ELIC 629
Digital Image Processing

Fall 2005

The 1D Fourier Transform & its Properties,
The 2D Fourier Transform and its Properties
Filtering in the Frequency Domain

Bill Kapralos

Monday, November 24 2005

Overview (1):

- **Before We Begin**
  - Administrative details
  - Review → some questions to consider
- **The 1D Discrete Fourier Transform**
  - Introduction
  - Properties
- **The 2D Discrete Fourier Transform**
  - Introduction
  - The two-dimensional Fourier transform and its inverse

Overview (2):

- **Filtering in the Frequency (Fourier) Domain (time permitting…)**
  - Properties of the frequency domain
  - Convolution Theorem
    - The Frequency vs. the spatial Domain (this is very important!)

Before We Begin

Administrative Details (1):

- **Lab Six Today**
  - No assignment
  - Lab report required
  - Continued from last week
  - Requires the use of Matlab
    - No camera required

Some Questions to Consider (1):

- What is the Laplacian operator ?
- How do compute the Laplacian of an image ?
- What is the Fourier domain ?
- Who first introduced the concept of the Fourier domain ?
- What is the difference between a Fourier series and the Fourier transform ?
- What is the discrete Fourier transform (DFT) ?
- After converting a spatial signal to the frequency domain, can we go back to the spatial domain without any loss of info. ?

The One-Dimensional Fourier Transform

Introduction (1):

- **Originally, Fourier Transform was Formulated with Continuous Time Signals**
  - We are dealing with sampled images
  - Finite intensity values and finite in duration
  - In other words, we are dealing with a discrete signal → remember, an image itself is a signal as in your DSP course, except we are now dealing with a two-dimensional signal as opposed to a one-dimensional signal you are familiar with
  - Discrete Fourier Transform (DFT) introduced to handle discrete signals

Discrete Fourier Transform (1):

- **One of the Most Common and Powerful Procedures Encountered in the Field of Digital Signal Processing in General**
  - Enables us to analyze, manipulate and synthesize signals in ways not possible with continuous (analog) signal processing
  - Used in every field of engineering
  - A solid understanding of the DFT is extremely important!

Discrete Fourier Transform (2):

- **What is the Discrete Fourier Transform ?**
  - A mathematical procedure used to determine the frequency (or harmonic) content of a discrete signal
    - Remember → discrete signal obtained by periodically sampling a continuous time signal in the time domain
  - Based on the Continuous Fourier Transform (CFT), denoted by X(f) (or F(u))

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt$$

Discrete Fourier Transform (3):

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}dt$$

- **Lets Analyze This Expression:**
  - f → frequency (spectral component)
  - x(t) → continuous time domain signal
  - $e^{-j2\pi ft}$ → a sinusoid (sine wave) of frequency f
  - In words → Fourier Transform of frequency component f is a correlation of the infinite input signal at each time step with a sine wave of frequency f → X(f) tells us "how much" of the sine wave of frequency f the signal contains

Discrete Fourier Transform (4):

- **Discrete Fourier Transform (DFT) Mathematically**

$$X[m] = \frac{1}{M}\sum_{n=0}^{M-1} x[n]e^{-j2\pi nm/N}$$

  - Using Euler's Relationship $e^{-j\theta} = \cos(\theta) - j\sin(\theta)$ we obtain:

$$X[m] = \frac{1}{M}\sum_{n=0}^{M-1} x[n](\cos(2\pi nm/N) - j\sin(2\pi nm/N))$$

Discrete Fourier Transform (5):

$$X[m] = \frac{1}{M}\sum_{n=0}^{M-1} x[n](\cos(2\pi nm/N) - j\sin(2\pi nm/N))$$

- X[m] → mth DFT output e.g., X[0], X[1] … X[M-1]
- m → index of the DFT output in frequency domain (m = 0, 1, 2, … M-1)
- x[n] → sequence of input (discrete) samples (x[0], x[1], x[2] … x[M-1])
- n → (discrete) time domain index of input samples
- j = sqrt(-1) (remember, complex numbers!)
- M → number of samples (same for input and DFT)

Discrete Fourier Transform (6):

- **Some Notes Regarding the DFT**
  - Indices for input samples and DFT output samples always go from 0 to M-1
    - With N input time domain samples, the DFT determines the spectral content of the input at N equally spaced frequency points
    - N is an important parameter and determines
      1. How many input samples are needed
      2. Resolution of the frequency domain results
      3. Amount of processing time required to calculate an M-point DFT

Discrete Fourier Transform (7):

- **Some Notes Regarding the DFT (cont…)**
  - In words:
    - Each X[m] DFT output is the sum of a *point for point* product between an input sequence of input values and a complex sinusoid of the form cos(θ) – jsin(θ)
    - Exact frequencies of the of the different sinusoids depend on sample rate f_s and number of samples M
    - Fundamental frequency of the sinusoids is f_s /M and all other X[m] analysis frequencies are integer multiples of the fundamental!

