CS-663 Assignment 5 Q2

Soham Naha (193079003)
Akshay Bajpai (193079002)
Mohit Agarwala (19307R004)

November 15, 2020

Question2

Consider a 1D image (for example, a single row from a 2D image). You know that given such an image, computing its gradients is trivial. An inquisitive student frames this as a convolution problem to yield $g = h * f$ where g is the gradient image (in 1D), h is the convolution kernel to represent the gradient operation, and f is the original 1D image. The student tries to develop a method to determine f given g and h. What are the fundamental difficulties he/she will face in this task? Justify your answer. You may assume appropriate boundary conditions. Now consider that you are given the gradients of a 2D image in the X and Y directions, and you wish to determine the original image. What are the difficulties you will face in this task? Justify your answer. Again, you may assume appropriate boundary conditions. $[10 + 10 = 20 points]$

Solution :

For 1D

For a 1D image, we can assume the gradient kernel to be $h = [-1 \ \ 0 \ \ 1]$.
Since we have $g = h \star f$, we have $G = HF$ after taking their Discrete Fourier Transform. Hence this would imply $f = F^{-1}(G/H)$. Let's try to compute the DFTs of these signals to understand the issue with this operation.

Let's assume the length of our image is K. Hence we would need to zero-pad our filter $K - 1$ times before finding its DFT. Hence we would have to take an N-point DFT with $N = K - 1 + 3 = K + 2$

$$\begin{aligned} H(k) &= \sum_n h[n] e^{-j\frac{2\pi}{N}kn} \\ &= -1 + e^{-j\frac{4\pi}{N}k}, k \in \{0, 1, 2 \ldots N-1\} \end{aligned}$$

Now for $k = 0$, we would have $H(k) = 0$. (In other words, a gradient operation removes all DC components from a signal). Also for large values of K, $H(k)$ will be close to 1 (in the complex domain).
Hence we won't be able to recover the DC components in the signal and also have a hard time uncovering the larger frequencies if the signal is very long and has high frequency components.

For 2D

Similar to 1D case, we will calculate the DFT of the 2-D kernel.
We use the kernel $h_x = \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix}$ for derivatives along the X-axis and $h_y = \begin{pmatrix} -1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{pmatrix}$ for derivatives along the Y-axis. Assuming we need to take a N_1, N_2-DFT after zero padding we

obtain,

$$
\begin{aligned}
H_x(k_1,k_2) &= \sum_x \sum_y h_x[x,y] e^{-j\frac{2\pi}{N_1}k_1 x} e^{-j\frac{2\pi}{N_2}k_2 y} \\
&= (-1+e^{-j\frac{4\pi}{N_1}k_1})(1+2e^{-j\frac{2\pi}{N_2}k_2}+e^{-j\frac{4\pi}{N_2}k_2}), k_1 \in \{0,1,2 \ldots N_1-1\}, k_2 \in \{0,1,2 \ldots N_2-1\} \\
H_y(k_1,k_2) &= \sum_x \sum_y h_y[x,y] e^{-j\frac{2\pi}{N_1}k_1 x} e^{-j\frac{2\pi}{N_2}k_2 y} \\
&= (1-e^{-j\frac{4\pi}{N_2}k_2})(1+2e^{-j\frac{2\pi}{N_1}k_1}+e^{-j\frac{4\pi}{N_1}k_1}), k_1 \in \{0,1,2 \ldots N_1-1\}, k_2 \in \{0,1,2 \ldots N_2-1\}
\end{aligned}
$$

Clearly, for the DC case ($k_1 = k_2 = 0$) we obtain $H_x = H_y = 0$ and we won't be able to recover the DC components correctly. Again, for large widths (for X-derivatives) and large heights (for Y-derivatives), $k_1 \to N_1$ and $k_2 \to N_2$, we won't get solutions.