Image Processing and Analysis – Math Part: The Discrete Fourier Transform

Ivar Farup

Based upon Chapter 2 of Broughton and Bryan's *Discrete Fourier Analysis and Wavelets* and Maciej Piętka's Lecture Notes from 2010

Overview

- The time domain and the frequency domain
- The one-dimensional DFT
- DFT examples
- Matrix formulation of the DFT
- The fast Fourier transform, FFT
- The two-dimensional DFT

The Time Domain and the Frequency Domain

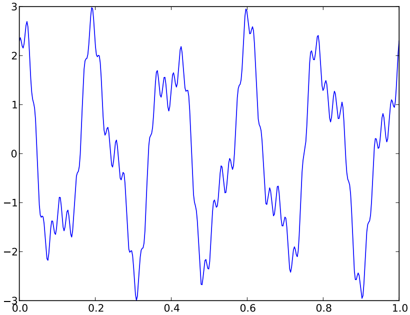
Basic Idea

- Sampled signals live in the time domain
- Sampled images live in the spatial domain
- Many operations are more easily performed in the frequency domain (e.g., removal of high frequencies)
- The discrete Fourier transform (DFT) and its inverse (IDFT) allow to move back and forth between the two representations

- The analog signal defined on $t \in [0, 1]$:

$$x(t) = 2\cos(2\pi \cdot 5t) + 0.8\sin(2\pi \cdot 12t) + 0.3\cos(2\pi \cdot 47t)$$

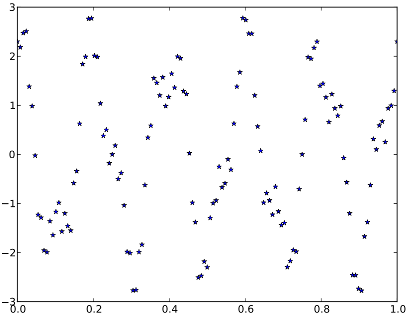
- Three waveforms with frequencies 5, 12, and 47 Hz



The Time Domain and the Frequency Domain

Example (cont)

- Sampled version of the signal with $\Delta T = 1/128$



- Goal: Determine the frequencies that make up $x(t)$ from the sampled signal
- Shannon–Nyquist: The analog signal $x(t)$ can be perfectly reconstructed if it contains no frequencies $> 64\,\text{Hz}$

The Time Domain and the Frequency Domain

Example (cont)

- Decompose $\mathbf{x}$ into a sum of basic waveforms $\mathbf{E}_{128,k}$ in the range $-64 < k \le 64$:

$$\mathbf{x} = \sum_{k=-63}^{64} c_k \mathbf{E}_{128,k}$$

- The coefficiencs c_k are equal to

$$c_k = \frac{(\mathbf{x}, \mathbf{E}_k)}{(\mathbf{E}_k, \mathbf{E}_k)} = \frac{1}{128} \sum_{m=0}^{127} x_m e^{-2\pi i k m/128}$$

- All c_k's are zero except for $c_{-47} = 0.15$, $c_{-12} = 0.4i$, $c_{-5} = 1$, $c_5 = 1$, $c_{12} = -0.4i$, $c_{47} = 0.15$
- Symmetry: $c_{-k} = \overline{c_k}$ for real-valued signals

- Reconstructed (synthesized) sampled signal:

$$\mathbf{x} = 1.0(\mathbf{E}_5 + \mathbf{E}_{-5}) - 0.4i(\mathbf{E}_{12} - \mathbf{E}_{-12}) + 0.15(\mathbf{E}_{47} + \mathbf{E}_{-47})$$

