E1 251—Linear and nonlinear optimization: Aug-Dec'2024
Assignment II

Let X be an $n \times n$ image, and let $\mathcal{S}_{\mathbf{r}_i}$ denote the operator that picks pixel from location $\mathbf{r}_i$. Suppose we have pixel values $\{m_i, i = 1, \ldots, L\}$ of the image from randomly selected points $\{\mathbf{r}_i, i = 1, \ldots, L\}$ such that $\mathcal{S}_{\mathbf{r}_i} X = m_i$. We want to find an estimate of X by solving the following minimization problem: $f(Y) = \sum_{i=1}^{L} (\mathcal{S}_{\mathbf{r}_i} X - m_i)^2 + \lambda(\|\mathcal{D}_{xx} Y\|_2^2 + \|\mathcal{D}_{yy} Y\|_2^2 + 2\|\mathcal{D}_{xy} Y\|_2^2)$, where $\mathcal{D}_{xx}$, $\mathcal{D}_{yy}$, and $\mathcal{D}_{xy}$ denote operators that computes derivatives $\frac{\partial^2}{\partial x^2}$, $\frac{\partial^2}{\partial y^2}$, $\frac{\partial^2}{\partial x \partial y}$, and where $\| \cdots \|_2^2$ is overloaded to denote the sum of the square of the elements if it takes the 2D array as the argument. Note that the list $\{\mathbf{r}_i, i = 1, \ldots, L\}$ can be compactly represented by a binary image M that has ones in the pixel locations $\{\mathbf{r}_i, i = 1, \ldots, L\}$ and zeros in other pixel locations. Similarly, the list $\{m_i, i = 1, \ldots, L\}$ can be compactly represented by and image G containing the pixels values $\{m_i, i = 1, \ldots, L\}$ at locations $\{\mathbf{r}_i, i = 1, \ldots, L\}$, and zeros in other locations. Note that $M \oplus X = G$ where $\oplus$ denotes element-wise multiplication of arrays. There exist an operator $\mathcal{Q} : \mathbb{R}^{n \times n} \rightarrow: \mathbb{R}^{n \times n}$ such that the gradient of $f(Y)$ can be expressed as $\mathcal{Q}Y - G$. You will be supplied with a MATLAB program that takes M, λ, and Y as inputs and returns $\mathcal{Q}Y$. For this exercise, obtain M by the following MATLAB command: $M = \text{rand}(n) < 0.6$.

1. Write a CG algorithm that computes the minimum of $f(Y)$. Write the algorithm such that all the variables are kept in 2D array form. Write in such a way that the iteration is terminated based the condition $\|\nabla f(Y)\|/\|G\| < \epsilon$ where ϵ is the tolerance that you have to choose (typically in the range 0.01 to 0.000001).

2. In practice, you will only have M and G, and the minimum of $f(Y)$, say $\hat{X}$, will be the required estimate of X. However, to test the method, you can choose a model image for X, simulate G, and obtain the estimate $\hat{X}$ by running the algorithm. For a chosen model image, X, and λ, plot $\|X - \hat{X}\|_2$ for various values of ϵ.

3. For a chosen model image, X, and M, and a properly chosen value for ϵ, plot $\|X - \hat{X}\|_2$ for various values of λ.

4. Repeat (1) and (2) for at least five model images, and choose the best λ and obtain the corresponding estimates. Display the estimates along with model images side-by-side.