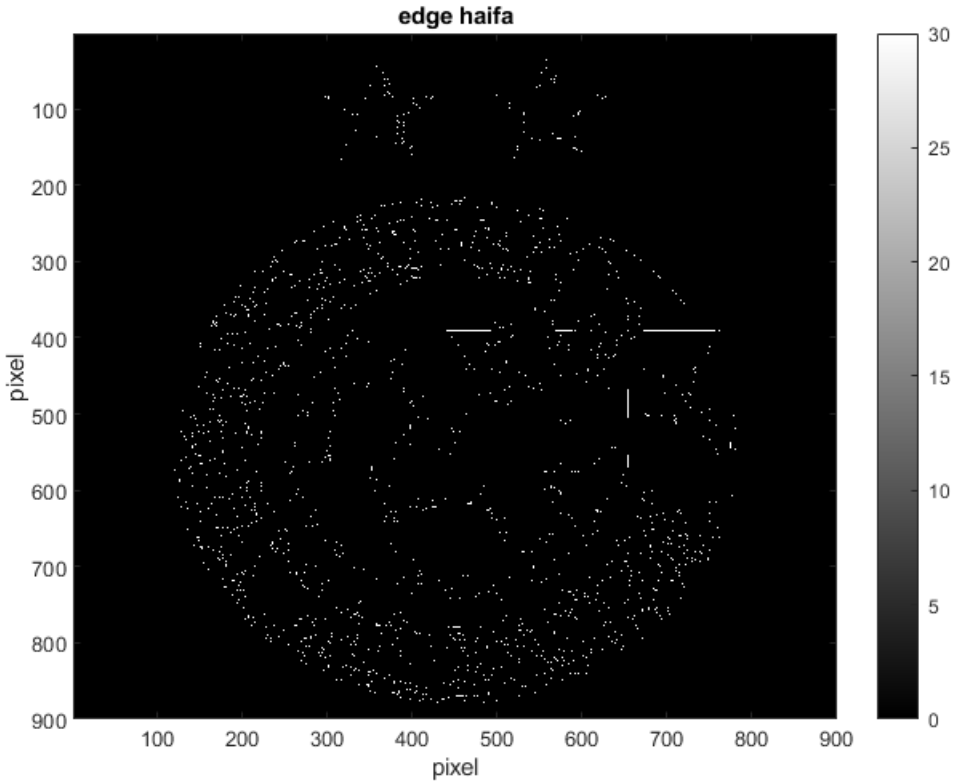

```
figure ;
imagesc(edge_GREEN_APES)
colormap("gray");
colorbar;
title('edge GREEN_APES','FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSi
```