- Computed vector components

$$\begin{aligned}
x_m &= (e^{2\pi i 5m/128} + e^{-2\pi i 5m/128}) \\
&\quad - 0.4i(e^{2\pi i 12m/128} - e^{-2\pi i 12m/128}) \\
&\quad + 0.15(e^{2\pi i 47m/128} + e^{-2\pi i 47m/128}) \\
&= 2.0\cos(2\pi \cdot 5m\Delta T) + 0.8\sin(2\pi \cdot 12m\Delta T) \\
&\quad + 0.3\cos(2\pi \cdot 47m\Delta T) \\
&= x(m\Delta T)
\end{aligned}$$

This is the original signal. Thus, the finite vector $\mathbf{x}$ carries all the information necessary to reconstruct the analog signal $x(t)$

- Recall that $||\mathbf{x}||^2$ is the measure of energy of a signal
- In the orthogonal basis $\mathbf{E}_k$ we get

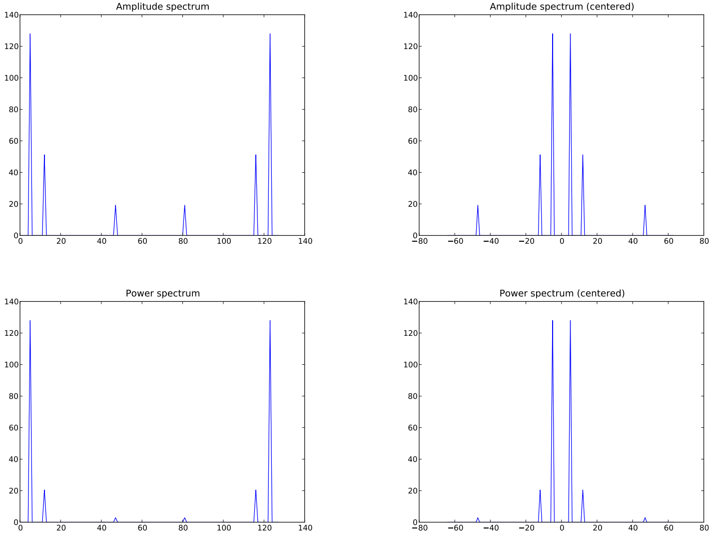
$$||\mathbf{x}||^2 = \sum_{k=-63}^{64} |c_k|^2 ||\mathbf{E}_k||^2 = N \sum_{k=-63}^{64} |c_k|^2$$

- The quantity $||c_k \mathbf{E}_k||^2 = 128|c_k|^2$ is the energy contributed by the waveform having the frequency index $|k|$.
  - $\epsilon_{5\,\mathrm{Hz}} = 128|c_{-5}|^2 + 128|c_5|^2 = 256$ (84.6% of the total energy
  - $\epsilon_{12\,\mathrm{Hz}} = 128|c_{-12}|^2 + 128|c_{12}|^2 = 40.96$ (13.5%)
  - $\epsilon_{47\,\mathrm{Hz}} = 128|c_{-47}|^2 + 128|c_{47}|^2 = 5.76$ (1.9%)
- c_k describes how the power is distributed with frequency

- c_k as a function of k is called the *spectrum* of the signal
- The plot of $|c_k|$ vs k is an *amplitude spectrum*
- $N|c_k|^2$ vs k is a *power spectrum* (or power spectral density, or energy spectrum)
- c_k is periodic because of aliasing; $c_k = c_{k+mN}$ for any $m \in \mathbb{Z}$
- $N|c_k|^2$ in the range $-N/2 < k \leq N/2$ or $0 \leq < k < N$: Two-sided spectrum (both positive and negative frequencies)
- $N(|c_k|^2 + |c_{-k}|^2)$ in the range $0 \leq k < N$: Single-sided spectrum (positive and negative frequencies lumped together)

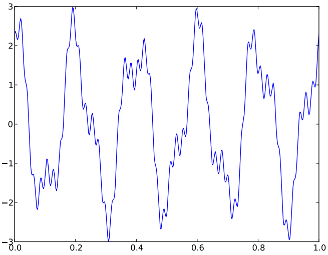
The Time Domain and the Frequency Domain

