The Fast Fourier Transform

Trigonometric Fourier series in $L_2(0, 2\pi)$

Complex exponential function

The Discrete Fourier Transform

The Fast Fourier Transform

Image compression by DFT

Numerical Analysis
The discrete and fast Fourier Transform

by Csaba Gáspár

Széchenyi István University

2020, autumn semester

Trigonometric Fourier series in $L_2(0, 2\pi)$

An arbitrary real function $f \in L_2(0, 2\pi)$ can be expressed as a trigonometric Fourier series which is convergent with respect to the $L_2(0, 2\pi)$-norm:

$$f(x) = a_0 + \sum_{k=1}^{\infty} a_k \cos kx + \sum_{k=1}^{\infty} b_k \sin kx$$

where the coefficients can be calculated as:

$$a_0 = \frac{1}{2\pi} \int_0^{2\pi} f(x)\, dx,$$

$$a_k = \frac{1}{\pi} \int_0^{2\pi} f(x) \cos kx\, dx, \quad b_k = \frac{1}{\pi} \int_0^{2\pi} f(x) \sin kx\, dx$$

Complex exponential function

For any $z \in \mathbf{C}$, define the **exponential series** as:

$$e^z := \sum_{k=0}^{\infty} \frac{z^k}{k!} = 1 + \frac{z}{1!} + \frac{z^2}{2!} + \frac{z^3}{3!} + \ldots$$

Euler's formula

For every $t \in \mathbf{R}$: $\quad e^{it} = \cos t + i \cdot \sin t$

Utilizing the well-known Taylor series of the sine and cosine functions:

$$\cos t = 1 - \frac{t^2}{2!} + \frac{t^4}{4!} - \ldots, \qquad \sin t = \frac{t}{1!} - \frac{t^3}{3!} + \frac{t^5}{5!} - \ldots$$

which implies that:

$$e^{it} = 1 + \frac{it}{1!} + \frac{i^2t^2}{2!} + \frac{i^3t^3}{3!} + \frac{i^4t^4}{4!} + \ldots = 1 + \frac{it}{1!} - \frac{t^2}{2!} - \frac{it^3}{3!} + \frac{t^4}{4!} + \ldots$$

Separating the real and imaginary parts, we have the theorem.

Complex exponential function

For any $z \in \mathbf{C}$, define the **exponential series** as:

$$e^z := \sum_{k=0}^{\infty} \frac{z^k}{k!} = 1 + \frac{z}{1!} + \frac{z^2}{2!} + \frac{z^3}{3!} + \ldots$$

Euler's formula

For every $t \in \mathbf{R}$: $\quad e^{it} = \cos t + i \cdot \sin t$

Utilizing the well-known Taylor series of the sine and cosine functions:

$$\cos t = 1 - \frac{t^2}{2!} + \frac{t^4}{4!} - \ldots, \qquad \sin t = \frac{t}{1!} - \frac{t^3}{3!} + \frac{t^5}{5!} - \ldots$$

which implies that:

$$e^{it} = 1 + \frac{it}{1!} + \frac{i^2t^2}{2!} + \frac{i^3t^3}{3!} + \frac{i^4t^4}{4!} + \ldots = 1 + \frac{it}{1!} - \frac{t^2}{2!} - \frac{it^3}{3!} + \frac{t^4}{4!} + \ldots$$

Separating the real and imaginary parts, we have the theorem.

Complex exponential function

For any $z \in \mathbf{C}$, define the **exponential series** as:

$$e^z := \sum_{k=0}^{\infty} \frac{z^k}{k!} = 1 + \frac{z}{1!} + \frac{z^2}{2!} + \frac{z^3}{3!} + \ldots$$

Euler's formula

For every $t \in \mathbf{R}$: $\quad e^{it} = \cos t + i \cdot \sin t$

Utilizing the well-known Taylor series of the sine and cosine functions:

$$\cos t = 1 - \frac{t^2}{2!} + \frac{t^4}{4!} - \ldots, \qquad \sin t = \frac{t}{1!} - \frac{t^3}{3!} + \frac{t^5}{5!} - \ldots$$

which implies that:

$$e^{it} = 1 + \frac{it}{1!} + \frac{i^2t^2}{2!} + \frac{i^3t^3}{3!} + \frac{i^4t^4}{4!} + \ldots = 1 + \frac{it}{1!} - \frac{t^2}{2!} - \frac{it^3}{3!} + \frac{t^4}{4!} + \ldots$$

Separating the real and imaginary parts, we have the theorem.

