LAB 2
RE C I T AT I O N 3
CONVOLUTION, IMAGE PROCESSING & EDGE DETECTION

You need to copy file EE171.mat. First go to www.emba.uvm.edu/~mirchand/teaching
Then go to EE-171; then go to images; then copy ee171.mat
Take ee171.mat into your MATLAB work directory. Then load images into MATLAB by typing

    >>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 & 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.

Command list:

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

3. Image Processing Demos.

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

>> demos

In the Demo window, go to “Toolboxes”, and then to “Image Processing”. Then invoke the “Edge Detection” Demo and observe just a bit more complex edge detection.

Hand In:

1) Results of 1-D convolution.
2) Results of one 2-D convolution (copies of before and after).
3) Show your 1-D convolution calculated results for both the 2 filters with the given signal and show that the results are similar to your simulation results.
4) Brief explanation of how one of the 2-D filters work. For example, how does convolving an image with a $2 \times 2$ filter give you some some particular (directional) edge in an image?