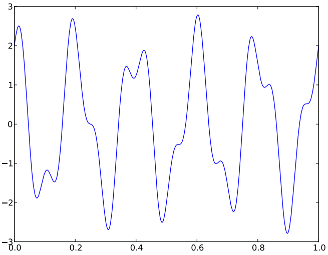
Spectra



- High frequency components considered to be artifacts due to noise
- Any spectral energy above the 'cutoff' frequency removed upon signal synthesis
- Denoised signal

$$\tilde{\mathbf{x}} = \sum_{k=-40}^{40} c_k \mathbf{E}_{128,k}$$





- Let $\mathbf{x} \in \mathbb{C}^N$ be a vector $(x_0, x_1, \ldots, x_{N-1})$
- The discrete Fourier transform of $\mathbf{x}$ is the vector $\mathbf{X} \in \mathbb{C}^N$ with components

$$X_k = (\mathbf{x}, \mathbf{E}_{N,k}) = \sum_{m=0}^{N-1} x_m e^{-2\pi i k m/N}$$

  for $0 \leq k \leq N-1$
- The DFT is a frequency decomposition of the signal

- The orthogonal decomposition of $\mathbf{x}$ can be written as

$$\mathbf{x} = \frac{1}{N} \sum_{k=0}^{N-1} X_k \mathbf{E}_{N,k}$$

- Thus, the inverse discrete Fourier transform (IDFT) of the vector $\mathbf{X} = (X_0, X_1, \ldots, X_{N-1}) \in \mathbb{C}^N$ is the vector $\mathbf{x} \in \mathbb{C}^N$ with components

$$x_m = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{2\pi i k m / N}$$

- (Another common definition uses a factor $1/\sqrt{N}$ in front of both DFT and IDFT.)

- The DFT coefficients X_k are periodic in k with period N
- X_k's can be computed in any range of length N
  - $0 \leq k \leq N-1$
  - $1 \leq k \leq N$
  - $-N/2 < k \leq N/2$
- The 'zero-frequency' coefficient c_0 is the constant-term contribution to $\mathbf{x}$, i.e., the arithmetic mean of $\mathbf{x}$,

$$c_0 = \frac{X_0}{N} = \frac{1}{N}\sum_{m=0}^{N-1} x_m,$$

  and are sometimes denoted the DC component of $\mathbf{x}$

- The c_k's (but sometimes also the X_k's) are called the *Fourier coefficients* of $\mathbf{x}$
- $|X_k|^2/N$ is the energy for the waveform $\mathbf{E}_{N,k}$
- Symmetry for real-valued signals:

$$\mathbf{x} \in \mathbb{R}^N \iff X_{-k} = \overline{X_k}$$

The sets of components x_m and X_k are representations of the signal in the *time domain* and *frequency domain*

Time domain: Basis $\mathbf{e}_m = (0, 0, \ldots, 0, 1, 0, \ldots, 0)$ with unity in the m-th place, sample taken at $t_m = mT/N$

$$\mathbf{x} = \sum_{m=0}^{N-1} x_m \mathbf{e}_m$$

Frequency domain: Basis $\mathbf{E}_{N,k}$, natural frequency $\omega = 2\pi k/T$

$$\mathbf{x} = \frac{1}{N} \sum_{k=0}^{N-1} X_k \mathbf{E}_{N,k}$$

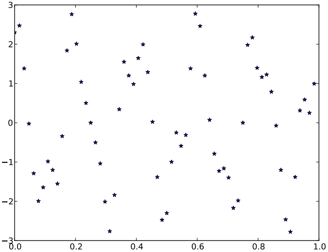
DFT examples

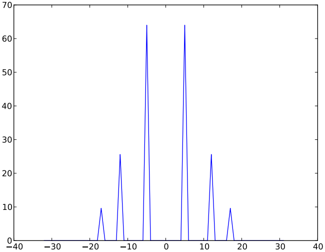
Aliasing

- The analog signal (as before) defined on $t \in [0, 1]$:

$$x(t) = 2\cos(2\pi \cdot 5t) + 0.8\sin(2\pi \cdot 12t) + 0.3\cos(2\pi \cdot 47t)$$

