Brandon Shumin
ECE 6310
Lab 1

2D Convolution (Method 1):

The main code for this method was simply a modification of the provided time-smooth.c code with an expanded convolution area. In the time-smooth.c code the area of convolution is 3x3 while in Method 1, the area is 7x7. This was simply accomplished by altering the for loops that control the pixels being averaged at each step of the smoothing process to expand the area being averaged to a 7x7 area.

This process ran (after averaging 10 runs) in around 0.021244s or 21.244ms.

Separable Filters (Method 2):

The separable filters method, Method 2, was implemented essentially by taking the summation the columns of the 7x7 area to be convolved and then taking the sum of the summed columns and averaging them. This process is then repeated for each pixel in the image that is not an edge case. The averaging was done as one step at the end rather than after each summation step due to integral truncation creating unreliable values when compared to the Method 1 produced image using the "diff" command.

This process ran (after averaging 10 runs) in around 0.006176s or 6.176ms.

Sliding Window with Separable Filters (Method 3):

The sliding window method, Method 3, is essentially the previous method with the addition of a process that expedites the mathematics so that the summation of all relevant pixels around a convolved pixel does not need to occur every time. The first time on a row that a convolution must be performed, the same process as above occurs, but then on subsequent operations on that row, the previous column average is removed and the new column average, which conceptually would be the right side of the matrix, is added on, meaning that computational power is saved not having to add as many pixels.

This process ran (after averaging 10 runs) in around 0.003908s or 3.908ms.

Summary:

So, as is obvious from the above-mentioned run-times, 2D convolution is the slowest, separable filters is the middle speed, and the sliding window with separable filters is the fastest. The 2D convolution is the slowest since it has a complexity of $O(n^4)$ using Big-O notation. This means that to perform this method, 4 nested for-loops are used and this is computationally slow compared to the other methods, which is reflected in its slower run-time. Separable filters are faster than 2D convolution since it is only complexity of about $O(2n^3)$ since there are 2 triple-nested for loops in the method. Sliding window with separable filters is faster than just separable filters because it can short-circuit some of the operations required for method 2 to function. So the first time on a row, normal separable filter convolution is performed, but after this first operation is performed, on subsequent operations on this row the previous left-most column average is removed and the right-most column average is added since the operation is equivalent to redoing the whole convolution process on the next pixel in a row. This saves time as the nested for-loop required for performing this convolution is now skipped on all but the first pixel of each row of the image.