The Fast Fourier Transform

Trigonometric Fourier series in $L_2(0, 2\pi)$

Complex exponential function

The Discrete Fourier Transform

The Fast Fourier Transform

Image compression by DFT

The Discrete Fourier Transform (DFT)

If $f_0, f_1, ..., f_{N-1} \in \mathbf{C}$ is a finite sequence, then define its **discrete Fourier transform** as:

$$\hat{f}_k := \sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi ikj}{N}} \qquad (k = 0, 1, ..., N-1)$$

Relationship with the Fourier series: Let f be a continuous function defined on the interval $[0, 2\pi)$, and denote by $f_j := f(\frac{2\pi j}{N})$. Then the sum $\frac{2\pi}{N} \cdot \sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi ikj}{N}}$ is a Riemann sum of the integral $\int_0^{2\pi} f(x)e^{ikx}\,dx$. Utilizing Euler's formula, we have:

$$\frac{1}{N}\hat{f}_k = \frac{1}{N}\sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi ikj}{N}} \approx \frac{1}{2\pi}\int_0^{2\pi} f(x)e^{ikx}\,dx =$$

$$= \frac{1}{2\pi}\int_0^{2\pi} f(x)\cos kx\,dx + i \cdot \frac{1}{2\pi}\int_0^{2\pi} f(x)\sin kx\,dx = a_k + ib_k,$$

where a_k, b_k are the trigonometric Fourier coefficients.

The Discrete Fourier Transform (DFT)

If $f_0, f_1, ..., f_{N-1} \in \mathbf{C}$ is a finite sequence, then define its **discrete Fourier transform** as:

$$\hat{f}_k := \sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi ikj}{N}} \qquad (k = 0, 1, ..., N-1)$$

Relationship with the Fourier series: Let f be a continuous function defined on the interval $[0, 2\pi)$, and denote by $f_j := f(\frac{2\pi j}{N})$. Then the sum $\frac{2\pi}{N} \cdot \sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi ikj}{N}}$ is a Riemann sum of the integral $\int_0^{2\pi} f(x)e^{ikx}\,dx$.

Utilizing Euler's formula, we have:

$$\frac{1}{N}\hat{f}_k = \frac{1}{N}\sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi ikj}{N}} \approx \frac{1}{2\pi}\int_0^{2\pi} f(x)e^{ikx}\,dx =$$

$$= \frac{1}{2\pi}\int_0^{2\pi} f(x)\cos kx\,dx + i\cdot\frac{1}{2\pi}\int_0^{2\pi} f(x)\sin kx\,dx = a_k + ib_k,$$

where a_k, b_k are the trigonometric Fourier coefficients.

The Discrete Fourier Transform (DFT)

Numerical Analysis

by Csaba Gáspár

The Fast Fourier Transform

Trigonometric Fourier series in $L_2(0, 2\pi)$

Complex exponential function

The Discrete Fourier Transform

The Fast Fourier Transform

Image compression by DFT

If $f_0, f_1, ..., f_{N-1} \in \mathbf{C}$ is a finite sequence, then define its **discrete Fourier transform** as:

$$\hat{f}_k := \sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi ikj}{N}} \qquad (k = 0, 1, ..., N-1)$$

Relationship with the Fourier series: Let f be a continuous function defined on the interval $[0, 2\pi)$, and denote by $f_j := f(\frac{2\pi j}{N})$. Then the sum $\frac{2\pi}{N} \cdot \sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi ikj}{N}}$ is a Riemann sum of the integral $\int_0^{2\pi} f(x)e^{ikx}\,dx$. Utilizing Euler's formula, we have:

$$\frac{1}{N}\hat{f}_k = \frac{1}{N}\sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi ikj}{N}} \approx \frac{1}{2\pi}\int_0^{2\pi} f(x)e^{ikx}\,dx =$$

$$= \frac{1}{2\pi}\int_0^{2\pi} f(x)\cos kx\,dx + i\cdot\frac{1}{2\pi}\int_0^{2\pi} f(x)\sin kx\,dx = a_k + ib_k,$$

where a_k, b_k are the trigonometric Fourier coefficients.

The Discrete Fourier Transform (DFT)

Numerical Analysis

by Csaba Gáspár

The Fast Fourier Transform

Trigonometric Fourier series in $L_2(0, 2\pi)$

Complex exponential function

The Discrete Fourier Transform

The Fast Fourier Transform

Image compression by DFT

If $f_0, f_1, ..., f_{N-1} \in \mathbf{C}$ is a finite sequence, then define its **discrete Fourier transform** as:

$$\hat{f}_k := \sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi ikj}{N}} \qquad (k = 0, 1, ..., N-1)$$

Relationship with the Fourier series: Let f be a continuous function defined on the interval $[0, 2\pi)$, and denote by $f_j := f(\frac{2\pi j}{N})$. Then the sum $\frac{2\pi}{N} \cdot \sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi ikj}{N}}$ is a Riemann sum of the integral $\int_0^{2\pi} f(x)e^{ikx}\, dx$. Utilizing Euler's formula, we have:

$$\frac{1}{N}\hat{f}_k = \frac{1}{N}\sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi ikj}{N}} \approx \frac{1}{2\pi}\int_0^{2\pi} f(x)e^{ikx}\, dx =$$

$$= \frac{1}{2\pi}\int_0^{2\pi} f(x)\cos kx\, dx + i \cdot \frac{1}{2\pi}\int_0^{2\pi} f(x)\sin kx\, dx = a_k + ib_k,$$

where a_k, b_k are the trigonometric Fourier coefficients.

