### Fourier Transform

**Outline**
- What is Fourier transform
- 1D Fourier transform
  - Properties of Fourier transform
  - Discrete Fourier transform
- 2D discrete Fourier transform
  - Properties of 2D transform
  - Fast Fourier transform

---

**Fourier Theory**
- Proposed by French mathematician Fourier in 1807
- Any periodic function can be expressed as the sum of sines and/or cosines of different frequencies, each multiplied by a different coefficient
- Allows frequency content (spectral) analysis of a signal

**Fourier Transform**
- Fourier transform:
  - Converts a periodic function to a coefficient function of frequencies
  - The original periodic function can be fully reconstructed using the coefficients
- Inverse Fourier transform
  - Conversion from coefficients to the original function

\[
F(u) = \int_{-\infty}^{\infty} f(x) \cos(2\piux) \, dx - j \int_{-\infty}^{\infty} f(x) \sin(2\piux) \, dx
\]

\[
\Phi(u) = \frac{1}{N} \sum_{x=0}^{N-1} F(u) e^{-j2\piux/N}
\]

---

**Polar Representation**
- Fourier transform can be expressed in polar coordinates:
  - \( F(u) = R(u) + jI(u) = |F(u)| e^{j\Phi(u)} \)
  - \( |F(u)| = \sqrt{R^2(u) + I^2(u)} \), \( \Phi(u) = \tan^{-1} \frac{I(u)}{R(u)} \)
  - \( R(u) \): real parts of \( F(u) \)
  - \( I(u) \): imaginary parts of \( F(u) \)
  - \( |F(u)| \): magnitude (or spectrum)
  - \( \Phi(u) \): phase angle (or phase spectrum)
  - \( P(u) = |F(u)|^2 = R^2(u) + I^2(u) \)
  - \( P(u) \): power spectrum

---

**Discrete Fourier Transform**
- For a complex series \( f[x] \) with \( N \) samples:
  - Assume that the series outside range \([0, N-1]\) is extended \( N \)-periodic
  - \( f[x] = f[x-N] \)
  - Fourier transform of the series also has \( N \) samples
  - Transform between the 2 series are called discrete Fourier transform (DFT)

\[
F[u] = \frac{1}{N} \sum_{x=0}^{N-1} f[x] \left( \frac{\cos(2\piux/N) + j\sin(2\piux/N)}{2} \right)
\]

\[
\Phi[u] = \frac{1}{N} \sum_{x=0}^{N-1} F[u] e^{-j2\piux/N}
\]
Properties of DFT

- 1st sample of the transformed series is the sum of input series
  \[ F[0] = \sum_{n=0}^{N-1} f[n] \]
- Called direct current (DC) component
- The transform of a constant function is a DC value only
- The transform of a delta function is a constant

Danielson-Lanczos Lemma

- DFT of a sequence can be calculated from the DFTs of 2 subsequences at odd & even locations
  \[
  F[u] = \sum_{k=0}^{N/2-1} f[2k] e^{-j2\pi uk/N} + \sum_{k=0}^{N/2-1} f[2k+1] e^{-j2\pi u(k+1)/N} = F_{\text{even}}[u] + F_{\text{odd}}[u] e^{-j\pi u/N}
  \]
- Used for deriving Fast Fourier Transform (FFT)
  - Recursively split the sequence by half each time
  - Require the image size to be the power of 2
  - Reduces complexity to \( O(N \log N) \)

Fast Fourier Transform (FFT)

- Base case:
  - When sequence has only 1 value: \( F[0] = f[0] \)
- Recursive case:
  - Reorder the sequence & split it by half
  - Decimation in time (DIT) or in frequency (DIF)
  - Process each half recursively
  - Apply Danielson-Lanczos Lemma

Properties of DFT (Cont'd)

- DFT of a real function is conjugate symmetric
- Spectrum of the Fourier transform is symmetric
- Inverse transform can be computed using forward transform:
  - Use the conjugate of \( F[u] \) as input
  - Take the conjugate of the output & divide the result by \( N \)