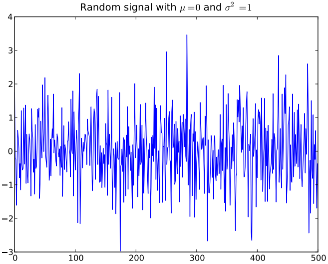
- Sampled at $\Delta T = 1/64\,\mathrm{s}$
- Nyquist frequency: 32 Hz
- Aliasing: $c_{47} = c_{64-47} = c_{17}$
- The 47 Hz energy masquerades as 17 Hz

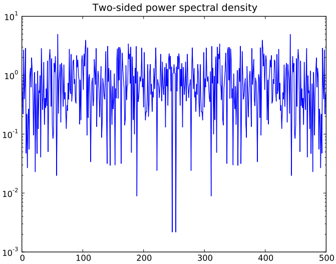




DFT examples
White Noise

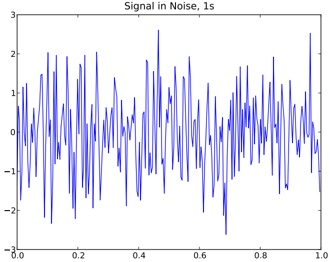
- A discrete signal $\mathbf{x} \in \mathbb{R}^N$ with each of the samples x_m being an independent normal variable with $\mu = 0$, $\sigma^2 = 1$
- $\Re(X_k)$ and $\Im(X_k)$ are both normal random variables
- The expectation value of $|X_k|^2/N$ is σ^2 for all k

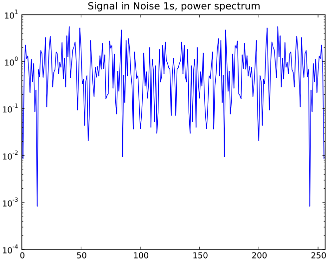




DFT examples
Signal in Noise

- Analog signal sampled at $\Delta T = 1/256\,\mathrm{s}$, $T = 1\,\mathrm{s}$
- $x_m = s_m + n_m$
- $s_m = 0.1\cos(2\pi f m \Delta T)$ with $f = 40\,\mathrm{Hz}$
- n_m is white Gaussian noise with $\sigma^2 = 1$
- σ^2 is $10\times$ the amplitude of the oscillation
- Periodic signal does not show up in the time domain nor in the power spectrum

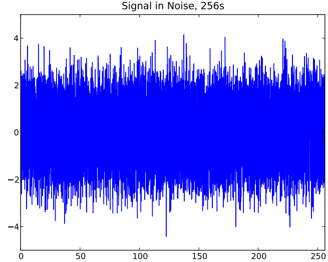


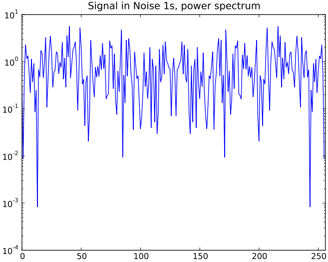


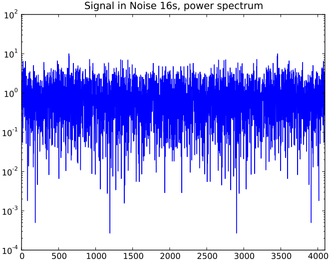
- How to detect the signal? It is audible!
- Energy per frequency bin due to white noise is $\epsilon_n = 2\sigma^2$, constant for all frequencies and independent of number of samples
- The energy of the oscillation is $\epsilon_s = 2N|c_{40\,\mathrm{Hz}}|^2 = N(0.1)^2 = N/100$
- By increasing the observation time T (and thus number of samples $N = T/\Delta T$, we improve the value of ϵ_s/ϵ_n (signal to noise ratio)

