Ashutosh Sharma (190100027)
Kritti Sharma (180100060)
Shreyas Chandgothia (180260037)

Homework Assignment 2
CS 663: Digital Image Processing

2021-08-31

Exercise-1

Consider a 1D image f of size $(1 \times n)$ given as $(f_0, f_1, f_2, ...f_{n-1})$ and a 1D convolution mask w of size (1×7) given as $(w_0, w_1, w_2, w_3, w_4, w_4, w_6)$. In order to compute the convolution of f with w, we first need to pad f with 6 zeros on both sides. Therefore, now we have f' given by $(0, 0, 0, 0, 0, 0, f_0, f_1, ...f_{n-1}, 0, 0, 0, 0, 0, 0)$. We also need to invert the convolution mask w' which is given by $(w_6, w_5, w_4, w_3, w_2, w_1, w_0)$. Let c', which denotes the padded result from this convolution, be given as $(c_{-3}, c_{-2}, c_{-1}, c_0, c_1, c_2...., c_{n-1}, c_n, c_{n+1}, c_{n+2})$. Next, to compute the convolution, we need to slide w' over f', take element-wise product and take sum.
For example, c_{-3} is computed as follows:

$$\begin{array}{ccccccccccccccccccccccc}
w_6 & w_5 & w_4 & w_3 & w_2 & w_1 & w_0 & & & & & & & & & & & & & & & & \\
\times & \times & \times & \times & \times & \times & \times & & & & & & & & & & & & & & & & \\
0 & 0 & 0 & 0 & 0 & 0 & f_0 & f_1 & f_2 & . & . & . & . & f_{n-4} & f_{n-3} & f_{n-2} & f_{n-1} & 0 & 0 & 0 & 0 & 0 & 0
\end{array}$$

$$c_{-3} = w_0 f_0$$

To compute c_{-2}, we slide the convolution mask with a stride of 1 to the right, take the element-wise product and compute the sum as follows:

$$\begin{array}{ccccccccccccccccccccccc}
 & w_6 & w_5 & w_4 & w_3 & w_2 & w_1 & w_0 & & & & & & & & & & & & & & & \\
 & \times & \times & \times & \times & \times & \times & \times & & & & & & & & & & & & & & & \\
0 & 0 & 0 & 0 & 0 & 0 & f_0 & f_1 & f_2 & . & . & . & . & f_{n-4} & f_{n-3} & f_{n-2} & f_{n-1} & 0 & 0 & 0 & 0 & 0 & 0
\end{array}$$

$$c_{-2} = w_1 f_0 + w_0 f_1$$

This way, we continue to compute $c_{-1}, c_0, c_1, c_2...$ until the last term:

$$\begin{array}{ccccccccccccccccccccccc}
 & & & & & & & & & & & & & & & & w_6 & w_5 & w_4 & w_3 & w_2 & w_1 & w_0 \\
 & & & & & & & & & & & & & & & & \times & \times & \times & \times & \times & \times & \times \\
0 & 0 & 0 & 0 & 0 & 0 & f_0 & f_1 & f_2 & . & . & . & . & f_{n-4} & f_{n-3} & f_{n-2} & f_{n-1} & 0 & 0 & 0 & 0 & 0 & 0
\end{array}$$

$$c_{n+2} = w_6 f_{n-1}$$

Finally, we crop out the first and the last three terms, i.e. $(c_{-3}, c_{-2}, c_{-1}, c_n, c_{n+1}, c_{n+2})$ to obtain the unpadded result of the convolution c. This is again a 1D image of size $(1 \times n)$ having elements $(c_0, c_1, c_2, ...c_{n-1})$

This entire computation can be written in a compact form as a matrix multiplication as follows:

$$\begin{bmatrix}
w_6 & w_5 & w_4 & w_3 & w_2 & w_1 & w_0 & 0 & 0 & ..... & 0 & 0 & 0 & 0 \\
0 & w_6 & w_5 & w_4 & w_3 & w_2 & w_1 & w_0 & 0 & ..... & 0 & 0 & 0 & 0 \\
0 & 0 & w_6 & w_5 & w_4 & w_3 & w_2 & w_1 & w_0 & ..... & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & w_6 & w_5 & w_4 & w_3 & w_2 & w_1 & ..... & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & w_6 & w_5 & w_4 & w_3 & w_2 & ..... & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & w_6 & w_5 & w_4 & w_3 & ..... & 0 & 0 & 0 & 0 \\
. & . & . & . & . & . & . & . & . & ..... & . & . & . & . \\
. & . & . & . & . & . & . & . & . & ..... & . & . & . & . \\
. & . & . & . & . & . & . & . & . & ..... & . & . & . & . \\
. & . & . & . & . & . & . & . & . & ..... & . & . & . & . \\
. & . & . & . & . & . & . & . & . & ..... & . & . & . & . \\
. & . & . & . & . & . & . & . & . & ..... & . & . & . & . \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & ..... & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & ..... & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & ..... & w_0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & ..... & w_1 & w_0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & ..... & w_2 & w_1 & w_0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & ..... & w_3 & w_2 & w_1 & w_0
\end{bmatrix}
\begin{bmatrix}
0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ f_0 \\ f_1 \\ . \\ . \\ . \\ . \\ f_{n-2} \\ f_{n-1} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0
\end{bmatrix}$$

The product of this $((n+6) \times (n+12))$ matrix with the $((n+12) \times 1)$ image vector will give an $((n+6) \times 1)$ output image vector, which will be exactly same as the padded convolution output c' described above. We can chop off the first and the last three elements to get the convolved image with the same size as that of the input image.

