EE1F2 Multimedia Data

Laboratory Exercise 1: Matlab Image Processing

By Luis Hernandez and Dr. Sandra I. Woolley

After creating a "1f2" folder in your Y:\ filespace and create a subdirectory inside it called "lab1", download the laboratory 1 resource files (ee1f2lab1files.zip) by right clicking and selecting "Save Target As" on the link on the course web page. Unzip these files into your lab1 directory and open Matlab. Note it may take a minute to load. When it opens, change the current directory on the main toolbar to Y:\1f2\lab1 so that it will find the files it needs to work with. You can do this by clicking on the button with dots on and browsing to "My Documents" where you can select the directory 1f2 and then lab1.

Your Matlab screen should look like Figure 1.

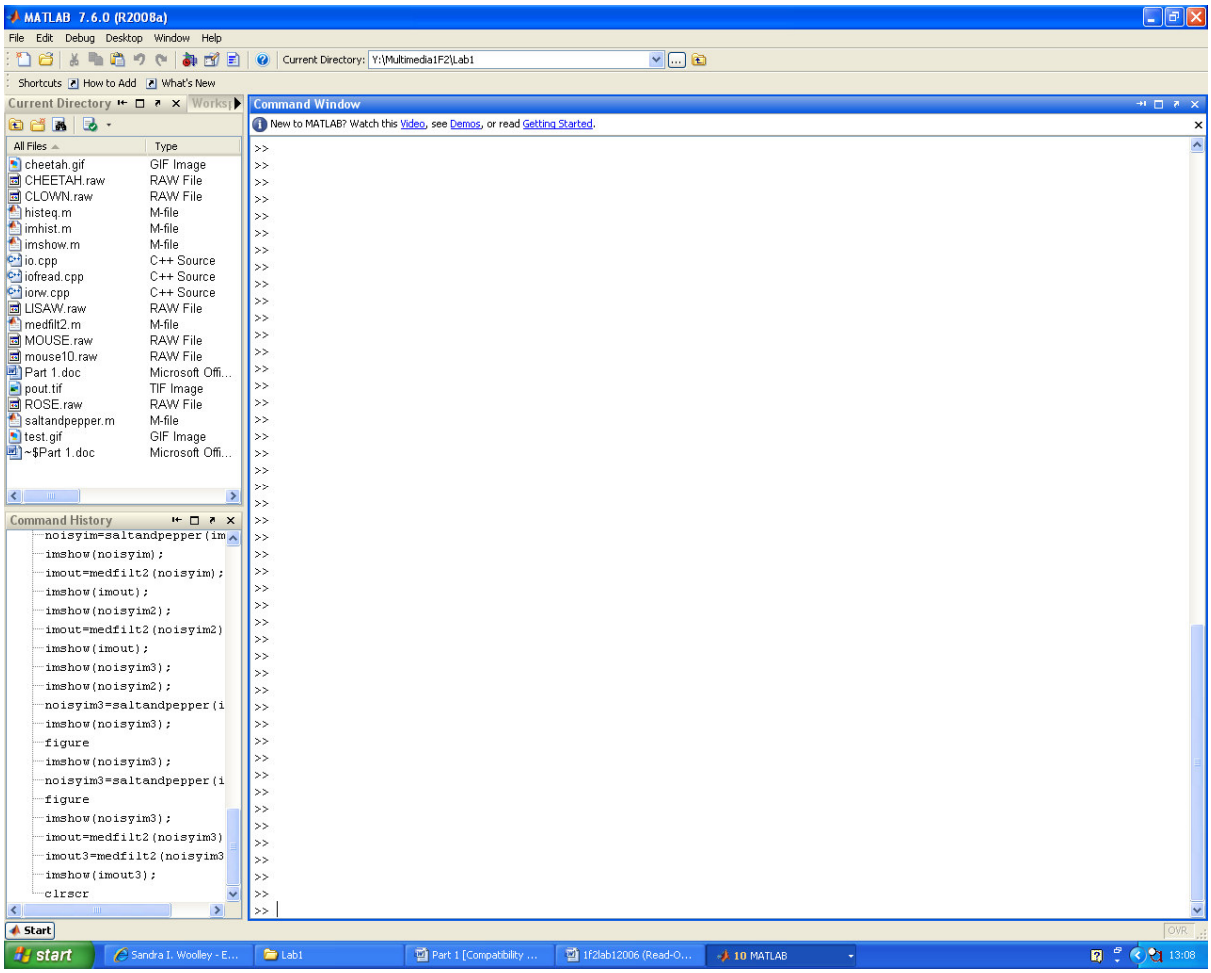

Figure 1. Matlab window.

