CS 736 : Assignment 1

Instructor : Suyash P. Awate

Due Date : 16 Feb 2014, Sunday, 11:59 pm

Please read the instructions for submission at :
http://www.cse.iitb.ac.in/%7Esuyash/cs736/submissionStyle.pdf

1. (20 points) X-Ray Computed Tomography : Radon Transform.

   Generate an image of the Shepp-Logan phantom using the Matlab function call "phantom (128)". Call this $f(x, y)$. Associate (logically) a coordinate frame with the image with the origin located at the center pixel.

   (a) (6 points) Given an image, implement a function (named myIntegration()) to perform integration of the image intensities along lines $L_{t,\theta}$ that are parametrized by t and θ, where $t \in (-\infty, \infty)$ is the (signed) distance of the line from the origin and $\theta \in [0, 180)$ is the angle (in degrees) made by the upward-pointing normal to the line with the $+X$ axis. You may use the "interp2()" function in Matlab. Choose a suitable value for the step size Δs for the discrete/summation approximation of the integral. Justify your choice. Choose a suitable image-interpolation scheme. Justify your choice.

   (b) (6 points) Use the function you implemented in the previous question to implement another function (named myRadonTrans()) that computes the Radon transform $Rf(t, \theta)$ for the discrete set of values $t = -90, -89, \cdots, 89, 90$ ($\Delta t = 1$ pixel-width unit) and the discrete set of values $\theta = 0, 3, 6, \cdots, 177$ ($\Delta\theta = 3$ degrees).

   (c) (5 points) Compare Radon transforms computed with different choices of parameters (during the integration). Specifically, compute and show the Radon-transform images with the following 3 Δs values: $\Delta s = 0.5$ pixel-width unit, $\Delta s = 1$ pixel-width unit, $\Delta s = 3$ pixel-width units. Show the 1D function plots for the Radon-transform image values for the following values θ : 0 degree, 90 degrees. Which 1D plot appears the smoothest/roughest and why ? Which image appears the smoothest/roughest and why ?

   (d) (3 points) If you were to design a CT scanner, what parameter settings (for $\Delta t, \Delta\theta$) would you use or *not* use ? Justify your choice (using suitable images / plots) and the underlying trade-offs.

2. (20 points) X-Ray Computed Tomography : Filtered Backprojection.

   Generate an image of the Shepp-Logan phantom using the Matlab function call "phantom (256)". Call this $f(x, y)$. Associate (logically) a coordinate frame with the image with the origin located at the center pixel.

(a) (8 points) Use the Matlab function “radon()” to compute a Radon transform of $f(x,y)$ with the values of $\theta = 0, 3, 6, \cdots, 177$ degrees. Use the Matlab function “iradon()”, with suitable parameters, to compute the backprojection of the Radon transform. Now, implement the Ram-Lak filter, Shepp-Logan filter, and the Cosine filter (in the Fourier domain; use Matlab functions “fft()” and “ifft()”), where L is a user-controlled parameter. Name the function as myFilter(). Use your function to filter (in all 3 ways) the Radon-transform data for 2 different values of L, i.e., $L = w_{\text{max}}$ and $L = w_{\text{max}}/2$, where w_{max} is the highest frequency in the (discrete) Fourier representation. Use the “iradon()” function to compute the backprojection of the filtered data. Show the resulting images and justify the similarities and the differences observed between the different combinations of filters and parameter values.

(b) (8 points) Generate a blurred version of the Shepp-Logan image by convolving it with a Gaussian. Use the Matlab function calls “mask = fspecial ('gaussian', 11, 1)” to create a convolution mask and then use “conv2 (image, mask, 'same')” for the smoothing. Generate another blurred image by using “mask = fspecial ('gaussian', 51, 5)”. Show the 3 different versions of the Shepp-Logan images, say S_0, S_1, S_5. For all these, compute the radon transform with $\theta = 0, 3, 6, \cdots, 177$ degrees, apply the Ram-Lak filter with $L = w_{\text{max}}$, and compute the backprojection. For all 3 filtered backprojections, say, R_0, R_1, R_5, compute the relative root-mean-squared errors (RRMSE) RRMSE(S_i, R_i), where the RRMSE for 2 images A and B is defined as :

RRMSE$(A,B) = \sqrt{\sum_p (A(p) - B(p))^2} / \sqrt{\sum_p A(p)^2}$, where the summation is over all pixels p. In which of the 3 cases is the RRMSE the highest and the lowest ? Explain theoretically.

(c) (4 points) For each of the 3 examples (i.e., Shepp-Logan phantoms S_0, S_1, S_5 and their filtered backprojections R_0, R_1, R_5 using the Ram-Lak filter), plot the RRMSE values as a function of L with $L = 1, 2, \cdots, w_{\text{max}}$. Explain your findings.

3. (10 points) Optimization using Gradient Descent.

Find the optimal values of $x, y \in \mathbb{R}$ to minimize the following function:

$f(x,y) = (1-x)^2 + 100(y - x^2)^2$.

Use gradient-descent optimization with the initial solution as $x = (-3), y = (-4)$.

In the gradient-descent optimization, ensure that every step that is taken decreases $f(x,y)$.

(a) (3 points) Use gradient descent with a fixed/static step size τ. Choose a value for the (fixed) stepsize τ that leads to a solution (x^*, y^*) where $f(x^*, y^*) < 0.0001$ in the least number of iterations. Plot the sequences of $x(n)$, $y(n)$, and $\log f(x(n), y(n))$ values over iteration n.

(b) (7 points) Use gradient descent with a dynamic/adaptive step size $\tau(n)$ as a function of iteration n. Choose a suitable value for the initial stepsize $\tau(0)$. Obtain a solution (x^*, y^*) where $f(x^*, y^*) < 0.0001$. Plot the sequences of $x(n)$, $y(n)$, and $\log f(x(n), y(n))$ values over changing iteration n. Which of the two versions of gradient descent converges to the solution faster, and why ? How critical is the choice of the (initial) stepsize for the two methods to successfully reach the solution, and why ?

4. (20 points) Diffusion Tensor Imaging.

Consider a diffusion-MRI experiment (in 2D) that performs diffusion imaging for a chosen set of N gradient directions $\{g_i\}_{i=1}^N$ and provides the values $\{S(g_i)\}_{i=1}^N$ corresponding to each direction.

The direction vectors are:

$\{g_i\}_{i=1}^6 = \{[1,0],[0.866,0.5],[0.5,0.866],[0,1],[-0.5,0.866],[-0.866,0.5]\}.$

For a particular pixel in the image, the acquired data for each direction vector (in the same sequence as above) are:

$\{S(g_i)\}_{i=1}^6 = \{0.5045, 0.6874, 0.3632, 0.3483, 0.2606, 0.2407\}.$

Use a diffusion-tensor model that represents diffusion using a 2×2 symmetric positive-definite matrix D.

Assume $S_0 = 1$ and $b = 0.1$.

(a) (14 points) Given this data, estimate D using gradient-descent optimization, report D, and plot the sequences of the logarithm of the objective function and the 4 entries in D over iteration.

(b) (3 points) Report the (principal) direction (unit vector) along which the diffusion in the 2D plane is the strongest.

(c) (3 points) How much more (by what multiplicative factor) is the diffusion in the principal direction as compared to the diffusion in the direction orthogonal to it ?