

EE3731C – Signal Processing Methods

Qi Zhao
Assistant Professor
ECE, NUS

Matlab and Image Processing Examples

Matlab and Images

- An image in MATLAB is a matrix
- Every pixel is a matrix element
- All the operators in MATLAB defined on matrices can be used on images: $+$, $-$, $*$, $/$, $^$, sqrt , sin , cos etc.

The [help](#) in Matlab is very helpful.

Images in MATLAB

- Intensity images: m-by-n
- RGB images: m-by-n-by-3

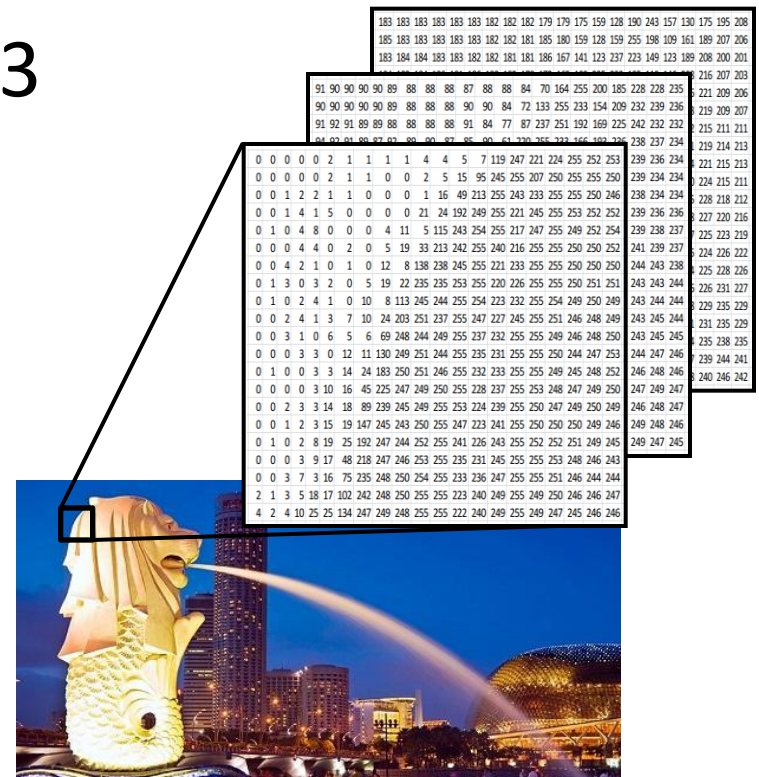
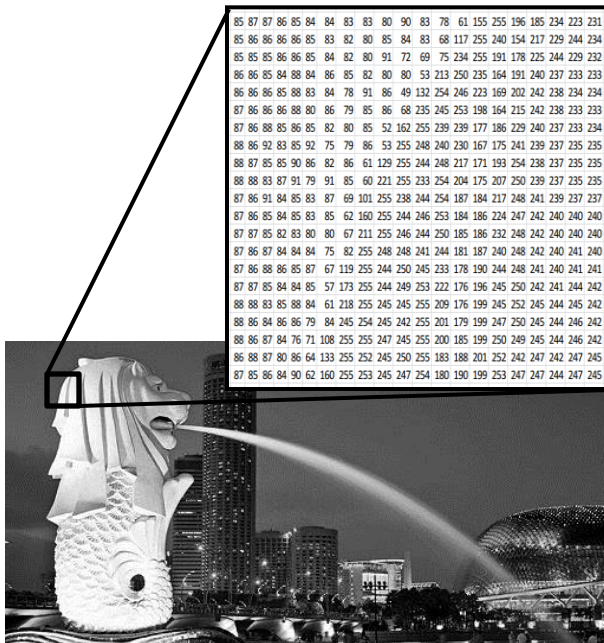


Image Import and Export

- Read and write images in Matlab

```
>> I = imread('singapore_rgb.jpg');
```

```
>> figure;
```

```
>> imshow(I);
```

```
>> size(I)
```

```
ans =
```

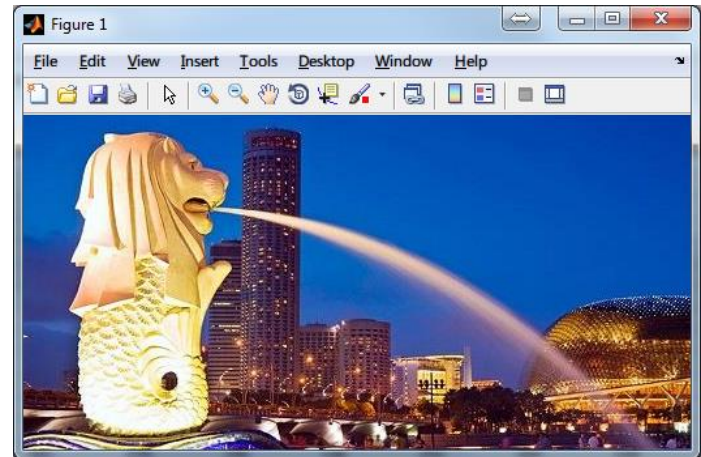
```
    256    512     3
```

```
>> I_gray = rgb2gray(I);
```

```
>> figure;
```

```
>> imshow(I_gray);
```

```
>> imwrite(I_gray, 'singapore_gray.jpg');
```