Part 1: Histogram Manipulation

```
>> im=imread('pout.tif');
>> imshow(im);
```

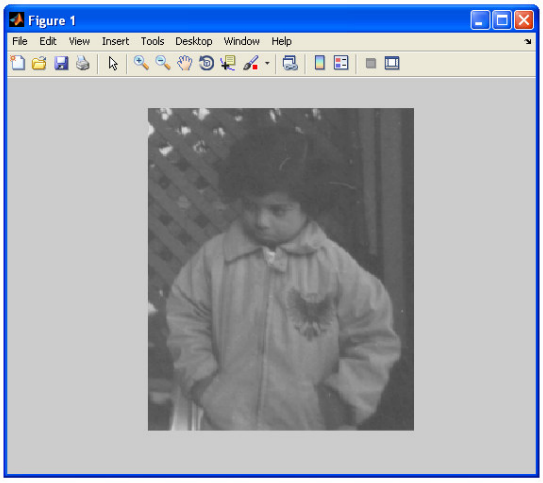

Figure 2. Loading the sample image. “pout.tif”

1.1 Analysing the image before the histogram equalisation:

```
>> im=imread('pout.tif');
>> imshow(im);
>> figure
>> imhist(im);
```

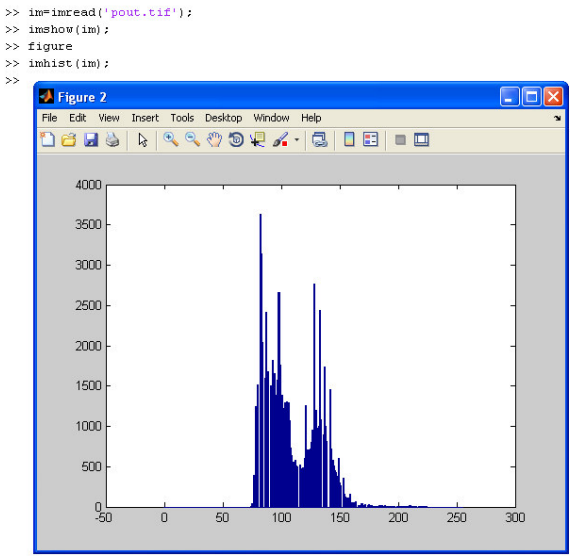

Figure 3. “pout.tif” histogram (Before equalisation).

1.2 Applying histogram equalisation:

```
>> imout=histeq(im);
>> figure
```

>> imshow(imout);

```
>> im=imread('pout.tif');
>> imshow(im);
>> figure
>> imhist(im);
>> imout=histeq(im);
>> figure
>> imshow(imout);
>>
```

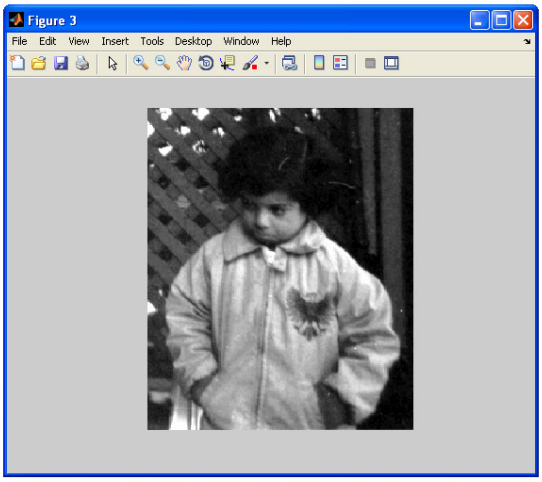

Figure 4. “pout.tif” image after the equalisation.