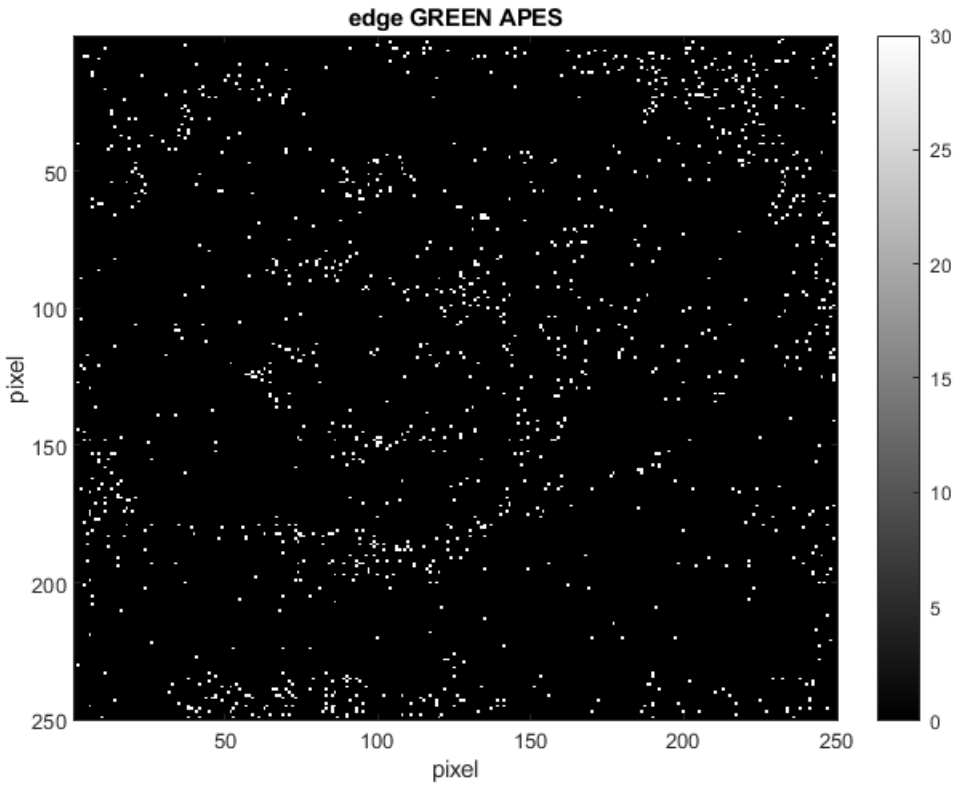

```
elapsed_time = toc;
mins = floor(elapsed_time / 60);
secs = round(mod(elapsed_time, 60));
fprintf('Total elapsed time: %02d:%02d\n', mins, secs);
```

```
Total elapsed time: 00:01
```

```
% marking edges
alpha = 30 ;
edge_haifa(edge_haifa > alpha) = 1;
edge_haifa(edge_haifa < alpha) = 0;
edge_GREEN_APES(edge_GREEN_APES > alpha) = 1;
edge_GREEN_APES(edge_GREEN_APES < alpha) = 0;
figure ;
imagesc(edge_haifa)
colormap("gray");
colorbar;
title('edge haifa','FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSize",1
```




```
figure ;
imagesc(edge_GREEN_APES)
colormap("gray");
colorbar;
title('edge GREEN APES','FontSize',18); xlabel("pixel", "FontSize",14); ylabel("pixel", "FontSi
```



function

Logarithmic Pixel processing

```
function abs_image_processed = Logarithmic(abs_image)
    type = class(abs_image);
    abs_image = double(abs_image) ;
    c = (double(intmax(type))) ./ (log10(double(intmax(type)) + 1)) ;
    abs_image_processed = c* (log10(abs_image+1)) ;
    if type == 'uint16'
        abs_image_processed = uint16(abs_image_processed) ;
    elseif type == 'uint8'
        abs_image_processed = uint8(abs_image_processed) ;
    end
end
```

Median filter

```
function processed_image = median(image,fx,fy)

    image_padd = padarray(image, [(fy-1)/2 (fx-1)/2], 0) ;

    processed_image = zeros(size(image), class(image)) ;
```

```
    i = 1 ;

    for row_step = 1 : size(image_padd,1) - fy + 1
        j = 1 ;

        for col_step = 1 : size(image_padd,2) - fx + 1

            mask = image_padd(row_step : row_step + fy - 1 ,col_step : col_step + fx - 1) ;

            b = sort(reshape(mask,1,numel(mask))) ;

            if mod(size(b), 2) == 1

                median = b((length(b)+1)/2);

            else

                first_median = b((length(b))/2);
                second_median = b((length(b)+2)/2);
                median = (first_median + second_median) / 2 ;

            end
            processed_image(i,j) = median ;
            j = j + 1 ;
        end
        i = i + 1 ;
    end

end
```

gaussian filter

```
function filtered_image = gaussian_filter(image, fx, fy, var_x, var_y)

% Create the Gaussian mask
[X,Y] = meshgrid(-(fx-1)/2:(fx-1)/2, -(fy-1)/2:(fy-1)/2);
gaussian_mask = exp(-(X.^2/(2*var_x^2)+Y.^2/(2*var_y^2)));
gaussian_mask = gaussian_mask / sum(gaussian_mask(:));

% Convolve the image with the Gaussian mask
filtered_image = conv2(double(image), double(gaussian_mask), 'same');

% Convert the output to the same type as the input
filtered_image = cast(filtered_image, class(image));

end
```