Numerical Analysis

by Csaba Gáspár

The Fast Fourier Transform

Trigonometric Fourier series in $L_2(0, 2\pi)$

Complex exponential function

The Discrete Fourier Transform

The Fast Fourier Transform

Image compression by DFT

The inverse Discrete Fourier Transform (iDFT)

Every finite sequence can uniquely be reconstructed from its DFT, namely:

$$f_k := \frac{1}{N} \cdot \sum_{j=0}^{N-1} \hat{f}_j \cdot e^{-\frac{2\pi ikj}{N}} \qquad (k = 0, 1, ..., N-1)$$

$$\frac{1}{N}\sum_{j=0}^{N-1} \hat{f}_j e^{-\frac{2\pi ikj}{N}} = \frac{1}{N}\sum_{j=0}^{N-1}\sum_{r=0}^{N-1} f_r e^{\frac{2\pi irj}{N}} e^{-\frac{2\pi ikj}{N}} = \sum_{r=0}^{N-1} f_r \cdot \frac{1}{N}\sum_{j=0}^{N-1} e^{\frac{2\pi i(r-k)j}{N}}$$

$$= \sum_{r=0}^{N-1} f_r \cdot \frac{1}{N}\sum_{j=0}^{N-1} z^j \qquad (z := e^{\frac{2\pi i(r-k)}{N}})$$

If $r = k$, then $z = 1$, therefore the inner sum equals to 1. If $r \neq k$, then the inner sum is the sum of a finite geometric sequence, which equals to 0. This completes the proof.

The inverse Discrete Fourier Transform (iDFT)

Every finite sequence can uniquely be reconstructed from its DFT, namely:

$$f_k := \frac{1}{N} \cdot \sum_{j=0}^{N-1} \hat{f}_j \cdot e^{-\frac{2\pi i k j}{N}} \qquad (k = 0, 1, ..., N-1)$$

$$\frac{1}{N} \sum_{j=0}^{N-1} \hat{f}_j e^{-\frac{2\pi i k j}{N}} = \frac{1}{N} \sum_{j=0}^{N-1} \sum_{r=0}^{N-1} f_r e^{\frac{2\pi i r j}{N}} e^{-\frac{2\pi i k j}{N}} = \sum_{r=0}^{N-1} f_r \cdot \frac{1}{N} \sum_{j=0}^{N-1} e^{\frac{2\pi i (r-k) j}{N}}$$

$$= \sum_{r=0}^{N-1} f_r \cdot \frac{1}{N} \sum_{j=0}^{N-1} z^j \qquad (z := e^{\frac{2\pi i (r-k)}{N}})$$

If $r = k$, then $z = 1$, therefore the inner sum equals to 1. If $r \neq k$, then the inner sum is the sum of a finite geometric sequence, which equals to 0. This completes the proof.

The inverse Discrete Fourier Transform (iDFT)

Every finite sequence can uniquely be reconstructed from its DFT, namely:

$$f_k := \frac{1}{N} \cdot \sum_{j=0}^{N-1} \hat{f}_j \cdot e^{-\frac{2\pi i k j}{N}} \qquad (k = 0, 1, ..., N-1)$$

$$\frac{1}{N}\sum_{j=0}^{N-1} \hat{f}_j e^{-\frac{2\pi i k j}{N}} = \frac{1}{N}\sum_{j=0}^{N-1}\sum_{r=0}^{N-1} f_r e^{\frac{2\pi i r j}{N}} e^{-\frac{2\pi i k j}{N}} = \sum_{r=0}^{N-1} f_r \cdot \frac{1}{N}\sum_{j=0}^{N-1} e^{\frac{2\pi i (r-k) j}{N}}$$

$$= \sum_{r=0}^{N-1} f_r \cdot \frac{1}{N}\sum_{j=0}^{N-1} z^j \qquad \left(z := e^{\frac{2\pi i (r-k)}{N}}\right)$$

If $r = k$, then $z = 1$, therefore the inner sum equals to 1. If $r \neq k$, then the inner sum is the sum of a finite geometric sequence, which equals to 0. This completes the proof.

The Discrete Fourier Transform, numerical features

Total computational cost: $\mathcal{O}(N^2)$, which is too high!

The Fast Fourier Transform (FFT)

Denote by $F_N : \mathbf{C}^N \to \mathbf{C}^N$ the linear operator of the DFT:

$$(F_N f)_k = \sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi i k j}{N}} \qquad (k = 0, 1, ..., N-1)$$

Assume that N is even: $N = 2N_1$. Let us separate the terms with even and odd indices in the expression of $(F_N f)_k$. First, let k be a 'small' index: $k = 0, 1, ..., N_1 - 1$:

$$(F_N f)_k = \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi i k \cdot 2\ell}{N}} + \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi i k \cdot (2\ell+1)}{N}} =$$

$$= \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi i k \cdot 2\ell}{N}} + e^{\frac{2\pi i k}{N}} \cdot \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi i k \cdot 2\ell}{N}} =$$

$$= \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi i k \ell}{N_1}} + e^{\frac{2\pi i k}{N}} \cdot \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi i k \ell}{N_1}}$$

The Fast Fourier Transform (FFT)