>> imshow(imout);
>> imhist(imout);

```
>> imhist(imout);
>>
```

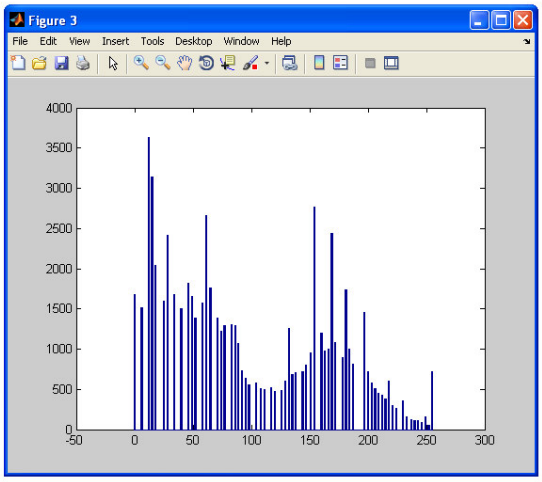

Figure 5. “pout.tif” histogram after the equalisation.

Part 2: Filtering

a) Low pass filtering

>> lp=[1 1 1; 1 1 1; 1 1 1]/9

lp =

0.1111	0.1111	0.1111
0.1111	0.1111	0.1111
0.1111	0.1111	0.1111

```
>> imout=conv2(double(im),lp);
>> imshow(imout);
```

