1. Objectives:
   - Observing the effects of using averaging filters for noise reduction and large object extraction.
   - Enhancing the images using statistical-order filters, especially the median filter.
   - Enhancing the fine details in images using sharpening filters.

2. Theory:

   The idea of spatial filters is to define a filter (mask) in the spatial domain, slide it over the image to be processed, and apply some operation in the gray levels of the image pixels under the mask. This process is repeated at each pixel in the input image and a new pixel is formed in the output image at the same position as the pixel under the center of the mask. Figure (1) illustrates the mechanisms of spatial filtering.

![Figure (1): The Mechanism of Spatial Filtering](image_url)
There are several types of spatial filters, each of which is excellent in a certain application:

I) Linear Spatial Filters:

The linear operations of interest in this section consist of multiplying each pixel in the neighborhood by a corresponding coefficient of the mask and summing the results to obtain a pixel in the output image. For a mask of size $m \times n$, we assume typically that $m = 2a + 1$ and $n = 2b + 1$, where $a$ and $b$ are nonnegative integers. Thus, linear filtering is described by:

$$g(x, y) = \sum_{s=-a}^{a-1} \sum_{t=-b}^{b-1} w(s, t) f(x + s, y + t)$$

The spatial averaging filters are example of linear filters which are used for blurring (smoothing) images, for noise reduction, and for large object extraction (in image recognition). The output of the spatial averaging filter is simply the average of the pixels contained in the neighborhood of the mask as shown by the following equation:

$$g(x, y) = \frac{\sum_{s=-a}^{a-1} \sum_{t=-b}^{b-1} w(s, t) f(x + s, y + t)}{\sum_{s=-a}^{a-1} \sum_{t=-b}^{b-1} w(s, t)}$$

II) Statistical-Order Filters:

Statistical-order filters also use pixel neighborhoods but do not explicitly use coefficients. The most famous application is noise reduction by median gray-level value computation in the neighborhood of the mask, which is called Median filtering. The Median filter is excellent in removing impulsive noise (called salt-and pepper noise in the image processing literature).

III) Sharpening-Order Filters:

Sharpening filters are used to enhance fine details that have blurred, either by error or as a natural effect of particular method of image acquisition. Using of image sharpening varies and includes applications ranging from electronic printing and medical merging imaging to industrial inspection and military systems. The basic idea is to define a mask capable of measuring the changes in the gray levels of the input image. This is based on 1st and 2nd derivatives, which can be implemented digitally by differences of gray levels in the neighborhoods. The Laplacian is the most famous operator that can define the 2nd derivative. It is capable of detecting slowly varying (ramp), isolated points, thin lines, and step variation in the gray levels of the input image. Moreover, it does not respond to constant gray level transitions. It can be defined as:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$
### 3. Experimental work:

1. Apply the average filter on: **avg.jpg, avg1.jpg and avg2.jpg.**
2. Apply the median filter on: **med1.jpg to med7.jpg.**
3. Apply the sharpen filter on: **fig5_2.tif, fig5_3.tif, and fig5_5.jpg**
4. Comment on each part individually.

```matlab
clear all
clc

% I) for blurring and noise reduction:
I=imread('avg.tif'); imshow(I)
M3=fspecial('average',3);
M9=fspecial('average',9);
M15=fspecial('average',15);
M35=fspecial('average',35);
J3=imfilter(I,M3);
J9=imfilter(I,M9);
J15=imfilter(I,M15);
J35=imfilter(I,M35);
figure(2), subplot(221),subimage(J3),title('Filtered by 3X3'),axis off
subplot(222),subimage(J9),title('Filtered by 9X9'),axis off
subplot(223),subimage(J15),title('Filtered by 15X15'),axis off
subplot(224),subimage(J35),title('Filtered by 35X35'),axis off
pause

% II) for large object extraction:
I=imread('avg1.jpg'); imshow(I)
figure
M15=fspecial('average',15);
J15=imfilter(I,M15);
M25=fspecial('average',25);
J25=imfilter(I,M25);
K15=im2bw(J15);
K25=im2bw(J25);
subplot(321),subimage(I),title('original'),axis off
subplot(322),subimage(J15),title('Filtered by 15X15'),axis off
subplot(323),subimage(J25),title('Filtered by 25X25'),axis off
subplot(324),subimage(K15),title('Thr.15X15'),axis off
subplot(325),subimage(J35),title('Filtered by 35X35'),axis off
subplot(326),subimage(K35),title('Thr.25X25'),axis off

% 3) Median Filter:
I=imread('med1.jpg'); imshow(I)
figure
Ic=medfilt2(I);
subplot(131),subimage(I),title('original'),axis off
subplot(132),subimage(Ic),title('Filtered'),axis off
I=imread('med1.jpg'); imshow(I)
figure
M3=fspecial('average',3);
figure
Ic=medfilt2(I);
```
4. Home Exercise:

   I) Write a function that implements the median filter, please do not use the median function built in the MATLAB.