Discrete Fourier Transform (8):

- **Some Notes Regarding the DFT (cont…)**
  - The N separate DFT analysis frequencies are

$$f_{analysis}(m) = \frac{mf_s}{M}$$

  - So, X[0] gives us magnitude of an 0Hz ("DC") component contained in the signal, X[1] gives us magnitude of the fundamental component, X[2] gives us magnitude of 2 x fundamental component contained in signal etc.
  - Finally, keep in mind, we are dealing with complex sinusoids → magnitude and phase!

Discrete Fourier Transform (9):

- **Determining the Magnitude and Phase Contained in each X[m] Term**
  - We can represent an arbitrary DFT output value X[m] by its real and imaginary parts

$X[m] = X_{real}[m] + jX_{imag}[m] = X_{mag}[m]$ at angle of X_0[m]

  - The magnitude of X[m] is

$$X_{mag}[m] = |X[m]| = \sqrt{X_{real}[m]^2 + X_{imag}[m]^2}$$

Discrete Fourier Transform (10):

- **Determining the Magnitude and Phase Contained in each X[m] Term (cont…)**
  - The phase angle of X[m], X_Θ[m] is

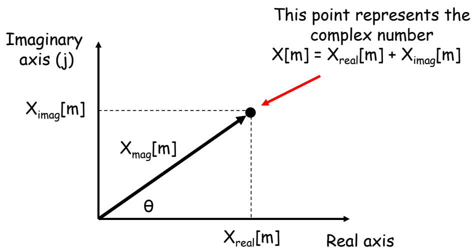
$$X_\Theta[m] = \tan^{-1}\left(\frac{X_{imag}[m]}{X_{real}[m]}\right)$$

  - The power of X[m], known as the power spectrum or spectral power is the magnitude squared

$$X_{PS}[m] = X_{mag}[m]^2 = X_{real}[m]^2 + X_{imag}[m]^2$$

Discrete Fourier Transform (11):

- **Graphical Illustration of Phase and Magnitude (Complex Plane)**



Some Properties of the 1D DFT

DFT Symmetry (1):

- **Symmetry in DFT Output is Obvious!**
  - Standard DFT is designed to accept complex input but most physical DFT inputs are "real" inputs
    - Non-zero real sample values
    - Imaginary values are assumed to be zero
  - With "real" input x[n] the complex DFT outputs for n = 1 to n = (M/2) – 1 are redundant with frequency output values for m > (M/2)
    - mth DFT output will have the same value as the (M-m)th DFT output
    - the phase angle of the mth output is the negative of the (M-m)th DFT output

DFT Symmetry (2):

- **Symmetry in DFT Output is Obvious! (cont…)**
  - What does this symmetry mean?
    - If we perform an M-point DFT on a real input sequence, we get M separate complex DFT output terms but only the first M/2 terms are independent
    - To obtain DFT of x[n], we need only compute the first M/2 values of X[m] where $0 \leq m \leq (M/2)-1$
    - The X[M/2] to X[M-1] DFT output terms provide no additional information about the spectrum of the real sequence x[n]

DFT Linearity (3):

- **DFT is Linear**
  - The DFT of the sum of two signals is equal to the sum of the transforms of each signal
    - Let $x_1[n]$ and $x_2[n]$ be two discrete input signals with DFT $X_1[m]$ and $X_2[n]$ respectively
    - Consider the sum of these two signals

$$x_{sum}[n] = x_1[n] + x_2[n]$$

    - The DFT of $x_{sum}[n]$ is

$$X_{sum}[m] = X_1[m] + X_2[n]$$

DFT Linearity (4):

- **DFT is Linear (cont…)**
  - Exercise:
    - Mathematically prove this linearity property for the DFT

Inverse DFT

Inverse DFT – IDFT (1):

- **Reverse the DFT Process**
  - DFT transforms time-domain data into frequency domain representation
  - With inverse DFT, we transform frequency domain representation into time-domain representation
    - Perform IDFT on X[m] frequency domain values

$$x[n] = \sum_{n=0}^{M-1} X[m](\cos(2\pi nm/N) - j\sin(2\pi nm/N))$$

Introduction to the Two-Dimensional Fourier Transform

Introduction (1):

- **Straightforward to Extend One-Dimensional DFT to Two Dimensions**
  - Two-dimensional DFT of a function (image) f(x,y) of size M x N is given by

$$F[u,v] = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f[x,y]e^{-j2\pi(ux/M+vy/N)}$$