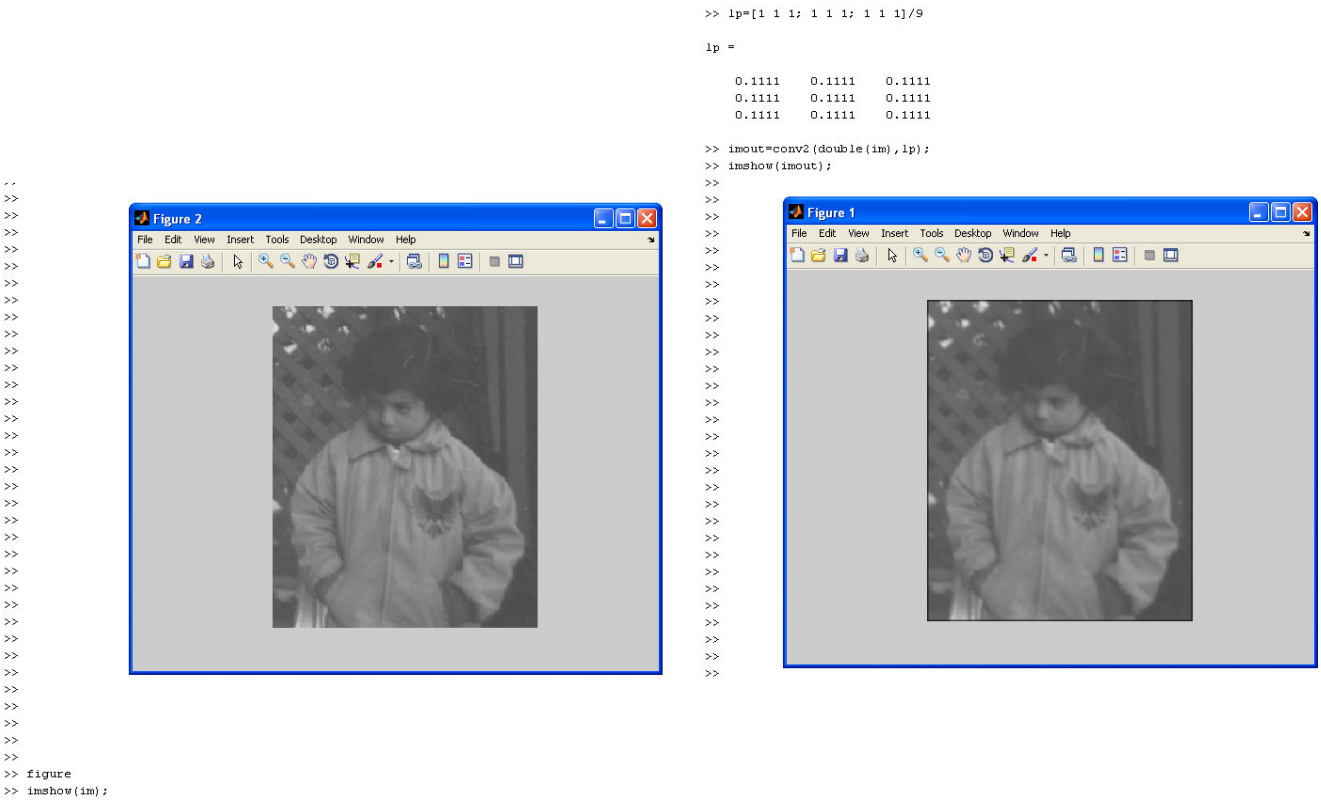

Figure 6. “pout.tif” before(left)/after(right) applying the low pass filtering.

b) High pass filtering

```
>> figure
>> imshow(im);
>> hp=[0 -1 0; -1 4 -1; 0 -1 0]

hp =

     0    -1     0
    -1     4    -1
     0    -1     0

>> figure
>> imout=conv2(double(im),hp);
>> imshow(imout);
```

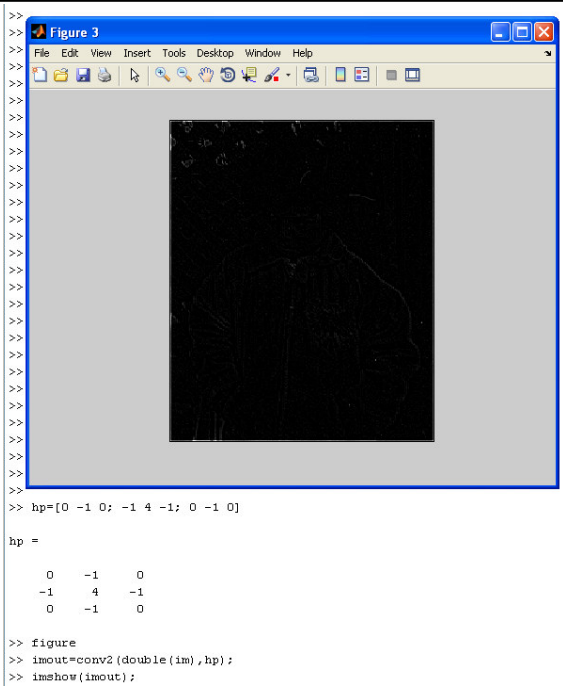

Figure 7. "pout.tif" after applying the high pass filtering.

b.1) Adding an offset to view better the image

>> imshow(imout+128);

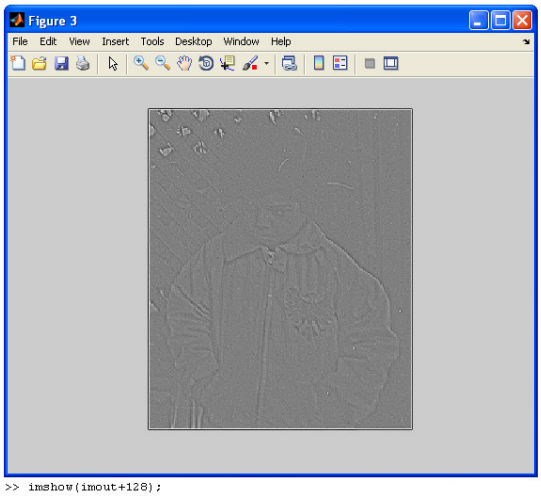

Figure 8. "pout.tif" after applying the high pass filtering and an offset.

c) Sobel Template

c.1) Applying Sobel Template 1

>> s1=[-1 -2 -1;0 0 0; 1 2 1]/3

```
s1 =

  -0.3333   -0.6667   -0.3333
         0         0         0
    0.3333    0.6667    0.3333

>> imout=conv2(double(im),s1);
>> figure
>> imshow(imout);
>> imshow(imout+128);
```

