E E 1 7 1 *Signals & Systems* Fall Semester 2010

L A B 2

CONVOLUTION, EDGE DETECTION & DEMOS
Due: Tuesday Nov.4, 2010

You need to copy file ee171.mat. First go to www.cems.uvm.edu/∼mirchand/classes/EE171/Images
Download ee171.mat.zip
Get MATLAB running. Take ee171.mat into your MATLAB work directory. Then load ee171.mat into MATLAB by typing at the command line:

```
>>load ee171
```

Then try the following commands:

```
pwd
ls
who
whos
imagesc(einstein)
image{einstein}
colormap(gray)
```

1. **1-DIMENSION CONVOLUTION**

(a) Let signal f=[1 2 3 4], filter h1=[1 -1] and filter h2=[1 1]. Calculate the convolution of f and h1 and f and h2 yourself and verify your answer with that obtained by Matlab. In Matlab convolution is obtained by using the command "conv(f,h1)". You can type "help conv" to obtain more information about convolution. Also, commute the signal and the filters to see if the result changes.

Command list: f=[1 2 3 4]; h1=[1 -1]; h2=[1 1]; g11=conv(f,h1) g12=conv(f,h2) g21=conv(h2,f)

Note that appending ";" at the end of the command does not show you the results of the command. Without it, it does.

2. **2-DIMENSION CONVOLUTION**

(a) Let f=bird (bird is a variable in your workspace).

(b) Design four filters right in your workspace as follows:

$$h1 = \begin{bmatrix} 1 & -1 \\ 1 & -1 \end{bmatrix} \quad h2 = \begin{bmatrix} 1 & 1 \\ -1 & -1 \end{bmatrix} \quad h3 = \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \quad h4 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$$

(c) Use Matlab to convolve f and the four filters.

(d) Compare the original and convolved images.

(e) Note that to save a figure, you need to go to **File** on the top left hand side of the figure on display and in the drop down menu, use **Save As**. You can name it, save it in any of your directories and save it any of the file formats available (say pdf or jpeg).

(f) You can print all commands typed in your command window, since the start of your session, by using **File** and then **Print**.

Command list:

```
f=bird;
h1=[1 -1; 1 -1];
h2=[1 1; -1 -1];
h3=[1 -1; -1 1];
h4=[1 1; 1 1];
figure(1)                 %active window #1
imagesc(f);               %figure of original image
colormap(gray)            %gray scale.
image(f)                  %see which is better
figure(2)
g1=conv2(f,h1);           %convolve image f with filter h1
imagesc(g1)               see the result of convolution
colormap(gray)            %gray scale
figure(3)
g2=conv2(f,h2);           %convolve image f with filter h2
imagesc(g2)               see the result of convolution
colormap(gray)            %gray scale
figure(4)
g3=conv2(f,h3);           %convolve image f with filter h3
imagesc(g3)               see the result of convolution
colormap(gray)            %gray scale
figure(5)
g4=conv2(f,h4);           %convolve image f with filter h4
imagesc(g4)               see the result of convolution
colormap(gray)            %gray scale
```

3. **EDGE DETECTION**

You can do edge detection, using various edge detection algorithms, such as Canny, Prewitt, etc.

Command list:

```
>> I = imread('circuit.tif');
>>imshow(I);
>>BW1 = edge(I,'prewitt');
>>BW2 = edge(I,'canny');
>>figure; imshow(BW1);
>>figure, imshow(BW2)
>>I = imread('coins.png');
>>imshow(I)
>>BW1 = edge(I,'sobel');
>>BW2 = edge(I,'canny');
>>imshow(BW1)
>>figure, imshow(BW2)
```

3. SIGNAL PROCESSING, COMMUNICATION & CONTROL DEMOS

You can experiment with many interesting MATLAB Image Processing demos. Type

```
>>demos
```

In the Demo window, go to "Toolboxes". Now you can explore a whole range of applications. You might want to look at "Square waves from Sine Waves" (about 1/3rd of the way down the right side menu). That shows you how sine waves add to make a square wave, like in Fourier series.

Hand In:

1) Results of one 2-D convolution(copies of before and after).
2) Brief explanation of how one of the 2-D filters work. For example, how does convolving an image with a 2×2 filter give you some some particular (directional) edge in an image?