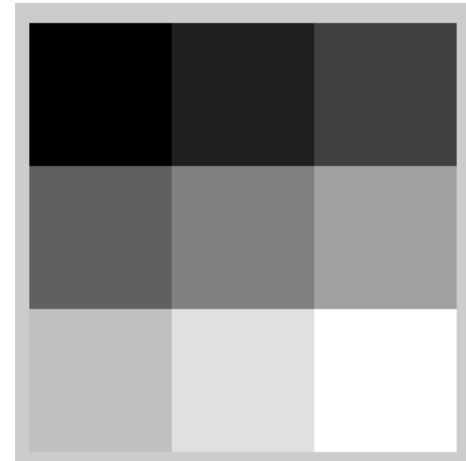
Images and Matrices

- How to build a matrix (or image)?

```
>> A = [ 1 2 3; 4 5 6; 7 8 9 ]
```

A =

1	2	3
4	5	6
7	8	9



Images and Matrices

- Accessing image elements (row, column)

```
>> A(2,1)
```

```
ans =
```

```
4
```

- Extract a whole column or row

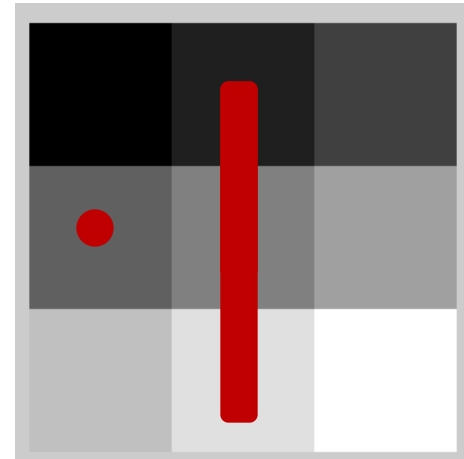
```
>> A(:,2)
```

```
ans =
```

```
2
```

```
5
```

```
8
```



A =

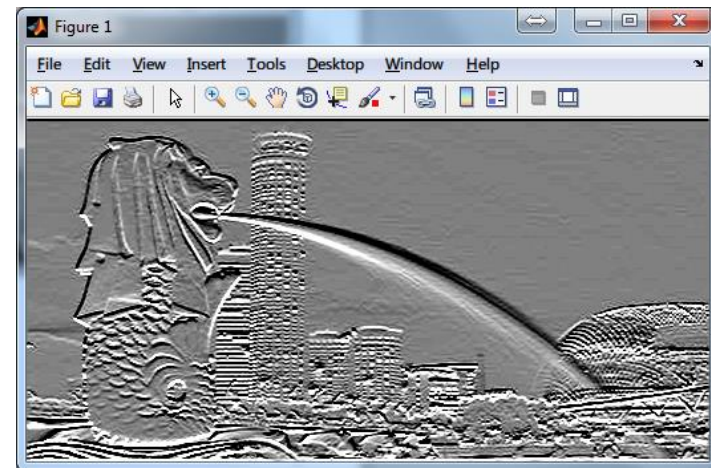
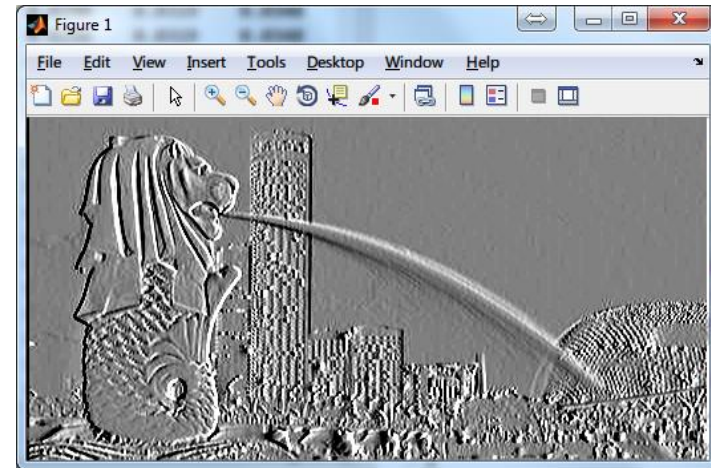
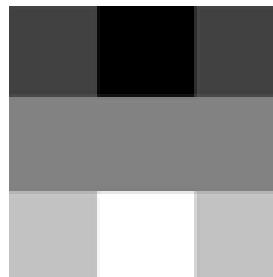
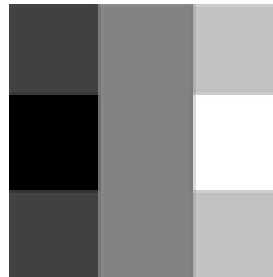
1	2	3
4	5	6
7	8	9

