1. Using several images of your choice, evaluate the results of gradient magnitudes for edge detection using Prewitt and Sobel filters, as well as the simple $[-1, 1]$ and $[-1, 1]^2$ filters described in class.

2. Another way to find edges is to identify areas in the image that have large variance (or standard deviation). Write a function that receives a gray-level image $I$ and an integer $n$ and builds and returns a gray level image $S$ where $S[i][j]$ contains the standard deviation of the $n \times n$ region surrounding pixel $I[i][j]$.

Recall that the standard deviation $\sigma$ and variance $\text{var}$ of a set of observations $X = x_1, \ldots, x_n$ are given by:

$$\text{var}(X) = \frac{\sum_{i=1}^{n} x_i^2}{n} - \left( \frac{\sum_{i=1}^{n} x_i}{n} \right)^2$$

$$\sigma(X) = \sqrt{\text{var}(X)}$$

3. The straightforward implementation of the algorithm to find the standard deviation image takes $O(rcn^2)$ for an image with $r$ rows, $c$ columns, and regions of size $n \times n$. It is possible to improve the running time to $O(rc)$, that is, the standard deviation in every region can be computed in constant time. Can you figure out how? Implement this improved function. (The instructor will provide hints).