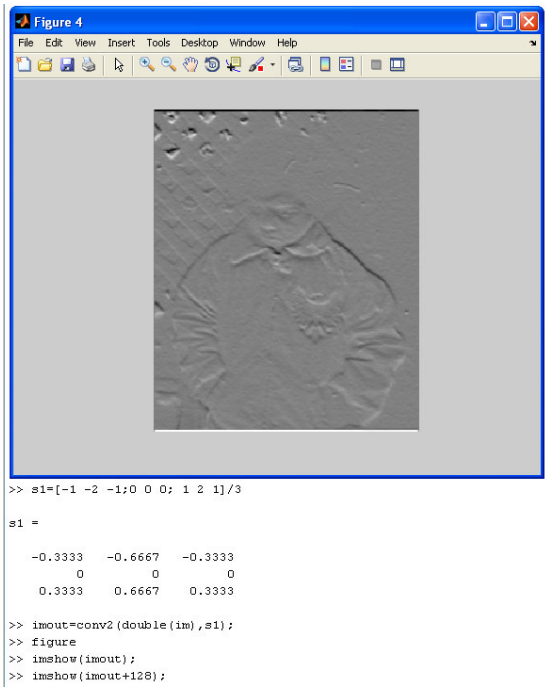

Figure 9. “pout.tif” after applying Sobel template 1 and an offset.

c.2) Applying Sobel Template 2

```
>> s2=[-1 0 1;-2 0 2;-1 0 1]/3

s2 =

  -0.3333         0    0.3333
  -0.6667         0    0.6667
  -0.3333         0    0.3333

>> imout=conv2(double(im),s2);
>> figure
>> imshow(imout+128);
```

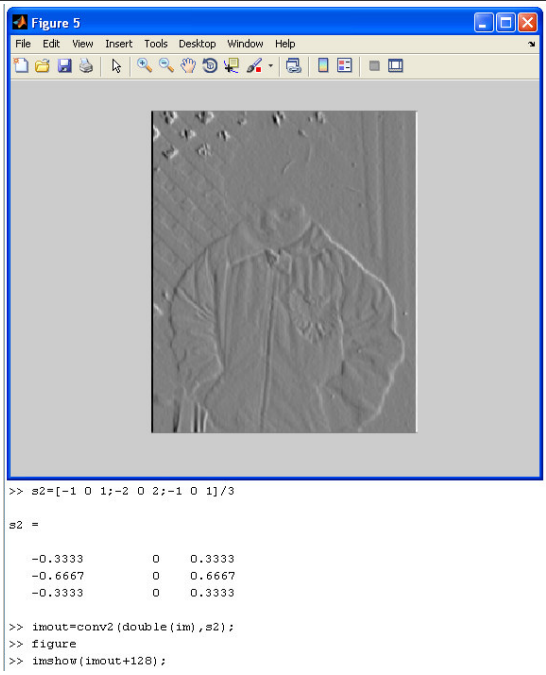

Figure 10. "pout.tif" after applying Sobel template 2 and an offset.

c.3) Applying S1 and S2 to the image (one after the other)

```
>> imout=conv2(double(im),s1);
>> imout=conv2(double(imout),s2);
>> imshow(imout+128);
```

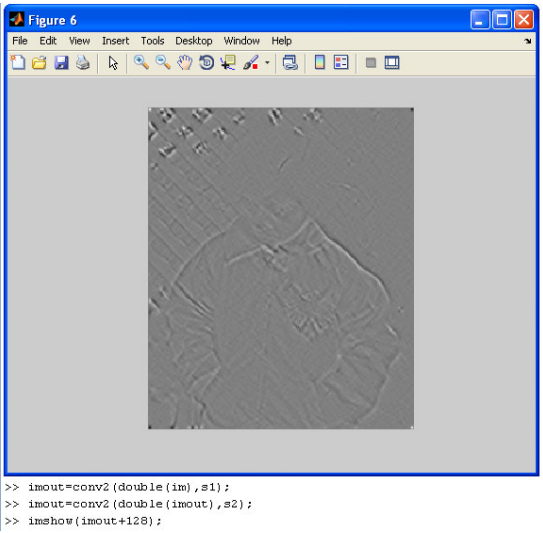

Figure 11. "pout.tif" after been filtered by Sobel 1 and Sobel 2 templates (with an offset).

d) Median filter

```
>> imout=medfilt2(im);
>> imshow(imout);
```

Figure 12. "pout.tif" after applying a Median filter.

d.1) Adding noise to the image

Noisy density: 0.02

```
>> noisyim=saltandpepper(im,0.02);
>> figure
>> imshow(noisyim);
```

