

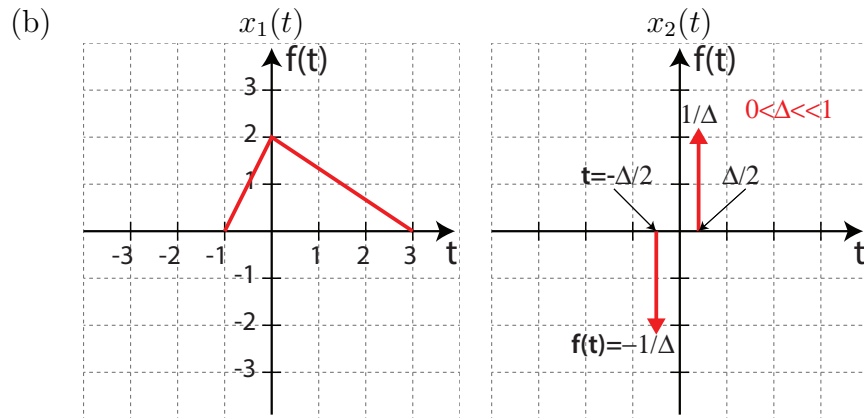
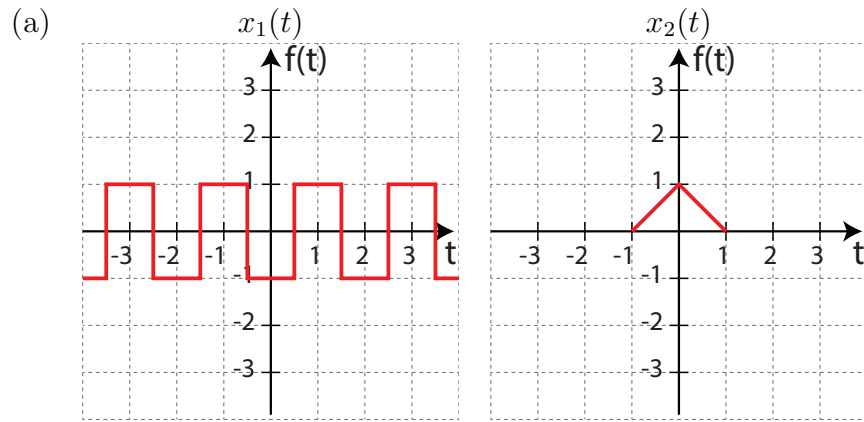
COMP 558: Assignment 1

Available: Thursday, January 24th, 2013

Due Date: Wednesday, February 6th, 2013, before midnight, via MyCourses.

Notes: You are required to use **MATLAB** which is currently installed on Unix and Windows machines in the CS labs. Your submitted solutions should reflect the work that you have put in and any experimental results should be based on your own implementations. If you have consulted a classmate it is best to mention this upfront. Your completed solutions should be submitted in electronic form (PDF only) via MyCourses.

Question 1 (30%) In each row, convolve the signal on the left with the one on the right. You should be able to perform these operations by hand (and not by using software). Provide a sketch of the resulting convolutions, showing the various steps. In each case be sure to label the significant time and amplitude values. Remember to reflect either $x_1(t)$ or $x_2(t)$ to get $x_1(\lambda - t)$ or $x_2(\lambda - t)$ before convolving the signals. (10% each)



c) Compute the convolution of the following two signals (10%):

$$x_1(t) = \begin{cases} 2 \sin(t + 5) & \text{if } t = -5 \dots 0 \\ 1 & \text{if } t = 1, 2 \\ 0 & \text{else} \end{cases} \quad (1)$$

$$x_2(t) = \delta(t) - \delta(t - 1) - \delta(t - 2) - \delta(t - 3). \quad (2)$$

Question 2 (10%) Assume that you were given an image and an identical but rotated copy of this image. What would you expect regarding their relative Fourier transform amplitude and phase spectra? For what computer vision problem(s) could this property be useful? **Whereas this claim is a bit open ended, if you could explain your answer mathematically by using the definition of a Fourier transform that would be helpful.**

Question 3 (60% + 10% Extra) The goal of this question is to consider convolution and frequency domain analysis of a 2D signal to achieve specific image processing goals. Consider the 'McGill.jpg' image given on <http://www.cim.mcgill.ca/~epiuze/?page=COMP558>.