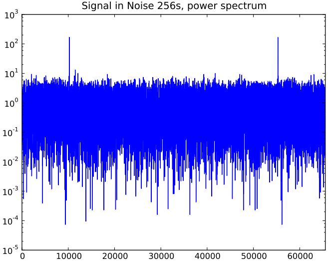
DFT examples

Signal in Noise









Matrix Formulation of the DFT

- The product $\mathbf{Ax}$ of an $M \times N$ matrix $\mathbf{A}$ and an N-dimensional vector $\mathbf{x}$ is an M-dimensional vector $\mathbf{v}$ with components

$$v_k = \sum_{m=0}^{N-1} A_{km} x_m$$

- Compare this to the DFT definition:

$$X_k = \sum_{m=0}^{N-1} e^{-2\pi ikm/N} x_m$$

- Thus, $\mathbf{X} = DFT(\mathbf{x})$ can be written as a product

$$\mathbf{X} = \mathbf{F}_N \mathbf{x}$$

$$F_N(k,m) = e^{-2\pi ikm/N}$$

- $\mathbf{F}_N$ is symmetric, $F_N(k, m) = F_N(m, k)$
- The inverse DFT $\mathbf{x} = \mathbf{F}_n^{-1}\mathbf{X}$ is given by

  $$\mathbf{F}_N^{-1} = \frac{1}{N}\overline{\mathbf{F}_N} = \frac{1}{N}\mathbf{F}_N^*$$

  where $\overline{\mathbf{F}_N}$ is the complex conjugate of $\mathbf{F}_N$, and $\mathbf{F}_N^*$ is the Hermitian transpose of $\mathbf{F}_N$ (conjugation and matrix transposition)
- DFT is linear:

  $$DFT(a\mathbf{x} + b\mathbf{y}) = a \cdot DFT(\mathbf{x}) + b \cdot DFT(\mathbf{y})$$

  for $\mathbf{x}, \mathbf{y} \in \mathbb{C}^N$ and $a, b \in \mathbb{C}$

- Let $z = e^{-2\pi i/N}$ (complex N-th root of unity). Then $F_N(k, m) = z^{km}$:

$$\mathbf{F}_N = \begin{pmatrix} 1 & 1 & 1 & 1 & \cdots & 1 \\ 1 & z & z^2 & z^3 & \cdots & z^{N-1} \\ 1 & z^2 & z^{2\cdot 2} & z^{2\cdot 3} & \cdots & z^{2\cdot(N-1)} \\ 1 & z^3 & z^{3\cdot 2} & z^{3\cdot 3} & \cdots & z^{3\cdot(N-1)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & z^{N-1} & z^{(N-1)\cdot 2} & z^{(N-1)\cdot 3} & \cdots & z^{(N-1)\cdot(N-1)} \end{pmatrix}$$

- Symmetries of $\mathbf{F}_N$ can be exploited to compute the DFT efficiently for large N without actual matrix multiplication

Matrix Formulation of the DFT

Example: $N = 4$

- $z = e^{-2\pi i/4} = -i$
- $z^2 = -1$, $z^3 = i$, $z^4 = 1$, $z^5 = -i$, etc.
- The DFT as a matrix operation:

$$\begin{pmatrix} X_0 \\ X_1 \\ X_2 \\ X_3 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{pmatrix} \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix}$$

- Note that $\mathbf{F}_4$ has only 4 distinct entries: $\pm i$ and $\pm i$

The Fast Fourier Transform, FFT

Introduction

- The FFT is a class of algorithms for computing the DFT and IDFT efficiently
- Exploits the regular structure of $\mathbf{F}_N$
  - $\mathbf{F}_N = \mathbf{F}_N^T$
  - $\mathbf{F}_N$ has only N distinct entries
- FFT is a 'divide and conquer' algorithm
  - Recursively divide a difficult problem into smaller subproblems of the same type
  - Solve the subproblem directly when simple enough
  - Combine the solutions to the subproblems
- Has many software implementations, e.g., FFTW, 'The Fastest Fourier Transform in the West'