Denote by $F_N : \mathbf{C}^N \to \mathbf{C}^N$ the linear operator of the DFT:

$$(F_N f)_k = \sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi ikj}{N}} \qquad (k = 0, 1, ..., N-1)$$

Assume that N is even: $N = 2N_1$. Let us separate the terms with even and odd indices in the expression of $(F_N f)_k$. First, let k be a 'small' index: $k = 0, 1, ..., N_1 - 1$:

$$(F_N f)_k = \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi ik\cdot 2\ell}{N}} + \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi ik\cdot(2\ell+1)}{N}} =$$

$$= \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi ik\cdot 2\ell}{N}} + e^{\frac{2\pi ik}{N}} \cdot \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi ik\cdot 2\ell}{N}} =$$

$$= \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi ik\ell}{N_1}} + e^{\frac{2\pi ik}{N}} \cdot \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi ik\ell}{N_1}}$$

Numerical Analysis

by Csaba Gáspár

The Fast Fourier Transform

Trigonometric Fourier series in $L_2(0, 2\pi)$

Complex exponential function

The Discrete Fourier Transform

The Fast Fourier Transform

Image compression by DFT

Denote by $F_N : \mathbf{C}^N \to \mathbf{C}^N$ the linear operator of the DFT:

$$(F_N f)_k = \sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi ikj}{N}} \qquad (k = 0, 1, ..., N-1)$$

Assume that N is even: $N = 2N_1$. Let us separate the terms with even and odd indices in the expression of $(F_N f)_k$. First, let k be a 'small' index: $k = 0, 1, ..., N_1 - 1$:

$$(F_N f)_k = \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi ik\cdot 2\ell}{N}} + \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi ik\cdot(2\ell+1)}{N}} =$$

$$= \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi ik\cdot 2\ell}{N}} + e^{\frac{2\pi ik}{N}} \cdot \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi ik\cdot 2\ell}{N}} =$$

$$= \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi ik\ell}{N_1}} + e^{\frac{2\pi ik}{N}} \cdot \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi ik\ell}{N_1}}$$

The Fast Fourier Transform (FFT)

Denote by $F_N : \mathbf{C}^N \to \mathbf{C}^N$ the linear operator of the DFT:

$$(F_N f)_k = \sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi i k j}{N}} \qquad (k = 0, 1, ..., N-1)$$

Assume that N is even: $N = 2N_1$. Let us separate the terms with even and odd indices in the expression of $(F_N f)_k$. First, let k be a 'small' index: $k = 0, 1, ..., N_1 - 1$:

$$(F_N f)_k = \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi i k \cdot 2\ell}{N}} + \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi i k \cdot (2\ell+1)}{N}} =$$

$$= \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi i k \cdot 2\ell}{N}} + e^{\frac{2\pi i k}{N}} \cdot \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi i k \cdot 2\ell}{N}} =$$

$$= \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi i k \ell}{N_1}} + e^{\frac{2\pi i k}{N}} \cdot \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi i k \ell}{N_1}}$$

The Fast Fourier Transform (FFT)

Numerical Analysis

by Csaba Gáspár

The Fast Fourier Transform

Trigonometric Fourier series in $L_2(0, 2\pi)$

Complex exponential function

The Discrete Fourier Transform

The Fast Fourier Transform

Image compression by DFT

Denote by $F_N : \mathbf{C}^N \to \mathbf{C}^N$ the linear operator of the DFT:

$$(F_N f)_k = \sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi i k j}{N}} \qquad (k = 0, 1, ..., N-1)$$

Assume that N is even: $N = 2N_1$. Let us separate the terms with even and odd indices in the expression of $(F_N f)_k$. Now consider the 'great' indices having the form $N_1 + k$, where $k = 0, 1, ..., N_1 - 1$:

$$(F_N f)_{N_1+k} = \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi i (N_1+k)\cdot 2\ell}{N}} + \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi i (N_1+k)\cdot(2\ell+1)}{N}} =$$

$$= \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi i (N_1+k)\cdot 2\ell}{N}} + e^{\frac{2\pi i (N_1+k)}{N}} \cdot \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi i (N_1+k)\cdot 2\ell}{N}} =$$

$$= \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi i k \ell}{N_1}} - e^{\frac{2\pi i k}{N}} \cdot \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi i k \ell}{N_1}}$$

The Fast Fourier Transform

Trigonometric Fourier series in $L_2(0, 2\pi)$

Complex exponential function

The Discrete Fourier Transform

The Fast Fourier Transform

Image compression by DFT

The Fast Fourier Transform (FFT)

Denote by $F_N : \mathbf{C}^N \to \mathbf{C}^N$ the linear operator of the DFT:

$$(F_N f)_k = \sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi i k j}{N}} \qquad (k = 0, 1, ..., N-1)$$