Sobel Filter

Kernel = $\begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix};$

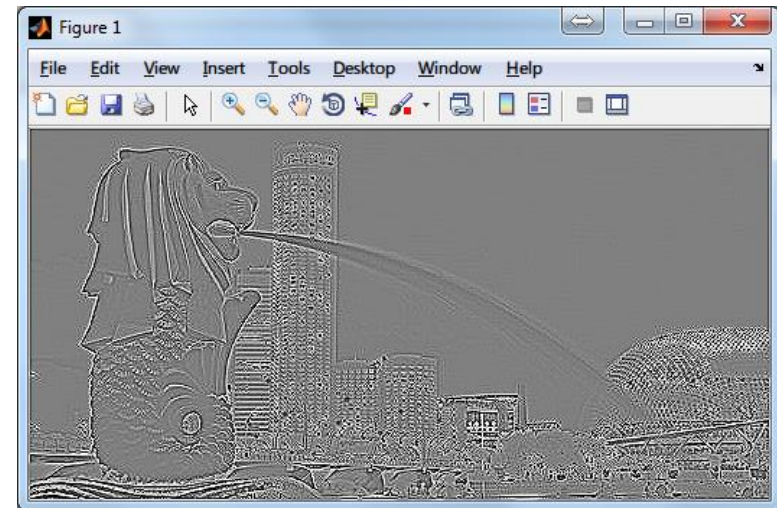
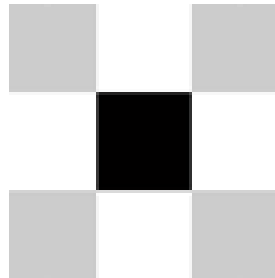
J1 = imfilter(I, kernel, 'replicate');

J2 = imfilter(I, kernel', 'replicate');



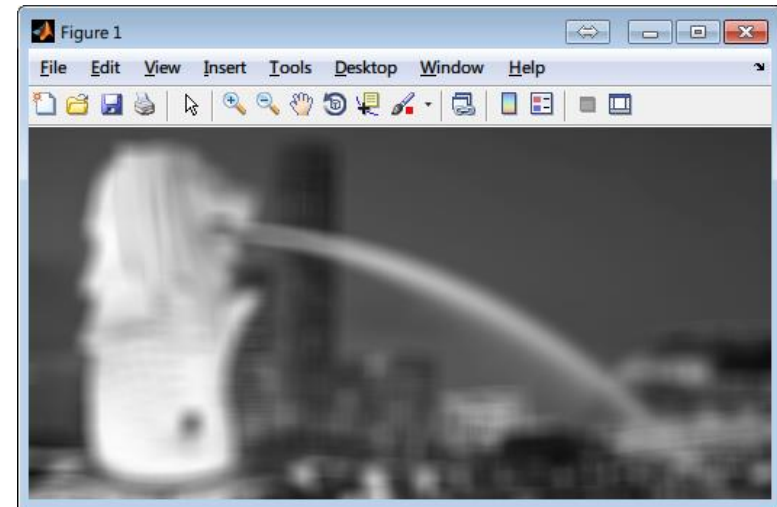
Laplacian Filter

Kernel = $\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$;



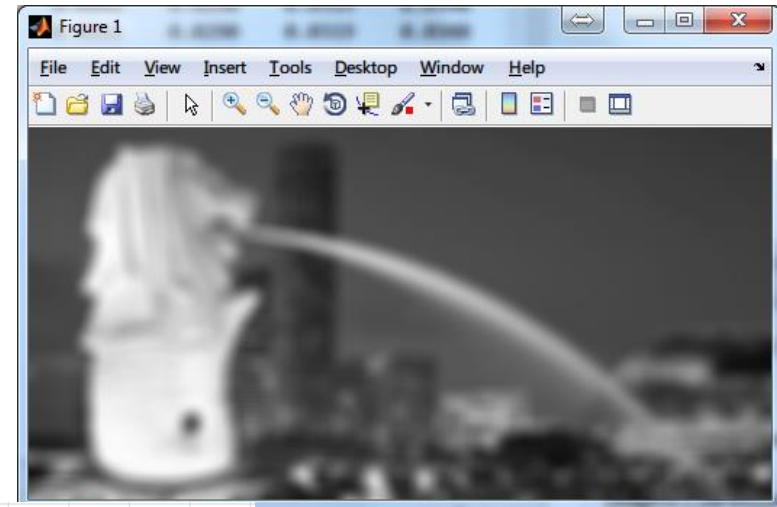
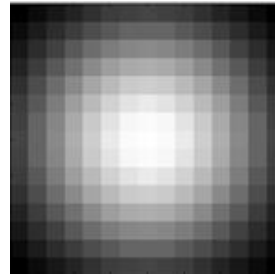
Mean Filter

Kernel = ones(15,15) / 15^2;



Gaussian Filter

Kernel = fspecial('gaussian', [15 15], 5);