- Number of floating point operations needed to compute

$$X_k = \sum_{m=0}^{N-1} e^{-2\pi ikm/N} x_m$$

  is

  - N complex multiplications,
  - $N - 1$ additions,

  i.e., $2N - 1$ operations
- Must be computed for all $k = 0, 1, \ldots, N - 1$
- Total operation count

$$(2N - 1)N = 2N^2 - N = \mathcal{O}(N^2)$$

- The FFT reduces the cost from $\mathcal{O}(N^2)$ to $\mathcal{O}(N \log N)$
- Significant saving!

n	$N = 2^n$	N^2	$N \log N$
10	1 024	1 048 576	10 240
12	4 096	16 777 216	49 152
14	16 384	268 435 456	229 376
16	65 536	4 294 967 296	1 048 576

- Performs best when N has no large prime factors
- Performs ideally for N being a power of two, $N = 2^n$

- Original problem: N-point DFT
- Split into two $N/2$-point DFTs instead
- Suppose N is even, split the sum into even and odd indices

$$
\begin{aligned}
X_k &= \sum_{m=0}^{N-1} x_m e^{-2\pi i k m/N} \\
&= \underbrace{\sum_{m=0}^{N/2-1} x_{2m} e^{-2\pi i k(2m)/N}}_{\text{even indices}} + \underbrace{\sum_{m=0}^{N/2-1} x_{2m+1} e^{-2\pi i k(2m+1)/N}}_{\text{odd indices}}
\end{aligned}
$$

The Fast Fourier Transform, FFT

The First $N/2$ components

- Let $k = k_0$ with $0 \leq k_0 < N/2$

$$
\begin{aligned}
X_{k_0} &= \sum_{m=0}^{N/2-1} x_{2m} e^{-2\pi i k_0 m/(N/2)} \\
&\quad + e^{-2\pi i k_0/N} \sum_{m=0}^{N/2-1} x_{2m+1} e^{-2\pi i k_0 m/(N/2)} \\
&= F_1(k_0) + e^{-2\pi i k_0/N} F_2(k_0)
\end{aligned}
$$

  where
  - $F_1(k_0)$ is the DFT of the vector $(x_0, x_2, x_4, \ldots, x_{N-2})$
  - $F_2(k_0)$ is the DFT of the vector $(x_1, x_3, x_5, \ldots, x_{N-1})$
- First half of $\mathbf{X}$: $\mathbf{F}_1 + \mathbf{M}\mathbf{F}_2$, where $\mathbf{M} = \text{diag}\{e^{-2\pi i k_0/N}\}$

The Fast Fourier Transform, FFT

The Last $N/2$ components

- Let $k = N/2 + k_0$ with $0 \leq k_0 < N/2$

$$
\begin{aligned}
X_{N/2+k_0} &= e^{-2\pi i m} \sum_{m=0}^{N/2-1} x_{2m} e^{-2\pi i k_0 m/(N/2)} \\
&\quad + e^{-2\pi i m} e^{-\pi i} e^{-2\pi i k_0/N} \sum_{m=0}^{N/2-1} x_{2m+1} e^{-2\pi i k_0 m/(N/2)} \\
&= F_1(k_0) - e^{-2\pi i k_0/N} F_2(k_0)
\end{aligned}
$$

- Last half of $\mathbf{X}$: $\mathbf{F}_1 - \mathbf{M}\mathbf{F}_2$

The N-point DFT is reduced to the computation and combination of two $N/2$-point DFTs

