Project 2 Posted 9/17/2024

Question 1: Implement the Averaging and Median filters.

In a first step, write a function that performs a convolution on a given image and filter. Both inputs and the output are of the Matrix type, which means that you first have to convert the input image into a matrix. This seems cumbersome but is more efficient in the long run, because sometimes we want to perform multiple successive convolutions on an image, and we would also like to be able to use real-valued convolution inputs. The function should thus have the following signature:

```
Matrix smoothing_filter(Matrix m1, Matrix m2)
```

As you know, `m1` and `m2` are in principle exchangeable, but only if they are padded with zeroes outside of their borders so that all overlaps of elements from `m1` and `m2` are considered. For our computer vision applications, however, we expect the filter to be smaller than the input image, and during convolution we never want any elements of the filter to be outside of the image. Therefore, we require that `m1` contains the values of the input image, and `m2` those of the filter. The resulting matrix is always of the same size as the input image, with zeroes in those positions that could not be reached by the center element of the filter. For example, if the input image has 7 by 7 entries and the filter has 3 by 3 entries, then the output is a 7 by 7 matrix with zeroes in its first and last rows and its first and last columns. The function assumes that we use the center of the filter as the indicator of where to store the output value. If the height or width of the filter is an even number of elements, the center value is rounded down. For example, if `m2` is a 2 by 2 filter, we use its upper-left position as its "anchor" for storing output values.

Having implemented the `smoothing` filter, implement a `median` filter in the function `median_filter(Matrix m1, Matrix m2),` the filter has 5 by 5 entries. The function takes an input image and outputs a filtered image of the same size.

Include all operations as discussed in class, including smoothing filter (averaging), and median filter. Again, the output should be scaled to use of the full range of intensity values from 0 to 255.

Submit your code and include the greyscale image that you choose in Canvas.