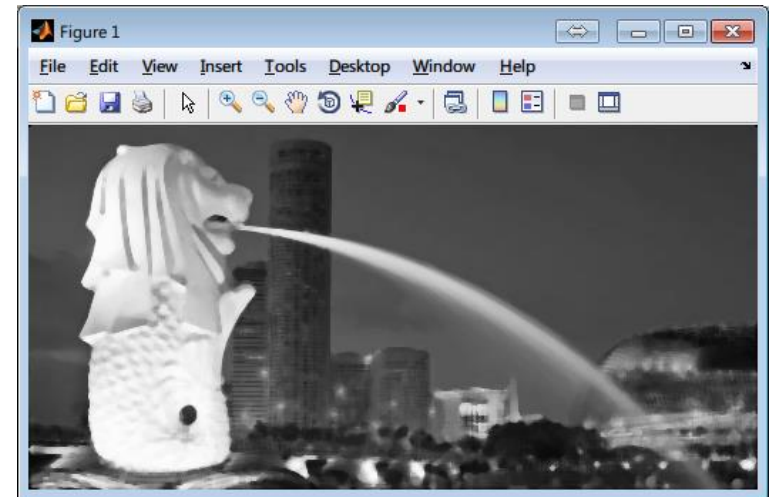


0.0012	0.0015	0.0019	0.0023	0.0027	0.0029	0.0031	0.0032	0.0031	0.0029	0.0027	0.0023	0.0019	0.0015	0.0012
0.0015	0.0020	0.0025	0.0030	0.0034	0.0038	0.0040	0.0041	0.0040	0.0038	0.0034	0.0030	0.0025	0.0020	0.0015
0.0019	0.0025	0.0031	0.0037	0.0043	0.0047	0.0050	0.0051	0.0050	0.0047	0.0043	0.0037	0.0031	0.0025	0.0019
0.0023	0.0030	0.0037	0.0045	0.0051	0.0057	0.0060	0.0061	0.0060	0.0057	0.0051	0.0045	0.0037	0.0030	0.0023
0.0027	0.0034	0.0043	0.0051	0.0059	0.0065	0.0069	0.0071	0.0069	0.0065	0.0059	0.0051	0.0043	0.0034	0.0027
0.0029	0.0038	0.0047	0.0057	0.0065	0.0072	0.0077	0.0078	0.0077	0.0072	0.0065	0.0057	0.0047	0.0038	0.0029
0.0031	0.0040	0.0050	0.0060	0.0069	0.0077	0.0081	0.0083	0.0081	0.0077	0.0069	0.0060	0.0050	0.0040	0.0031
0.0032	0.0041	0.0051	0.0061	0.0071	0.0078	0.0083	0.0085	0.0083	0.0078	0.0071	0.0061	0.0051	0.0041	0.0032
0.0031	0.0040	0.0050	0.0060	0.0069	0.0077	0.0081	0.0083	0.0081	0.0077	0.0069	0.0060	0.0050	0.0040	0.0031
0.0029	0.0038	0.0047	0.0057	0.0065	0.0072	0.0077	0.0078	0.0077	0.0072	0.0065	0.0057	0.0047	0.0038	0.0029
0.0027	0.0034	0.0043	0.0051	0.0059	0.0065	0.0069	0.0071	0.0069	0.0065	0.0059	0.0051	0.0043	0.0034	0.0027
0.0023	0.0030	0.0037	0.0045	0.0051	0.0057	0.0060	0.0061	0.0060	0.0057	0.0051	0.0045	0.0037	0.0030	0.0023
0.0019	0.0025	0.0031	0.0037	0.0043	0.0047	0.0050	0.0051	0.0050	0.0047	0.0043	0.0037	0.0031	0.0025	0.0019
0.0015	0.0020	0.0025	0.0030	0.0034	0.0038	0.0040	0.0041	0.0040	0.0038	0.0034	0.0030	0.0025	0.0020	0.0015
0.0012	0.0015	0.0019	0.0023	0.0027	0.0029	0.0031	0.0032	0.0031	0.0029	0.0027	0.0023	0.0019	0.0015	0.0012