a) **Spatial Blurring (5%)** Create a MATLAB function to generate a 2D isotropic Gaussian signal.

$$H(x, y) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (3)$$

```
function result = gaussianKernel(sig,wSize)
% sig: standard deviation sigma
% wSize: window size of the kernel
```

and smooth the input image I by convolving it with your Gaussian function for $\sigma = 0.5, 1, 2, 4$. Plot your figures into a grid by using MATLAB's *subplot* function. It might be easier to express the 2D isotropic Gaussian using polar coordinates which can be done by treating $x^2 + y^2$ as r^2 .

b) **Filtering in the Frequency Domain (5%)** Create a MATLAB function that removes high frequencies in the image by thresholding frequencies in its Fourier transform. Show and comment on your results for different frequency filtering windows. Explain the relationship between this filtering and what you did in a. Note that you can compute the Fourier transform $Y = F(X)$ and inverse Fourier transform $X = F^{-1}(Y)$ of a 2D image X using the MATLAB functions

```
Y = fft(X),
X = ifft(Y).
```

- c) **Edge strengthening (10%)** Pick one of the blurred images you obtained in a) or b). Convolve this image with the following two 3×3 kernels:

$$\mathbf{H}_1 = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix}, \quad \mathbf{H}_2 = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} \quad (4)$$

- Plot your results and explain the effect of each filter. (5%)
 - Based on these filters suggest a measure that computes edge strength and justify your suggestion. Implement this measure and provide a figure that supports your claim. (15%)
- d) **Edge direction (10%)** Suggest how we could compute the edge direction at each pixel (use the blurred input image you obtained in a) or b)) and implement a MATLAB script that displays the angle of the edge found at each pixel.
- e) **Edge map coloring (10%)** According to the edge map of the blurred input image computed in section c) and the edge direction map computed in section d), color edges as the following categorization:

- yellow: if $0 \leq \theta < 22.5$ or $157.5 \leq \theta < 180$
- green: if $22.5 \leq \theta < 67.5$
- blue: if $67.5 \leq \theta < 112.5$
- red: if $112.5 \leq \theta < 157.5$

where θ represents the edge direction angle.

- f) **Kirsch edge detector (10%)** Take the blurred input image from a) or b) and apply each of the following filters:

$$\mathbf{g}^{(1)} = \begin{bmatrix} +5 & +5 & +5 \\ -3 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix}, \quad \mathbf{g}^{(2)} = \begin{bmatrix} +5 & +5 & -3 \\ +5 & 0 & -3 \\ -3 & -3 & -3 \end{bmatrix}, \quad \mathbf{g}^{(3)} = \begin{bmatrix} +5 & -3 & -3 \\ +5 & 0 & -3 \\ +5 & -3 & -3 \end{bmatrix}, \quad \mathbf{g}^{(4)} = \begin{bmatrix} -3 & -3 & -3 \\ +5 & 0 & -3 \\ +5 & +5 & -3 \end{bmatrix}$$

$$\mathbf{g}^{(5)} = \begin{bmatrix} -3 & -3 & -3 \\ -3 & 0 & -3 \\ +5 & +5 & +5 \end{bmatrix}, \quad \mathbf{g}^{(6)} = \begin{bmatrix} -3 & -3 & -3 \\ -3 & 0 & +5 \\ -3 & +5 & +5 \end{bmatrix}, \quad \mathbf{g}^{(7)} = \begin{bmatrix} -3 & -3 & +5 \\ -3 & 0 & +5 \\ -3 & -3 & +5 \end{bmatrix}, \quad \mathbf{g}^{(8)} = \begin{bmatrix} -3 & +5 & +5 \\ -3 & 0 & +5 \\ -3 & -3 & -3 \end{bmatrix}$$

The edge magnitude of the Kirsch operator is calculated as the maximum magnitude across all those predetermined directions (N, NW, W, SW, S, SE, E, and NE). After applying this edge detector, color the edge map using (e) and comment on the result.

- g) **Posterization (10%)** Blur the original image and use any combination of the filters seen so far to darken its edges by tuning their weighting. You should obtain a nice posterized figure.