  - Using Euler's relationship, we have the following

$$F[u,v] = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f[x,y](\cos(-j2\pi(ux/M+vy/N))+j\sin(-j2\pi(ux/M+vy/N)))$$

Introduction (2):

- **Straightforward to Extend One-Dimensional DFT to Two Dimensions (cont...)**
  - We can also easily extend the IDFT to two-dimensions as well. Given F[u,v], IDFT is

$$f[x,y] = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F[u,v]e^{-j2\pi(ux/M+vy/N)}$$

  - Using Euler's relationship, we have the following

$$f[x,y] = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F[u,v](\cos(-j2\pi(ux/M+vy/N))+j\sin(-j2\pi(ux/M+vy/N)))$$

Introduction (3):

- **Some Notes About 2D DFT**
  - x = 0, 1, 2, ..., M-1 and y =0, 1, 2, ..., N-1
  - Variables u and v are the *transform* or *frequency* variables and x, y are the *spatial* or *image* variables
  - As with 1D DFT, we can define the magnitude, phase and power spectrum in a similar manner
  - Magnitude

$$|F[u,v]| = \sqrt{R[u,v]^2 + I[u,v]^2}$$

Introduction (4):

- **Some Notes About 2D DFT (cont…)**
  - Phase φ[u,v]

$$|\phi[u,v]| = \tan^{-1}\left[\frac{I[u,v]}{R[u,v]}\right]$$

  - Power spectrum P[u,v]

$$P[u,v] = |F[u,v]|^2 = R[u,v]^2 + I[u,v]^2$$

  - where R[u,v] and I[u,v] are the real and imaginary components of the DFT F[u,v] respectively

Introduction (5):

- **Some Notes About 2D DFT (cont…)**
  - Typically we multiply input image by $(-1)^{x+y}$ (pixel-by-pixel multiplication) prior to computing the DFT
    - Shifts the origin of the DFT to frequency coordinates (M/2, N/2) → the center of the M x N 2D DFT
    - M, N → even integers
  - After the multiplication, the DFT becomes

$$F[u,v] \quad = \quad \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}(f[x,y]e^{-j2\pi(ux/M+vy/N)})(-1)^{x+y}$$

Introduction (6):

- **Some Notes About 2D DFT (cont…)**
  - Which is equal to

$$F[u,v] \quad = \quad \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}(f[x,y]e^{-j2\pi(ux/M+vy/N)})(-1)^{x+y} = F(u-M/2, v-N/2)$$

  - When we implement 2D DFT summations run from u = 1 to M and v = 1 to N.
  - The center of the transform is at u = (M/2) + 1 and v = (N/2) +1

Introduction (7):

- **DC Component**
  - DFT at the origin (0,0) in the frequency domain is equal to the average gray level (intensity) of image f(x,y)

$$F[0,0] = \frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)$$

Some 2D DFT Relationships (1):

- **Conjugate Symmetry**
  - If image f(x,y) is real, its Fourier transform is conjugate symmetric

$$F(u, v) = F^*(-u, -v)$$

    - where "*" indicates standard conjugate operation on a complex number
  - This implies the spectrum of the Fourier transform is symmetric

$$|F(u, v)| = |F(-u, -v)|$$

Some 2D DFT Relationships (2):

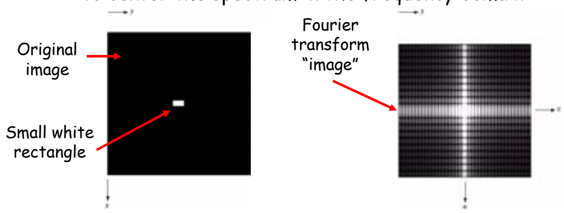
- **Conjugate Symmetry (cont…)**
  - Conjugate symmetry and centering property simplify the specification of circularly symmetric filters in the frequency domain
- **Relationship Between Samples in the Frequency and Spatial Domains**

$$\Delta u = 1/(M\,\Delta x) \text{ and } \Delta u = 1/(N\,\Delta y)$$

  - In other words → inverse relationship between spatial and frequency domain resolution

2D DFT Example (1):

- **2D DFT of a "Simple" Image**
  - 20 x 40 rectangle superimposed on black background of size 512 x 512
  - Image multiplied by $(-1)^{x+y}$ prior to computing DFT to center the spectrum in the frequency domain



2D DFT Example (2):

- **Some Comments regarding the Example**
  - Inverse spatial vs. frequency relationship
    - Separation of "spectrum zeros" in u direction is twice separation in v direction → 1 to 2 size ratio of rectangle in the image
  - Spectrum was processed using log transform prior to displaying to enhance gray level
    - Recall, dynamic range of DFT is huge and if we didn't process it, little detail would be evident
    - Most DFT spectra are processed with the log transform prior to displaying