FFT Implementation

- void reorder(complex f[], int n) {
  \[
  \begin{align*}
  \text{complex b[]} & = \text{new complex}[n/2]; \\
  \text{for (int i=0; i<n/2; i++)} & \\
  \text{\quad b[i] = f[i+n/2];} \\
  \text{\quad for (int i=0; i<n/2; i++)} & \\
  \text{\quad f[i+n/2] = b[i];} \\
  \end{align*}
  \]
  \}
- void fft2(complex X[], int N) {
  \[
  \begin{align*}
  \text{if (N < 2)} & \\
  \text{\quad return;} & \\
  \text{\quad reorder(X,N);} & \\
  \text{\quad fft2(X, N/2);} & \\
  \text{\quad fft2(X+N/2, N/2);} & \\
  \text{\quad X[k] = w * X[k] + w^* * a;} & \\
  \text{\quad X[k+N/2] = w * X[k] - w^* * a;} & \\
  \end{align*}
  \]
}
2D Fourier Transform

- Continuous definition:
  - \( F(u, v) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-j2\pi ux/M} e^{-j2\pi vy/N} dx \cdot dy \)
  - \( f(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(u, v) e^{j2\pi ux/M} e^{j2\pi vy/N} du \cdot dv \)
  - \( u \) & \( v \): the transform (or frequency) variables
  - \( x \) & \( y \): the spatial (or image) variables

- Discrete form:
  - \( F[u, v] = \frac{1}{M} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f[x, y] e^{-j2\pi ux/M} e^{-j2\pi vy/N} \)
  - \( F[x, y] = \frac{1}{N} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F[u, v] e^{j2\pi ux/M} e^{j2\pi vy/N} \)
  - \( F[0, 0] \) is the sum of the input (DC component)

Translate Origin

- Based on the following translation properties:
  - \( f[x, y] e^{j2\pi (ux + vy)/N} \leftrightarrow F[u - u_0, v - v_0] \)
  - \( f[x - x_0, y - y_0] \leftrightarrow F[u, v] e^{-j2\pi (ux_0 + vy_0)/N} \)
  - Multiply the input image by \((-1)^{uxy}\) moves the origin of its Fourier transform from \((0, 0)\) to \(\left(\frac{M}{2}, \frac{N}{2}\right)\)
  - \( f[x, y] (-1)^{uxy} \leftrightarrow F[u - M/2, v - N/2] \)
  - This is a common practice to show low frequency at the center of the Fourier transform

Example of 2D DFT

Periodicity & Conjugate Symmetric Properties

- Both the Fourier transform and the inverse transform have periodicity properties
  - \( f[x, y] = f[x + M, y] = f[x, y + N] = f[x + M, y + N] \)
  - When the input are real numbers, such as a 2D image:
    - The Fourier transform is conjugate symmetric
    - \( f[u, v] = F[-u, -v] \)
    - The spectrum of the Fourier transform is symmetric
    - \( |F[u, v]| = |F[-u, -v]| \)

Computing Inverse Transform Using Forward Algorithm

- Similar to the 1D case, the inverse Fourier transform can be computed using forward transform algorithm:
  - Input the complex conjugate of \( F(u) \) into forward transform algorithm
  - Take the complex conjugate of the output
  - Divide the result by \( M \times N \)
  - \( f[x, y] = \frac{1}{M} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F[u, v] e^{j2\pi ux/M} e^{j2\pi vy/N} \) \( \Rightarrow \)
  - \( f'[x, y] = \frac{1}{N} \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} F'[u, v] e^{-j2\pi ux/M} e^{-j2\pi vy/N} \)

Separability Property

- 2D DFT is separable:
  - First apply 1D transform to each row in the input image, then apply 1D transform to each column in the previous result
  - \( F[u, v] = \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} f[x, y] e^{-j2\pi ux/M} e^{-j2\pi vy/N} \) \( \Rightarrow \)
  - \( \sum_{u=0}^{M-1} \sum_{v=0}^{N-1} f'[x, y] e^{j2\pi ux/M} e^{j2\pi vy/N} \) \( \Rightarrow \)
  - Multiply both FFT & separability reduces complexity from \( O(M^2 \times N^2) \) to \( O(MN \log M + \log N) \)