|  |  |  |  |
| --- | --- | --- | --- |
| **Course Name:** | **Digital Signal & Image Processing Laboratory** | **Semester:** | **VI** |
| **Date of Performance:** | **22 /01 / 2025** | **Batch No:** | **B - 2** |
| **Faculty Name:** | **Dr. Om Goswami** | **Roll No:** | **16014022050** |
| **Faculty Sign & Date:** |  | **Grade/Marks:** | **\_\_\_ / 20** |

**Experiment No: 2**

**Title:** Implementation of the **Convolution** and **Correlation**

|  |
| --- |
| **Objective:**  To understand and implement the concepts of **convolution** and **correlation** in digital signal processing and analyze their results. |

|  |
| --- |
| **COs to be achieved:** |
| **CO1:** Identify various discrete time signals and systems and perform signal manipulation |

|  |
| --- |
| **Materials Required:** MATLAB software  **Books/ Journals/ Websites referred:**   1. Nagoor Kani “Digital Signal Processing”, 2nd Edition, TMH Education. 2. Alan V. Oppenheim and Ronald W. Schafer, "Discrete-Time Signal Processing." 3. MATLAB Documentation: <https://www.mathworks.com/help/matlab/> |

|  |
| --- |
| Theory:Convolution: Convolution is a mathematical operation used to determine the output of a linear time-invariant (LTI) system given its input signal and impulse response. For discrete-time signals, the convolution of two signals and is given by: In MATLAB, convolution can be performed using the conv() function.Correlation: Correlation measures the similarity between two signals as a function of the time-lag applied to one of them. For discrete signals and, the cross-correlation is given by:  For auto-correlation, is correlated with itself.  **In MATLAB, correlation can be performed using the xcorr() function.** |

|  |
| --- |
| **Stepwise-Procedure:** ****Convolution****  1. Define two discrete signals 2. Perform convolution using the conv() function. 3. Plot the input signals and the resulting output signal.   Write code for performing convolution of the sequences *x(n)=* [3, 2, 1, 2] and *h(n)=* [1, 2, 1, 2] with and without *conv()* function  **Code (with inbuilt Function):**  x = [3, 2, 1, 2];  h = [1, 2, 1, 2];  y = conv(x, h);  n\_x = 0:length(x)-1;  n\_h = 0:length(h)-1;  n\_y = 0:length(y)-1;  figure;  subplot(3,1,1);  stem(n\_x, x, 'filled');  title('Input Signal x(n)');  xlabel('n');  ylabel('Amplitude');  subplot(3,1,2);  stem(n\_h, h, 'filled');  title('Impulse Response h(n)');  xlabel('n');  ylabel('Amplitude');  subplot(3,1,3);  stem(n\_y, y, 'filled');  title('Output Signal y(n) = x(n) \* h(n)');  xlabel('n');  ylabel('Amplitude');  **Code (without Function):**  x = [3, 2, 1, 2];  h = [1, 2, 1, 2];  len\_x = length(x);  len\_h = length(h);  N = len\_x + len\_h - 1;  y = zeros(1, N);  for n = 1:N  for k = 1:len\_x  if (n-k+1 > 0) && (n-k+1 <= len\_h)  y(n) = y(n) + x(k) \* h(n-k+1);  end  end  end  n\_x = 0:len\_x-1;  n\_h = 0:len\_h-1;  n\_y = 0:N-1;  % plotting the signals  figure;  subplot(3,1,1);  stem(n\_x, x, 'filled');  title('Input Signal x(n)');  xlabel('n');  ylabel('Amplitude');  subplot(3,1,2);  stem(n\_h, h, 'filled');  title('Impulse Response h(n)');  xlabel('n');  ylabel('Amplitude');  subplot(3,1,3);  stem(n\_y, y, 'filled');  title('Output Signal y(n) = x(n) \* h(n)');  xlabel('n');  ylabel('Amplitude');  **Output:**   ****Correlation****  1. Define two discrete signals. 2. Perform cross-correlation using the xcorr() function. 3. Perform auto-correlation on one of the signals using xcorr(). 4. Plot the signals and their correlation results.   Write code for performing cross- co**rrelation** of the sequences *x(n)=* [1, 2, 1, 1] and *y(n)=* [1, 1, 2, 1] with and without inbuilt function  **Code (with inbuilt Function):**  x = [1, 2, 1, 1];  y = [1, 1, 2, 1];  [cross\_corr, lags] = xcorr(x, y);  auto\_corr\_x = xcorr(x);  n\_x = 0:length(x)-1;  n\_y = 0:length(y)-1;  figure;  subplot(4, 1, 1);  stem(n\_x, x, 'filled');  title('Signal x(n)');  xlabel('n');  ylabel('Amplitude');  subplot(4, 1, 2);  stem(n\_y, y, 'filled');  title('Signal y(n)');  xlabel('n');  ylabel('Amplitude');  subplot(4, 1, 3);  stem(lags, cross\_corr, 'filled');  title('Cross-Correlation of x(n) and y(n)');  xlabel('Lag');  ylabel('Correlation');  lags\_auto = -length(x)+1:length(x)-1; subplot(4, 1, 4);  stem(lags\_auto, auto\_corr\_x, 'filled');  title('Auto-Correlation of x(n)');  xlabel('Lag');  ylabel('Correlation');  **Code (without Function):**  x = [1, 2, 1, 1];  y = [1, 1, 2, 1];  len\_x = length(x);  len\_y = length(y);  N\_cross = len\_x + len\_y - 1;  N\_auto = len\_x + len\_x - 1;  cross\_corr\_manual = zeros(1, N\_cross);  auto\_corr\_manual = zeros(1, N\_auto);  for lag = -(len\_y-1):(len\_x-1)  for n = max(1, lag+1):min(len\_x, len\_x+lag)  cross\_corr\_manual(lag + len\_y) = cross\_corr\_manual(lag + len\_y) + ...  x(n) \* y(n - lag);  end  end  for lag = -(len\_x-1):(len\_x-1)  for n = max(1, lag+1):min(len\_x, len\_x+lag)  auto\_corr\_manual(lag + len\_x) = auto\_corr\_manual(lag + len\_x) + ...  x(n) \* x(n - lag);  end  end  n\_x = 0:len\_x-1;  n\_y = 0:len\_y-1;  figure;  % plot signal x  subplot(4, 1, 1);  stem(n\_x, x, 'filled');  title('Signal x(n)');  xlabel('n');  ylabel('Amplitude');  % plot signal y  subplot(4, 1, 2);  stem(n\_y, y, 'filled');  title('Signal y(n)');  xlabel('n');  ylabel('Amplitude');  % manual cross-correlation result  subplot(4, 1, 3);  stem(-len\_y+1:len\_x-1,cross\_corr\_manual,'filled');  title('Manual Cross-Correlation of x(n) and y(n)');  xlabel('Lag');  ylabel('Correlation');  % plot manual auto-correlation result  subplot(4, 1, 4);  stem(-len\_x+1:len\_x-1 ,auto\_corr\_manual,'filled');  title('Manual Auto-Correlation of x(n)');  xlabel('Lag');  ylabel('Correlation');  **Output:** |