In order to get rid of padding, we can simply write the final unpadded convolved image c in terms of the following matrix product, by deleting the first and the last three rows, along with the first and the last six columns from the previous matrix:

$$\begin{bmatrix} w_3 & w_2 & w_1 & w_0 & 0 & 0 & 0 & 0 & 0 & ..... & 0 & 0 & 0 & 0 \\ w_4 & w_3 & w_2 & w_1 & w_0 & 0 & 0 & 0 & 0 & ..... & 0 & 0 & 0 & 0 \\ w_5 & w_4 & w_3 & w_2 & w_1 & w_0 & 0 & 0 & 0 & ..... & 0 & 0 & 0 & 0 \\ w_6 & w_5 & w_4 & w_3 & w_2 & w_1 & w_0 & 0 & 0 & ..... & 0 & 0 & 0 & 0 \\ 0 & w_6 & w_5 & w_4 & w_3 & w_2 & w_1 & w_0 & 0 & ..... & 0 & 0 & 0 & 0 \\ 0 & 0 & w_6 & w_5 & w_4 & w_3 & w_2 & w_1 & w_0 & ..... & 0 & 0 & 0 & 0 \\ . & . & . & . & . & . & . & . & . & ..... & . & . & . & . \\ . & . & . & . & . & . & . & . & . & ..... & . & . & . & . \\ . & . & . & . & . & . & . & . & . & ..... & . & . & . & . \\ . & . & . & . & . & . & . & . & . & ..... & . & . & . & . \\ . & . & . & . & . & . & . & . & . & ..... & . & . & . & . \\ . & . & . & . & . & . & . & . & . & ..... & . & . & . & . \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & ..... & w_1 & w_0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & ..... & w_2 & w_1 & w_0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & ..... & w_3 & w_2 & w_1 & w_0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & ..... & w_4 & w_3 & w_2 & w_1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & ..... & w_5 & w_4 & w_3 & w_2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & ..... & w_6 & w_5 & w_4 & w_3 \end{bmatrix} \begin{bmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ . \\ . \\ . \\ . \\ . \\ . \\ f_{n-6} \\ f_{n-5} \\ f_{n-4} \\ f_{n-3} \\ f_{n-2} \\ f_{n-1} \end{bmatrix}$$

Note that this is the product of a $(n \times n)$ matrix with a $(n \times 1)$ image vector. Thus the output will give a $(n \times 1)$ image vector, exactly as desired.

Looking at the matrix, we note that this is a ***Toeplitz*** and ***Heptadiagonal Matrix***. A ***Toeplitz Matrix***, also known as a *Diagonal-constant Matrix* is a matrix in which each descending diagonal from left to right has constant enrties. A ***Heptadiagonal Matrix*** is a matrix that is nearly diagonal; to be exact, it is a matrix in which the only nonzero entries are on the main diagonal, and the first three diagonals above and below it. Such matrices have important properties in terms of storing them in memory and performing computation on them, a few of which are as follows:

- A Heptadiagonal matrix is a sparse matrix, therefore it can be stored more efficiently than a general matrix by using a special storage scheme.
- The determinant of a Heptadiagonal matrix can be computed from a seven-term recurrence relation.
- Two Toeplitz matrices of order n may be added in $\mathcal{O}(n)$ time (by storing only one value of each diagonal) and multiplied in $\mathcal{O}(n^2)$ time
- Toeplitz matrices are closely connected with Fourier series, because the multiplication operator by a trigonometric polynomial, compressed to a finite-dimensional space, can be represented by such a matrix (similar to how a linear convolution is represented as multiplication by a Toeplitz matrix here)
- Toeplitz matrices commute asymptotically. This means they diagonalize in the same basis when the row and column dimension tends to infinity.

There are many potential application of such a matrix-based construction to represent a convolution (or any theoretically continuous operation). This is because matrices are discrete-natured entities which can be stored very efficiently in computers. Performing computations and algorithms through matrix-based calculations in computers is a widely-researched area and there exist numerous efficient techniques to do so. Also, such a matrix-based construction can be formulated for other common operations in the domain of image processing, such as autocorrelation, cross-correlation, moving average, trigonometric multiplicatio, etc.