Assume that N is even: $N = 2N_1$. Let us separate the terms with even and odd indices in the expression of $(F_N f)_k$. Now consider the 'great' indices having the form $N_1 + k$, where $k = 0, 1, ..., N_1 - 1$:

$$(F_N f)_{N_1+k} = \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi i (N_1+k)\cdot 2\ell}{N}} + \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi i (N_1+k)\cdot(2\ell+1)}{N}} =$$

$$= \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi i (N_1+k)\cdot 2\ell}{N}} + e^{\frac{2\pi i (N_1+k)}{N}} \cdot \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi i (N_1+k)\cdot 2\ell}{N}} =$$

$$= \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi i k \ell}{N_1}} - e^{\frac{2\pi i k}{N}} \cdot \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi i k \ell}{N_1}}$$

The Fast Fourier Transform
Trigonometric Fourier series in $L_2(0, 2\pi)$
Complex exponential function
The Discrete Fourier Transform
The Fast Fourier Transform
Image compression by DFT

The Fast Fourier Transform (FFT)

Denote by $F_N : \mathbf{C}^N \to \mathbf{C}^N$ the linear operator of the DFT:

$$(F_N f)_k = \sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi i k j}{N}} \qquad (k = 0, 1, ..., N-1)$$

Assume that N is even: $N = 2N_1$. Let us separate the terms with even and odd indices in the expression of $(F_N f)_k$. Now consider the 'great' indices having the form $N_1 + k$, where $k = 0, 1, ..., N_1 - 1$:

$$(F_N f)_{N_1+k} = \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi i (N_1+k)\cdot 2\ell}{N}} + \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi i (N_1+k)\cdot(2\ell+1)}{N}} =$$

$$= \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi i (N_1+k)\cdot 2\ell}{N}} + e^{\frac{2\pi i (N_1+k)}{N}} \cdot \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi i (N_1+k)\cdot 2\ell}{N}} =$$

$$= \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi i k \ell}{N_1}} - e^{\frac{2\pi i k}{N}} \cdot \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi i k \ell}{N_1}}$$

The Fast Fourier Transform (FFT)

Denote by $F_N : \mathbf{C}^N \to \mathbf{C}^N$ the linear operator of the DFT:

$$(F_N f)_k = \sum_{j=0}^{N-1} f_j \cdot e^{\frac{2\pi ikj}{N}} \qquad (k = 0, 1, ..., N-1)$$

Assume that N is even: $N = 2N_1$. Let us separate the terms with even and odd indices in the expression of $(F_N f)_k$. Now consider the 'great' indices having the form $N_1 + k$, where $k = 0, 1, ..., N_1 - 1$:

$$(F_N f)_{N_1+k} = \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi i(N_1+k)\cdot 2\ell}{N}} + \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi i(N_1+k)\cdot(2\ell+1)}{N}} =$$

$$= \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi i(N_1+k)\cdot 2\ell}{N}} + e^{\frac{2\pi i(N_1+k)}{N}} \cdot \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi i(N_1+k)\cdot 2\ell}{N}} =$$

$$= \sum_{\ell=0}^{N_1-1} f_{2\ell} \cdot e^{\frac{2\pi ik\ell}{N_1}} - e^{\frac{2\pi ik}{N}} \cdot \sum_{\ell=0}^{N_1-1} f_{2\ell+1} \cdot e^{\frac{2\pi ik\ell}{N_1}}$$

The Fast Fourier Transform (FFT)

For both the 'small' and 'great' indices, both sums on the right-hand sides are discrete Fourier transforms with smaller vectors. The procedure can recursively be continued, if N is a power of two.

Trigonometric Fourier series in $L_2(0, 2\pi)$

Complex exponential function

The Discrete Fourier Transform

The Fast Fourier Transform

Image compression by DFT

The Fast Fourier Transform, recursive algorithm

With recursive invocations:

- $N_1 := N/2$

$$f^{even} := (f_0, f_2, ..., f_{2N_1-2}), \quad f^{odd} := (f_1, f_3, ..., f_{2N_1-1})$$

- $\hat{f}^{even} := F_{N_1} f^{even}, \quad \hat{f}^{odd} := F_{N_1} f^{odd}$
-

$$(F_N f)_k \quad := \hat{f}_k^{even} + e^{\frac{2\pi i k}{N}} \cdot \hat{f}_k^{odd} \qquad (k = 0, 1, ..., N_1 - 1)$$

$$(F_N f)_{N_1+k} := \hat{f}_k^{even} - e^{\frac{2\pi i k}{N}} \cdot \hat{f}_k^{odd} \qquad (k = 0, 1, ..., N_1 - 1)$$

where for $N = 1$, $F_1 f := f$ (f has one component only)

The Fast Fourier Transform, recursive algorithm

With recursive invocations:

- $N_1 := N/2$

$$f^{even} := (f_0, f_2, ..., f_{2N_1-2}), \quad f^{odd} := (f_1, f_3, ..., f_{2N_1-1})$$

- $\hat{f}^{even} := F_{N_1} f^{even}, \quad \hat{f}^{odd} := F_{N_1} f^{odd}$

-

$$(F_N f)_k \quad := \hat{f}^{even}_k + e^{\frac{2\pi i k}{N}} \cdot \hat{f}^{odd}_k \qquad (k = 0, 1, ..., N_1 - 1)$$

$$(F_N f)_{N_1+k} := \hat{f}^{even}_k - e^{\frac{2\pi i k}{N}} \cdot \hat{f}^{odd}_k \qquad (k = 0, 1, ..., N_1 - 1)$$

where for $N = 1$, $F_1 f := f$ (f has one component only)