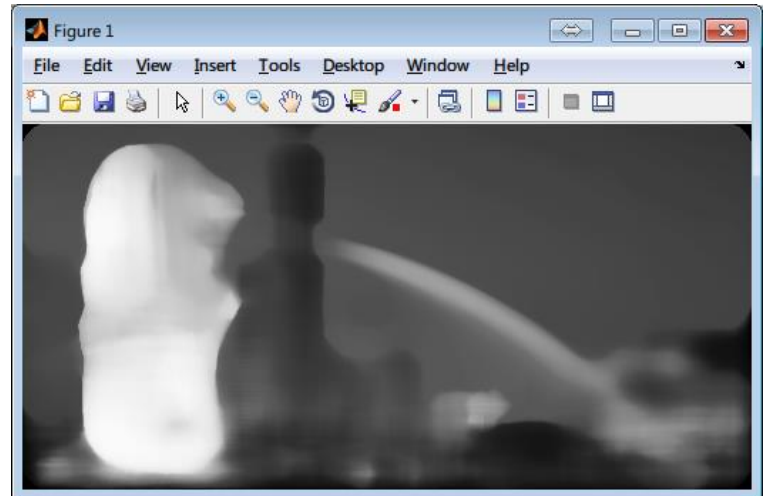
Median Filter



5x5



25x25



Sharpening



$\times 2 -$

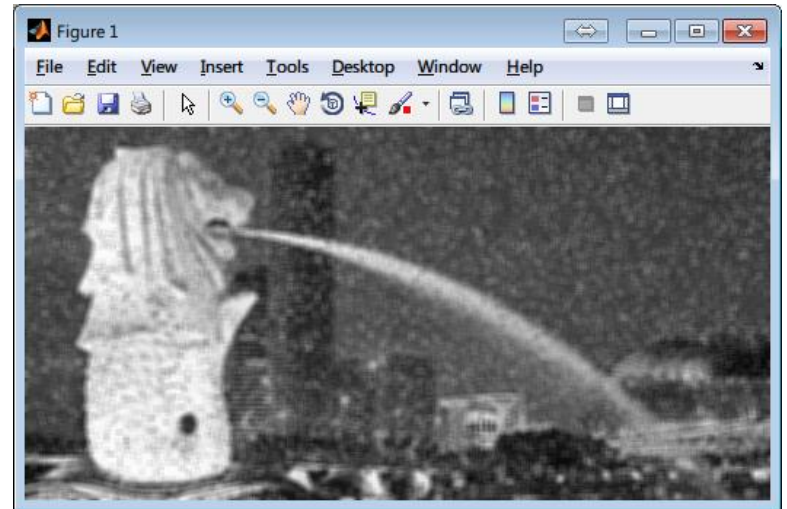


$=$

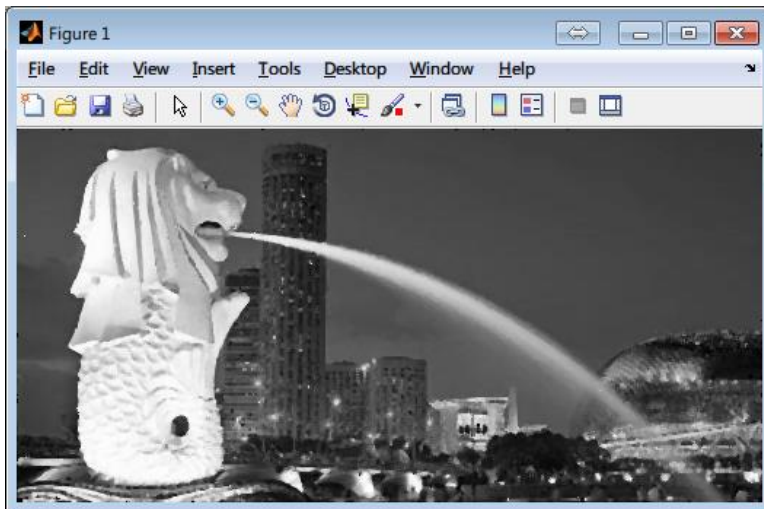


Denoising

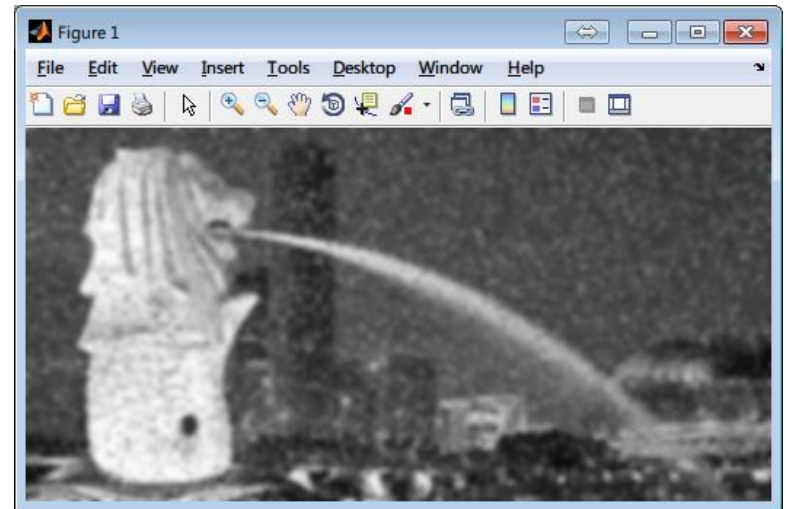