- Let W_N be the cost of computing the full N-point FFT
- Recursive equation

$$W_N = \underbrace{2W_{N/2}}_{\text{2 FFTs}} + \underbrace{2N}_{\text{adding the results}}$$

  with $W_1 = 0$
- Explicit formula

$$W_N = 2N \log N$$

The Two-Dimensional DFT

- Consider $m \times n$ matrix $\mathbf{A}$ (e.g., a rectangular image)
- The generalization of a 1D basic waveform $\mathbf{E}_k$ is a matrix $\mathcal{E}_{k,l}$ with entries

$$\mathcal{E}_{k,l}(r,s) = e^{2\pi ikr/m}e^{2\pi ils/n} = e^{2\pi i(kr/m+ls/n)}$$

- These waveforms are orthogonal in $M_{m,n}(\mathbb{C})$ with respect to the usual inner product
- $\mathcal{E}_{k,l}$ for $0 \leq k \leq m-1$ and $0 \leq l \leq n-1$ form an orthogonal basis for $M_{m,n}(\mathbb{C})$

- Let $\mathbf{A} \in M_{m,n}(\mathbb{C})$ have components a_{rs}
- The two-dimensional DFT of $\mathbf{A}$ is the matrix $\hat{\mathbf{A}} \in M_{m,n}(\mathbb{C})$ with components

$$\hat{a}_{k,l} = (\mathbf{A}, \mathcal{E}_{k,l}) = \sum_{r=0}^{m-1} \sum_{s=0}^{n-1} a_{r,s} e^{-2\pi i(kr/m + ls/n)}$$

  where $0 \leq k \leq m-1$ and $0 \leq l \leq n-1$
- Straightforward generalisation of the 1D DFT
- Aliasing: $\hat{a}_{k,l} = \hat{a}_{k+pm,l+qn}$ for $p, q \in \mathbb{Z}$

- Let $\hat{\mathbf{A}} \in M_{m,n}(\mathbb{C})$ have components $\hat{a}_{rs}$
- The inverse two-dimensional DFT of $\hat{\mathbf{A}}$ is the matrix $\mathbf{A} \in M_{m,n}(\mathbb{C})$ with components

$$a_{k,l} = \frac{1}{mn} \sum_{r=0}^{m-1} \sum_{s=0}^{n-1} \hat{a}_{r,s} e^{2\pi i(kr/m + ls/n)}$$

  where $0 \leq k \leq m-1$ and $0 \leq l \leq n-1$
- Thus, the orthogonal decomposition of the image $\mathbf{A}$ becomes

$$\mathbf{A} = \frac{1}{mn} \sum_{r=0}^{m-1} \sum_{s=0}^{n-1} \hat{a}_{k,l} \mathcal{E}_{k,l}$$

The Two-Dimensional DFT

Matrix View

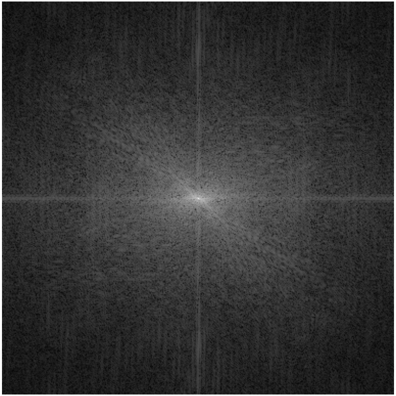
- The two-dimensional DFT can be computed as

$$\hat{\mathbf{A}} = \mathbf{F}_m \mathbf{A} \mathbf{F}_n^T$$

- The operation $\mathbf{A} \to \mathbf{F}_m \mathbf{A}$ performs a one-dimensional m-point DFT on each column of $\mathbf{A}$
- The operation $\mathbf{F}_m \mathbf{A} \to \mathbf{F}_m \mathbf{A} \mathbf{F}_n^T$ performs a one-dimensional n-point DFT on each row of $\mathbf{F}_m \mathbf{A}$
- The two-dimensional DFT can be computed as a sequence of one-dimensional DFTs

The Two-Dimensional DFT

Example



Do the following exercises in the text book: 2.2, 2.8, 2.9, 2.10