Numerical Analysis
by Csaba Gáspár
The Fast Fourier Transform
Trigonometric Fourier series in $L_2(0,2\pi)$
Complex exponential function
The Discrete Fourier Transform
The Fast Fourier Transform
Image compression by DFT

With recursive invocations:

- $N_1 := N/2$

$$f^{even} := (f_0, f_2, ..., f_{2N_1-2}), \quad f^{odd} := (f_1, f_3, ..., f_{2N_1-1})$$

- $\hat{f}^{even} := F_{N_1} f^{even}, \quad \hat{f}^{odd} := F_{N_1} f^{odd}$
-

$$(F_N f)_k := \hat{f}_k^{even} + e^{\frac{2\pi ik}{N}} \cdot \hat{f}_k^{odd} \qquad (k = 0, 1, ..., N_1 - 1)$$

$$(F_N f)_{N_1+k} := \hat{f}_k^{even} - e^{\frac{2\pi ik}{N}} \cdot \hat{f}_k^{odd} \qquad (k = 0, 1, ..., N_1 - 1)$$

where for $N = 1$, $F_1 f := f$ (f has one component only)

The Fast Fourier Transform, recursive algorithm

With recursive invocations:

- $N_1 := N/2$

$$f^{even} := (f_0, f_2, ..., f_{2N_1-2}), \quad f^{odd} := (f_1, f_3, ..., f_{2N_1-1})$$

- $\hat{f}^{even} := F_{N_1} f^{even}, \quad \hat{f}^{odd} := F_{N_1} f^{odd}$
-

$$(F_N f)_k \quad := \hat{f}_k^{even} + e^{\frac{2\pi i k}{N}} \cdot \hat{f}_k^{odd} \qquad (k = 0, 1, ..., N_1 - 1)$$

$$(F_N f)_{N_1+k} := \hat{f}_k^{even} - e^{\frac{2\pi i k}{N}} \cdot \hat{f}_k^{odd} \qquad (k = 0, 1, ..., N_1 - 1)$$

where for $N = 1$, $F_1 f := f$ (f has one component only)

The Fast Fourier Transform, numerical features

Total computational cost: $\mathcal{O}(N \cdot \log N)$, which is much smaller than the computational cost of the original DFT.

Example:

$$f_0 \quad f_1 \quad f_2 \quad f_3 \quad f_4 \quad f_5 \quad f_6 \quad f_7$$

$$f_0 \quad f_2 \quad f_4 \quad f_6 \;|\; f_1 \quad f_3 \quad f_5 \quad f_7$$

$$f_0 \quad f_4 \;|\; f_2 \quad f_6 \;|\; f_1 \quad f_5 \;|\; f_3 \quad f_7$$

$$f_0 \;|\; f_4 \;|\; f_2 \;|\; f_6 \;|\; f_1 \;|\; f_5 \;|\; f_3 \;|\; f_7$$

The Fast Fourier Transform, numerical features

Numerical Analysis

by Csaba Gáspár

The Fast Fourier Transform

Trigonometric Fourier series in $L_2(0, 2\pi)$

Complex exponential function

The Discrete Fourier Transform

The Fast Fourier Transform

Image compression by DFT

Total computational cost: $\mathcal{O}(N \cdot \log N)$, which is much smaller than the computational cost of the original DFT.

Example:

$$f_0 \quad f_1 \quad f_2 \quad f_3 \quad f_4 \quad f_5 \quad f_6 \quad f_7$$

$$f_0 \quad f_2 \quad f_4 \quad f_6 \;\Big|\; f_1 \quad f_3 \quad f_5 \quad f_7$$

$$f_0 \quad f_4 \;\Big|\; f_2 \quad f_6 \;\Big|\; f_1 \quad f_5 \;\Big|\; f_3 \quad f_7$$

$$f_0 \;\Big|\; f_4 \;\Big|\; f_2 \;\Big|\; f_6 \;\Big|\; f_1 \;\Big|\; f_5 \;\Big|\; f_3 \;\Big|\; f_7$$

The Fast Fourier Transform, numerical features

Numerical Analysis

by Csaba Gáspár

The Fast Fourier Transform

Trigonometric Fourier series in $L_2(0, 2\pi)$

Complex exponential function

The Discrete Fourier Transform

The Fast Fourier Transform

Image compression by DFT

Total computational cost: $\mathcal{O}(N \cdot \log N)$, which is much smaller than the computational cost of the original DFT.