Mean



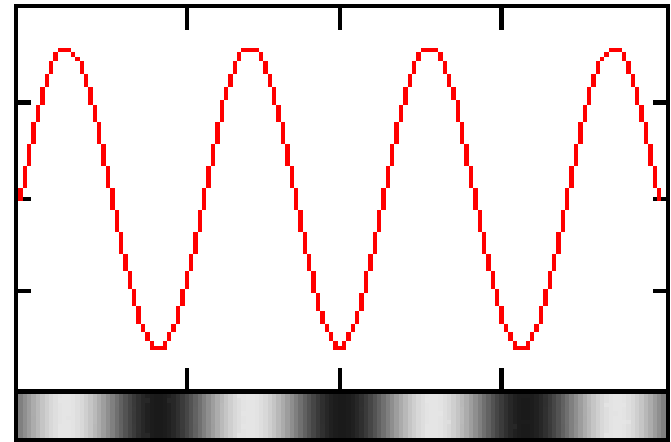
Median



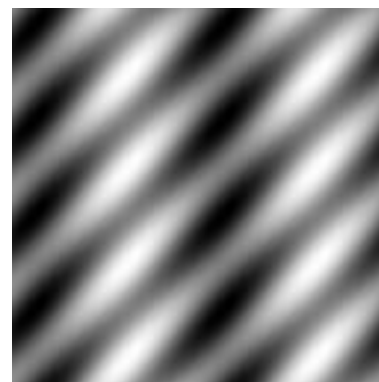
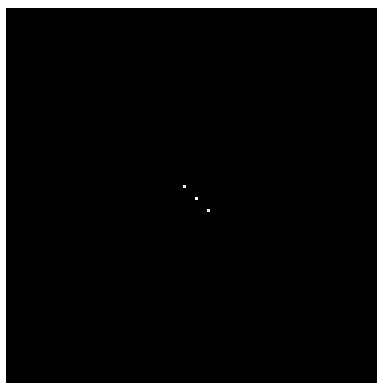
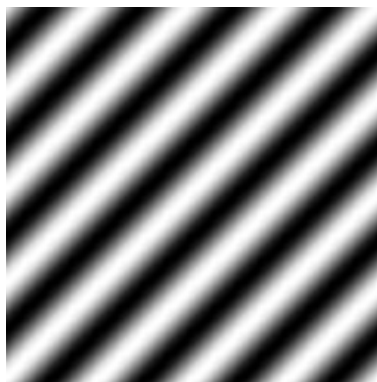
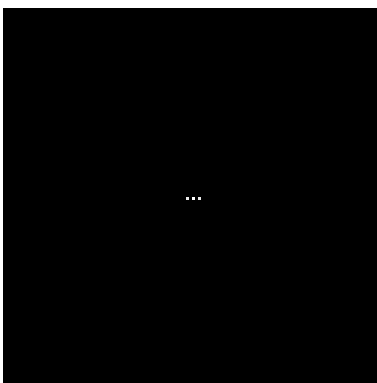
Gaussian