|  |
| --- |
| **Conclusion:** |
| The experiment successfully demonstrated the implementation of convolution and correlation using MATLAB, both with and without inbuilt functions. It provided a clear understanding of their significance in analyzing and processing discrete-time signals. |

|  |
| --- |
| **Post Lab Questions:** |
| 1. **Explain the difference between convolution and correlation.**   Convolution: It is used to analyze the output of a linear time-invariant (LTI) system by combining the input signal and the system's impulse response. It involves flipping one of the signals and shifting it before multiplying and summing.  Correlation: It measures the similarity between two signals as a function of the time-lag applied to one of them. Unlike convolution, the signal is not flipped.   1. **How is auto-correlation different from cross-correlation?**   Auto-correlation: Measures the similarity of a signal with a time-shifted version of itself. It helps in identifying repeating patterns or periodicity in the signal.  Cross-correlation: Measures the similarity between two different signals as a function of time-lag, often used to determine how one signal relates to another.   1. **Can correlation be used to detect time delay between two signals? How?**   Yes, correlation can detect time delay. The time-lag at which the cross-correlation is maximum indicates the time shift or delay between the two signals.   1. **How does the length of the output signal differ in convolution and correlation?**   For two signals of lengths NN and MM:   * Convolution: Output length is N+M−1N + M - 1. * Correlation: Output length is also N+M−1N + M - 1, but the time-lags range symmetrically around zero.  1. **Explain with the help of an example the steps required to transform linear convolution with circular convolution and vice-versa.**   Linear to Circular Convolution: Pad both signals with zeros to make their lengths equal to N+M−1N + M - 1, where NN and MM are the original lengths of the signals. Perform circular convolution on the padded signals.  Circular to Linear Convolution: Ensure the lengths of both signals match N+M−1N + M - 1. Perform circular convolution, and the result will match the linear convolution.  Example: Given x(n)=[1,2]x(n) = [1, 2] and h(n)=[3,4]h(n) = [3, 4]:   * Linear convolution result: [3,10,8][3, 10, 8]. * Circular convolution (with padding to length 3): [3,10,8][3, 10, 8]. |

|  |
| --- |
| **Signature of faculty in-charge with Date:** |