bilateral filter

```
function processed_image = bilateral_filter(image, fx, fy, spatial_var, radiometric_var)
```

```
    type = class(image);

    % Calculate the padding required for the input image

    image_padd = padarray(image, [(fy-1)/2 (fx-1)/2], 0) ;

    % Initialize the output image
    processed_image = zeros(size(image), class(image)) ;


    % Create the Gaussian spatial kernel
    [X,Y] = meshgrid(-(fx-1)/2:(fx-1)/2, -(fy-1)/2:(fy-1)/2);
    spatial_kernel = exp(-0.5*(X.^2 + Y.^2)/(spatial_var^2));
    spatial_kernel = spatial_kernel ./ sum(spatial_kernel(:)) ;

    i = 1 ;
    for row_step = 1 : size(image_padd,1) - fy + 1
        j = 1 ;

        for col_step = 1 : size(image_padd,2) - fx + 1

            mask = double(image_padd(row_step : row_step + fy - 1 ,col_step : col_step + fx - 1
            pixel = mask((fy+1)/2 , (fx+1)/2) ;
            % Create the radiometric kernel
            radiometric_kernel = exp(-(abs(mask-pixel).^2/(2*radiometric_var^2)));


            % Calculate the weights for each pixel in the neighborhood
            weights = radiometric_kernel .* spatial_kernel ;

            % Normalize the weights
            weights = weights / sum(weights(:));

            % Calculate the filtered pixel value
            processed_image(i,j) = round(sum(mask(:) .* weights(:)));
            j = j + 1 ;

        end
        i = i + 1 ;
    end


end
```

Edge highlight

```
function [output_image, edge_image] = edge_highlight(image, c)

    % Convert the input image to double type
    image1 = double(image);
```

```
    % Create a Laplacian filter
    laplacian_filter = [0 1 0;1 -4 1; 0 1 0];

    % Compute the Laplacian of the input image using the Laplacian filter
    laplacian_image = conv2(image1, laplacian_filter, 'same');

    % Compute the output image by adding the Laplacian image to the input image

    edge_image = c*laplacian_image;
    output_image = image1 + edge_image;


    % Convert the output image to the same type as the input image
    output_image = cast(output_image, class(image));
    edge_image = cast(edge_image, class(image));
end
```

print image stats

```
function print_image_stats(image_names, abs_HH_processed, abs_VV_processed, haifa_gray, GREEN_A

    % Calculate mean and std for the original and processed images
    std_original_HH = std(double(abs_HH_processed(:)));
    mean_original_HH = mean(double(abs_HH_processed(:)));
    std_original_VV = std(double(abs_VV_processed(:)));
    mean_original_VV = mean(double(abs_VV_processed(:)));
    std_original_haifa = std(double(haifa_gray(:)));
    mean_original_haifa = mean(double(haifa_gray(:)));
    std_original_GREEN_APES = std(double(GREEN_APES_gray(:)));
    mean_original_GREEN_APES = mean(double(GREEN_APES_gray(:)));

    std_processed_HH = std(double(processed_HH(:)));
    mean_processed_HH = mean(double(processed_HH(:)));
    std_processed_VV = std(double(processed_VV(:)));
    mean_processed_VV = mean(double(processed_VV(:)));
    std_processed_haifa = std(double(processed_haifa(:)));
    mean_processed_haifa = mean(double(processed_haifa(:)));
    std_processed_GREEN_APES = std(double(processed_GREEN_APES(:)));
    mean_processed_GREEN_APES = mean(double(processed_GREEN_APES(:)));

    % Create the table
    image_stats_table = table(image_names, ...
        [mean_original_HH; mean_original_VV; mean_original_haifa; mean_original_GREEN_APES], ..
        [std_original_HH; std_original_VV ; std_original_haifa ;std_original_GREEN_APES], ...
        [mean_processed_HH; mean_processed_VV; mean_processed_haifa; mean_processed_GREEN_APES]
        [std_processed_HH; std_processed_VV ; std_processed_haifa ; std_processed_GREEN_APES],
        'VariableNames', {'Image name', 'Mean original', 'Std original', 'Mean processed', 'Std

    % Display the table
```

```
    disp(image_stats_table)
end
```