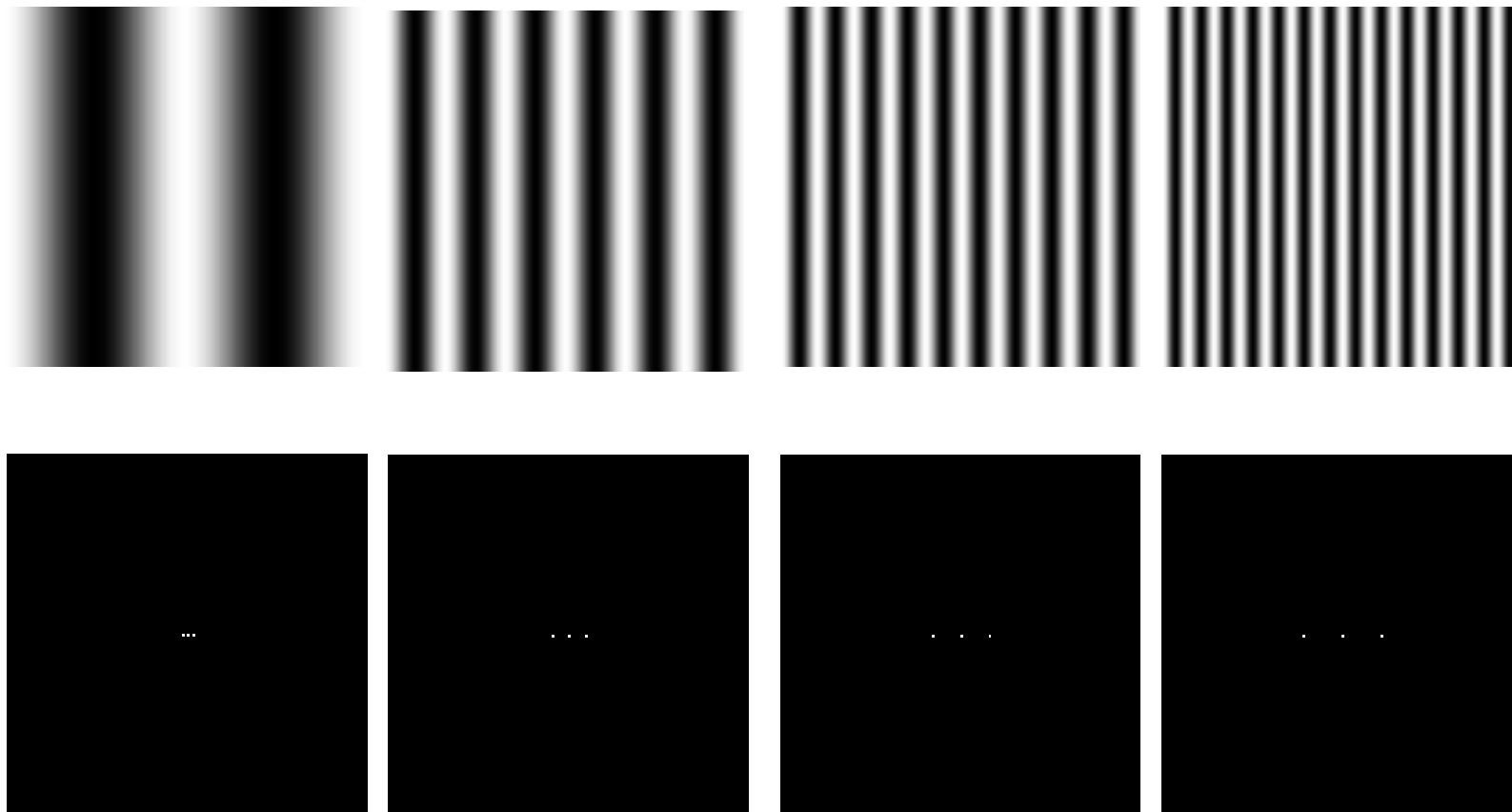
Images are Waves



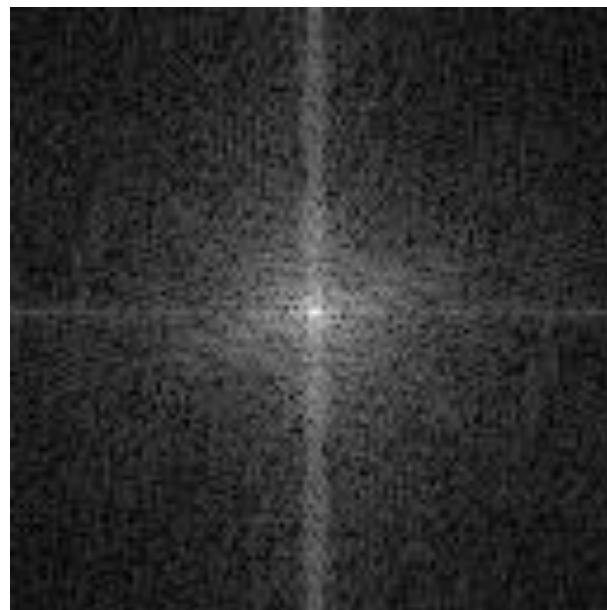
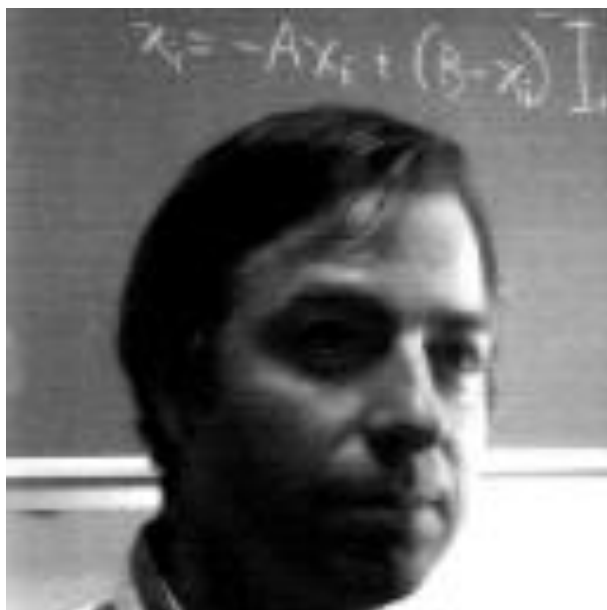
Fourier Transform on Images



Fourier Transform on Images

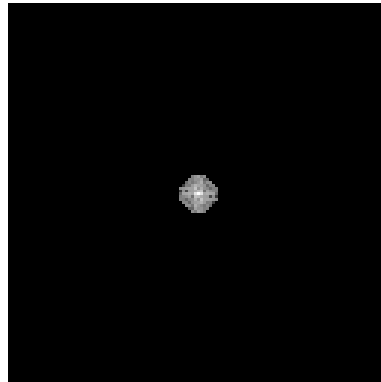


Fourier Transform on Images



Fourier Transform on Images

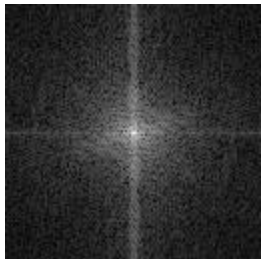
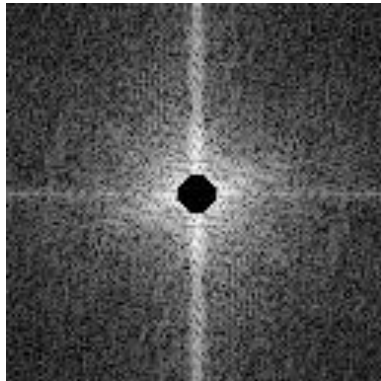
Low-pass
Filtered