Example:

$$f_0 \quad f_1 \quad f_2 \quad f_3 \quad f_4 \quad f_5 \quad f_6 \quad f_7$$

$$f_0 \quad f_2 \quad f_4 \quad f_6 \;\big|\; f_1 \quad f_3 \quad f_5 \quad f_7$$

$$f_0 \quad f_4 \;\big|\; f_2 \quad f_6 \;\big|\; f_1 \quad f_5 \;\big|\; f_3 \quad f_7$$

$$f_0 \;\big|\; f_4 \;\big|\; f_2 \;\big|\; f_6 \;\big|\; f_1 \;\big|\; f_5 \;\big|\; f_3 \;\big|\; f_7$$

The Fast Fourier Transform, numerical features

Total computational cost: $\mathcal{O}(N \cdot \log N)$, which is much smaller than the computational cost of the original DFT.

Example:

$$f_0 \quad f_1 \quad f_2 \quad f_3 \quad f_4 \quad f_5 \quad f_6 \quad f_7$$

$$f_0 \quad f_2 \quad f_4 \quad f_6 \;|\; f_1 \quad f_3 \quad f_5 \quad f_7$$

$$f_0 \quad f_4 \;|\; f_2 \quad f_6 \;|\; f_1 \quad f_5 \;|\; f_3 \quad f_7$$

$$f_0 \;|\; f_4 \;|\; f_2 \;|\; f_6 \;|\; f_1 \;|\; f_5 \;|\; f_3 \;|\; f_7$$

The 2D Discrete Fourier Transform

The DFT of a matrix $f \in \mathbf{M}_{N\times N}$ is the matrix $\hat{f} \in \mathbf{M}_{N\times N}$ with the following entries:

$$\hat{f}_{k,j} := \sum_{r=0}^{N-1} \sum_{s=0}^{N-1} f_{r,s} \cdot e^{\frac{2\pi ikr}{N}} e^{\frac{2\pi ijs}{N}}$$

The algorithm of the computation of the DFT of a matrix:

- For every row of the matrix f, substitute the 1D DFT of the corresponding row.
- For every column of this matrix, substitute the 1D DFT of the corresponding column.

This results in the 2D DFT of the original matrix f.

The 2D Discrete Fourier Transform

The DFT of a matrix $f \in \mathbf{M}_{N\times N}$ is the matrix $\hat{f} \in \mathbf{M}_{N\times N}$ with the following entries:

$$\hat{f}_{k,j} := \sum_{r=0}^{N-1} \sum_{s=0}^{N-1} f_{r,s} \cdot e^{\frac{2\pi ikr}{N}} e^{\frac{2\pi ijs}{N}}$$

The algorithm of the computation of the DFT of a matrix:

- For every row of the matrix f, substitute the 1D DFT of the corresponding row.
- For every column of this matrix, substitute the 1D DFT of the corresponding column.

This results in the 2D DFT of the original matrix f.

The Fast Fourier Transform

Trigonometric Fourier series in $L_2(0, 2\pi)$

Complex exponential function

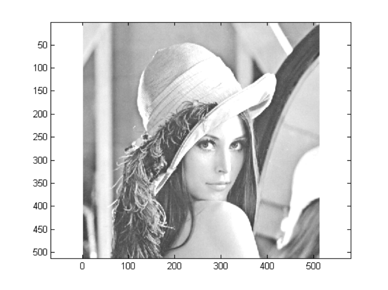
The Discrete Fourier Transform

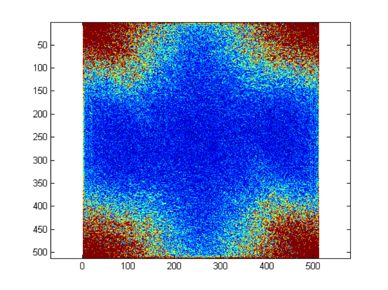
The Fast Fourier Transform

Image compression by DFT

Image compression by DFT

An N-by-N grayscale image can be considered an N-by-N matrix. In general, *there are a lot of Fourier coefficients that almost equal to* 0.

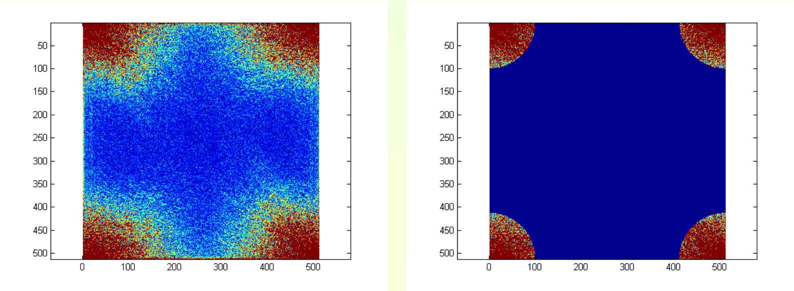




An 512-by-512 image and its Discrete Fourier Transform

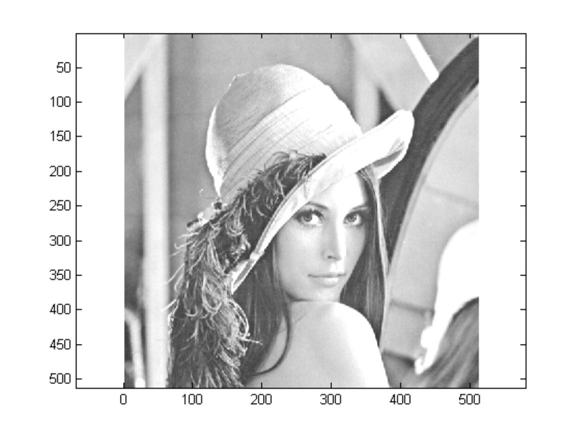
Image compression by DFT

The idea of the image compression: *to keep the Fourier coefficients which are closer to the corners than* r*, and to set the others to zero.* From the truncated DFT, the image can be approximately reconstructed.