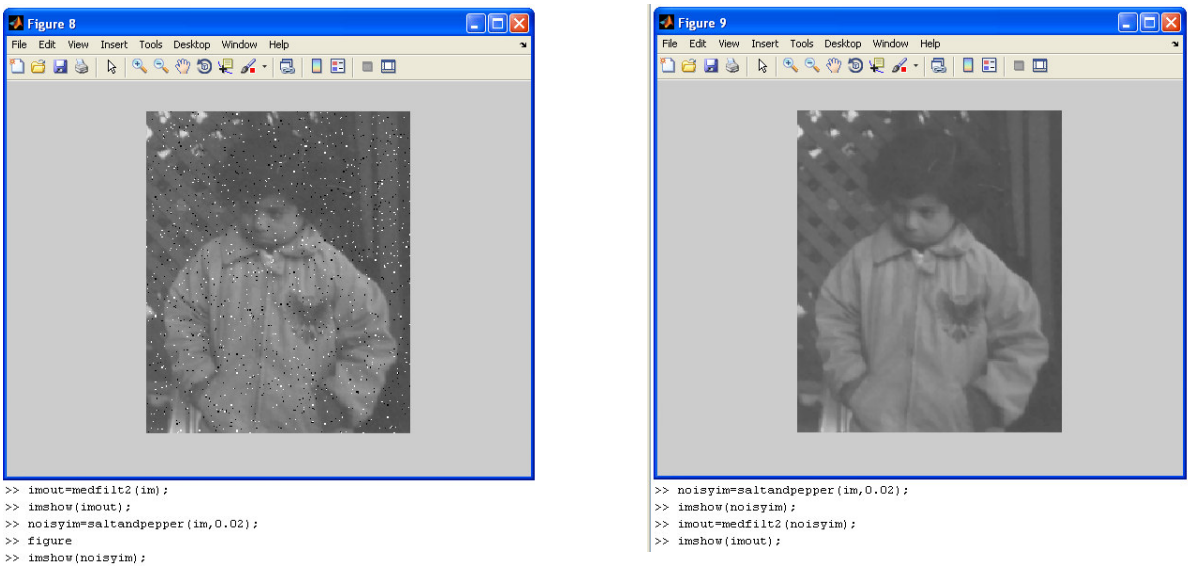

Figure 13. Left: "pout.tif" with salt and pepper noise (0.02); Right: after applying a Median filter.

Noisy density: 0.2

```
>> noisyim2=saltandpepper(im,0.2);
>> figure
>> imshow(noisyim2);
```