Inverse
Transformed

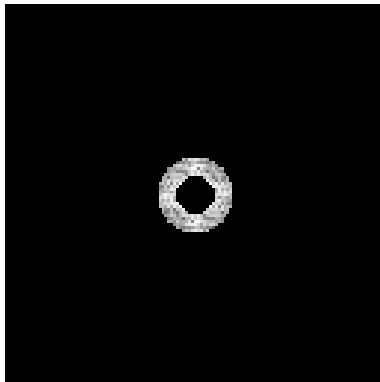


High-pass
Filtered

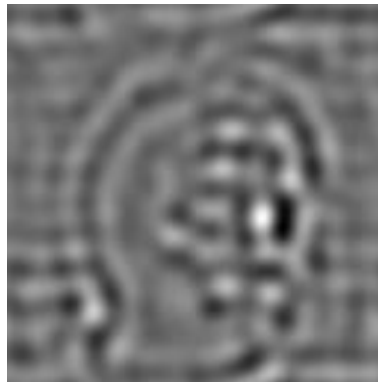


Fourier Transform on Images

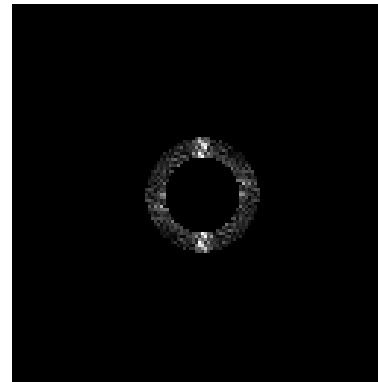
Band-pass
Filtered



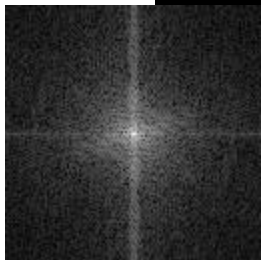
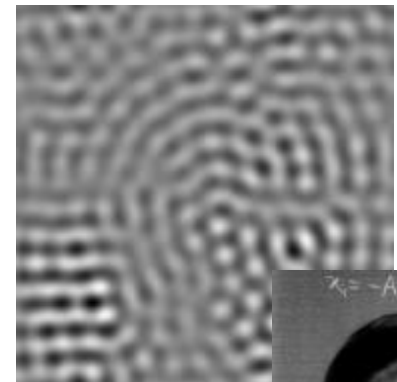
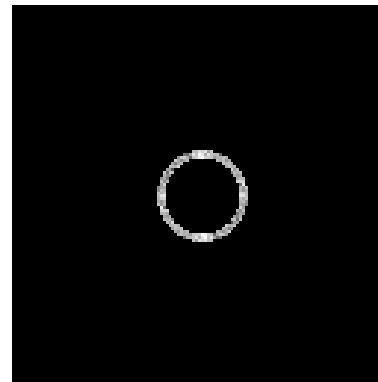
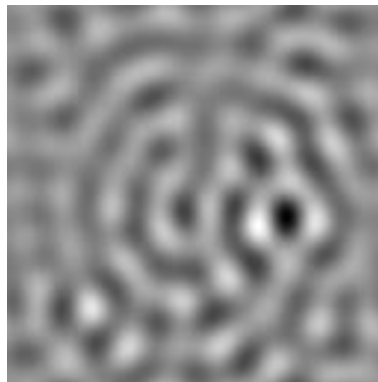
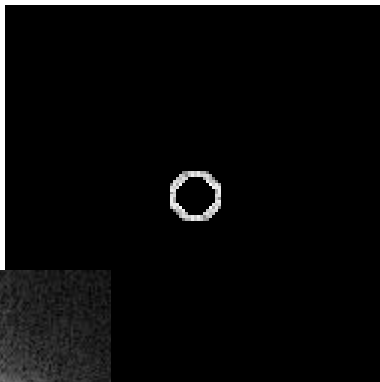
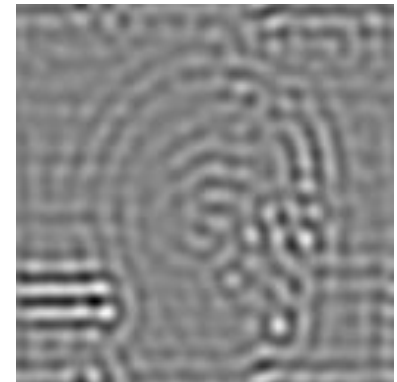
Inverse
Transformed



Band-pass
Filtered



Inverse
Transformed



Blending

```
>> w=repmat(0:1/(size(Apple,2)-1):1,[size(Apple,1), 1, 3]);  
>> I=(1-w).*Apple+w.*Orange;  
>> imshow(I)
```