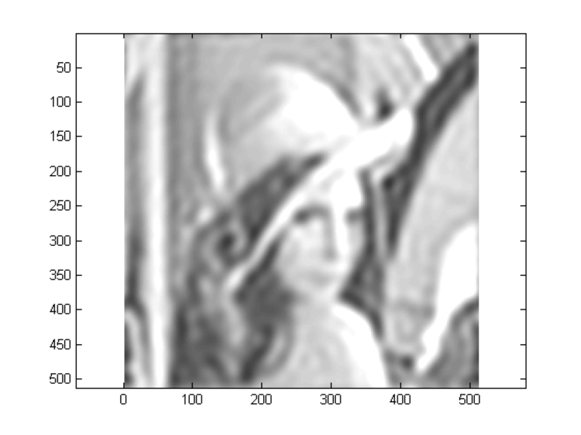
The original and truncated DFT of the image $(r = 100)$

Image compression by DFT



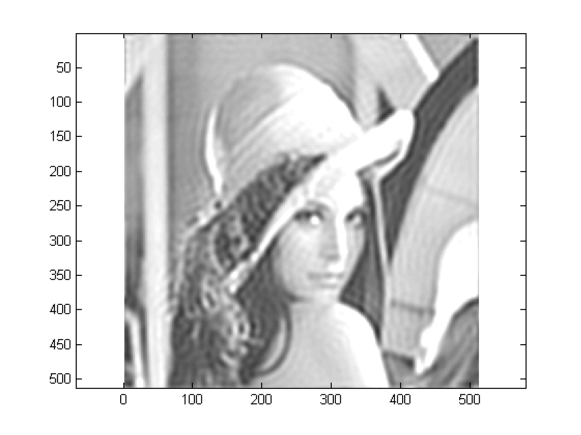
Original image

Image compression by DFT



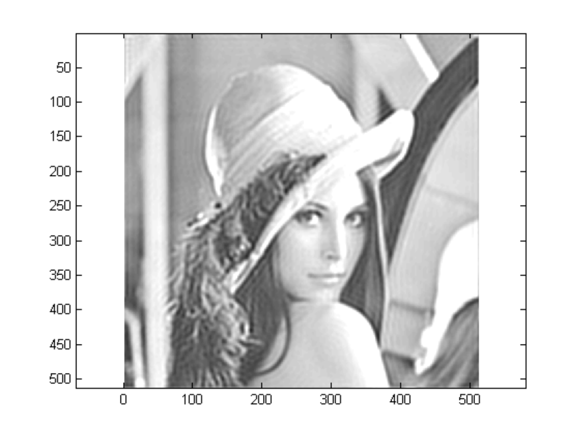
Reconstructed image, $r = 20$

Image compression by DFT



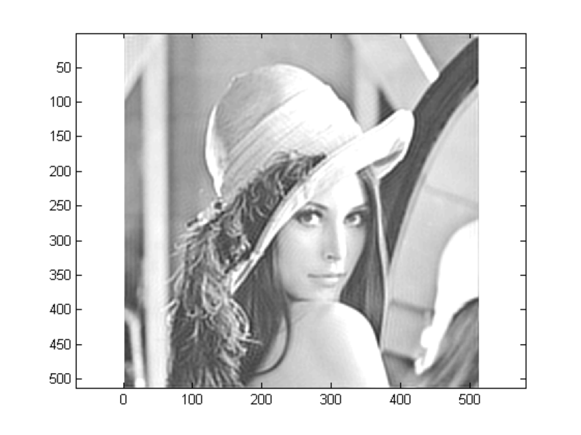
Reconstructed image, $r = 40$

Image compression by DFT



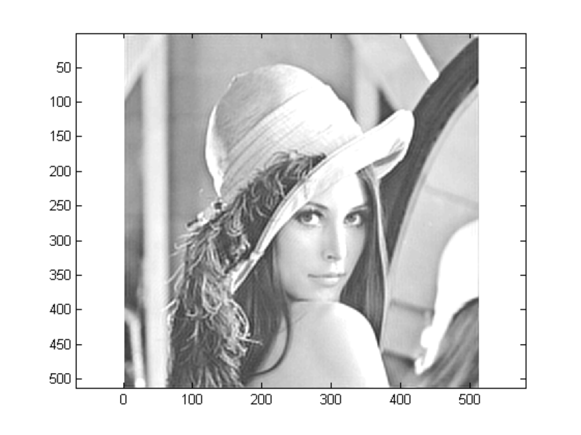
Reconstructed image, $r = 60$

Image compression by DFT



Reconstructed image, $r = 80$

Image compression by DFT



Reconstructed image, $r = 100$