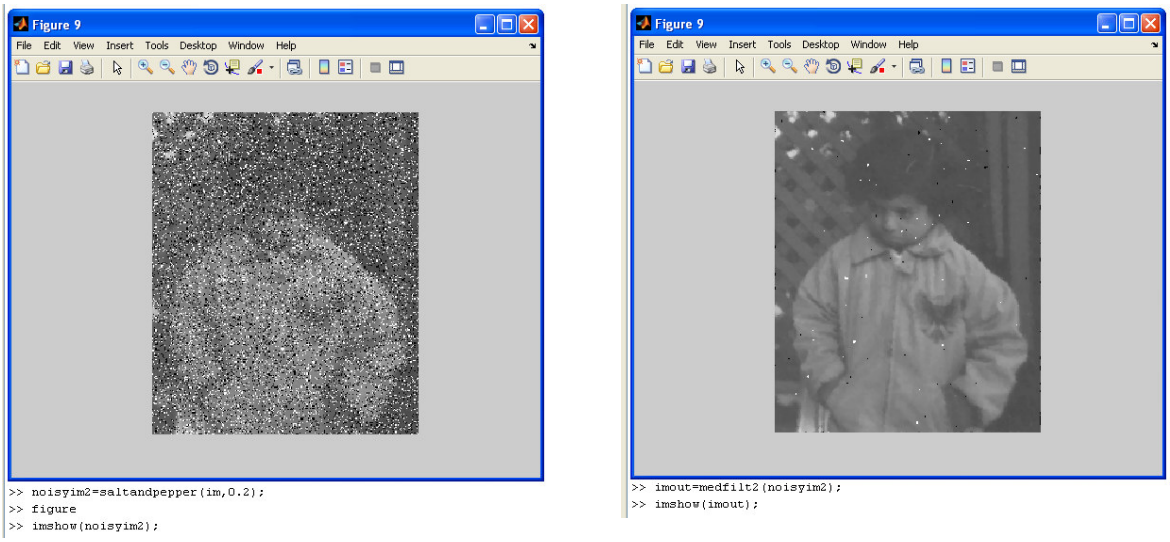

Figure 14. Left: "pout.tif" with salt and pepper noise (0.2); Right: after applying a Median filter.

Noisy density: 0.5

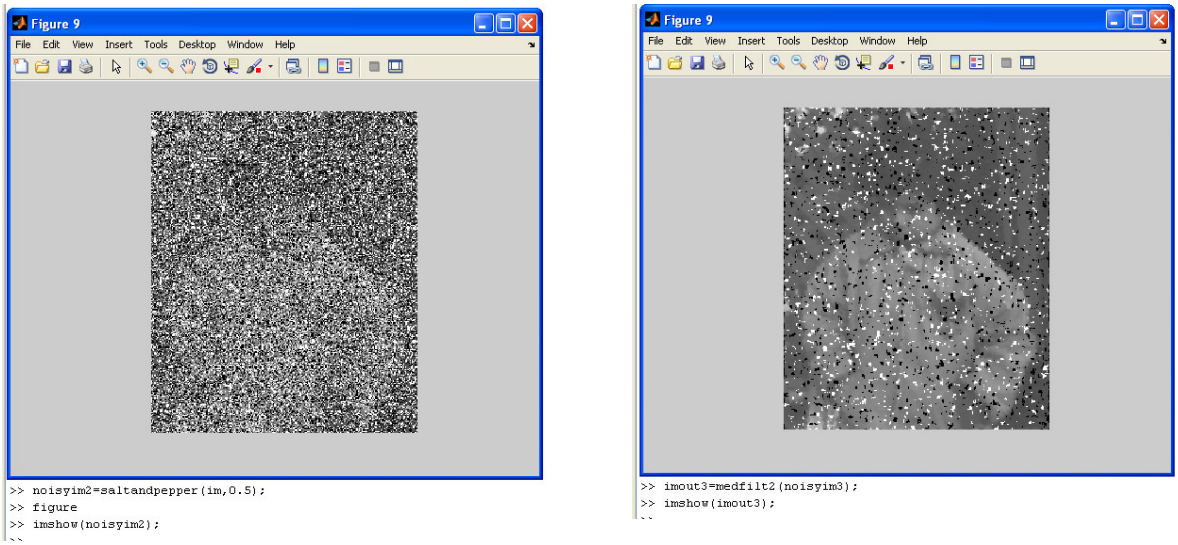

Figure 15. Left: "pout.tif" with salt and pepper noise (0.5); Right: after applying a Median filter.

EE1F2 Multimedia Data

Laboratory 1 Assessment Sheet : Matlab Image Processing

NAME: (print clearly please) ……

SIGNATURE: ……..

Part 1

Describe the effect of histogram equalization (explaining briefly why this is the case).

Histogram equalization increases the dynamic range of pixel values stretching the values across the range. It will have the effect of increasing contrast. Very useful in making low contrast images features more visible to the human eye.

Part 2

a) Comment on the results of the 3x3 low-pass filter (again explaining why this is the case).

It is a low pass averaging filter – averages over 9 pixels. This has the effect of blurring or smoothing the image.

b) Comment on the results of the high-pass filter (explaining why) - compare with the results of the low-pass filter.

It “detects” or highlights edges –removes low frequencies. Complements with the low pass filter.

c) Comment on the results of the Sobel filters (s1 and s2).

These are high pass horizontal and vertical filters. S1 – Horizontal edge detector S2 – Vertical edge detector

d) Describe the effects of different values of 'salt&pepper' noise *and* median filtering.

Media filtering better preserves edges but remove outliers type noise (as found in “salt and pepper”). The lower the noisy density (salt and pepper noisy), the easier to remove it using median filter. Median filter removes outliers, cleaning up the images from noise.