NumCSE

Autumn Semester 2017

Prof. Rima Alaifari

Exercise sheet 6
Convolution and fast Fourier transform

A. Dabrowski

Problem 6.1: Convolution and FFT

We examine a computationally useful link between discrete and circular convolution via the fast fourier transform. We apply it to calculate the filtering of a noisy signal.

Template: `conv.cpp`

Let $\mathbf{x} \in \mathbb{R}^m$ and $\mathbf{y} \in \mathbb{R}^n$ be two vectors with $n < m$. Implement `C++` functions which take as arguments Eigen vectors `x` and `y` and return:

(a) the discrete convolution of `x` and `y` (implement the definition of convolution);

(b) the vector given by `ifft(fft(x)fft(ym))`, where `ym` has been obtained by appending $n - m$ zeros to `y`;

(c) the vector given by `ifft(fft(xL)fft(yL))`, where `xL` and `yL` have been obtained by padding with zeros `x` and `y` to length $L = m + n - 1$.

(d) Will the result of Point a coincide/differ with the result from Point b or Point c? Why?

(e) Test your code for `x` an input vector given by a constant signal with some random uniform noise and `y` a Gaussian truncated filter (see the template). Does the application of the filter reduce the noise of the signal?

Problem 6.2: 2D convolution, FFT2 and Laplace filter

We review two dimensional convolution, its relation with the discrete Fourier transform and we implement a discrete Laplacian filter.

Template: `conv2.cpp`

Consider the 2-dimensional infinite arrays $X = (X_{k_1,k_2})_{k_1,k_2 \in \mathbb{Z}}$ and $Y = (Y_{k_1,k_2})_{k_1,k_2 \in \mathbb{Z}}$. We can define their convolution as the 2-dimensional infinite array $X * Y$ such that

$$(X * Y)_{k_1,k_2} = \sum_{j_1=-\infty}^{\infty} \sum_{j_2=-\infty}^{\infty} X_{j_1,j_2} Y_{k_1-j_1,k_2-j_2}.$$

In the same way as in the 1-dimensional case, given two matrices $X \in \mathbb{R}^{m_1,m_2}$ and $Y \in \mathbb{R}^{n_1,n_2}$ we can define their *discrete convolution* as the convolution between the zero extensions of X and Y trimmed of unnecessary zeros, and their *circular convolution* as the smallest period of the convolution between the periodic extension of X and the zero extension of Y.

Consider two matrices $A \in \mathbb{R}^{n,m}$ and $F \in \mathbb{R}^{k,k}$ with $k < n, m$.

(a) How should we extend A and F to larger matrices $\tilde{A}$ and $\tilde{F}$ so that the discrete convolution of A with F is equal to the circular convolution of $\tilde{A}$ with $\tilde{F}$?

(b) By recalling to the 1-dimensional fast fourier transform, implement a `C++` function `fft2` which returns the 2-dimensional fast fourier transform of a matrix.

(c) Implement an efficient `C++` function with arguments A and F which implements their discrete convolution.

Hint: use the result derived in the previous steps and the two dimensional circular convolution theorem.

The discrete Laplacian filter is defined as

$$F = \begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{pmatrix}.$$

It is often used to detect edges in images.

(d) Test your filter on the black and white "picture" provided in the template.