DIFFERENCES BETWEEN ONE AND MULTI DIMENSIONAL SIGNAL PROCESSING

• More data for M-D signal processing.
  1. 1-D Speech → 10K samples per second.
  2. M-D Television → 500 × 500 pixels per frame, 30 frames a second, 7.5 Mega samples per second.

• Mathematics for M-D is not as complete as 1-D:
  1. 1-D systems are described by differential equations, M-D by partial differential equations.
  2. Fundamental theorem of algebra does not hold in M-D, but holds in 1-D → Factorability of polynomials in 1-D is guaranteed, but not in higher dimensions.
  3. This affects filter design, IIR filter stability, signal reconstruction, etc.

• Causality.
SEQUENCES

- Notation for 2-D sequences: \( x(n_1, n_2) \).
- Unit sample sequence:
  \[
  \delta(n_1, n_2) = \begin{cases} 
  1 & n_1 = n_2 = 0 \\
  0 & \end{cases}
  \]
  (1)
- Line impulse:
  \[
  \delta(n_1) = \begin{cases} 
  1 & n_1 = 0 \\
  0 & \end{cases}
  \]
  (2)
- Line impulse \( \delta(n_2) \) defined similarly.
- \( \delta(n_1 - n_2) \) is 1 along \( n_1 = n_2 \).
- Step Sequence:
  \[
  u(n_1, n_2) = u(n_1)u(n_2) = \begin{cases} 
  1 & n_1 \geq 0 \quad n_2 \geq 0 \\
  0 & \end{cases}
  \]
  (3)
- \( u(n_1) \) and \( u(n_2) \) and \( u(n_1 - n_2) \) defined similarly.
- Exponential sequences: \( a^{n_1}b^{n_2} \).
SEQUENCES (cont’d)

• **Definition:** Seperable sequences \( x(n_1, n_2) \) are those that can be written as the product \( x_1(n_1)x_2(n_2) \).

• Separable sequences are important because a large number of 1-D results can be applied to systems with separable response.

• Examples:
  1. \( \delta(n_1, n_2) = \delta(n_1)\delta(n_2) \).
  2. \( u(n_1, n_2) = u(n_1)u(n_2) \)
  3. \( a^{n_1}b^{n_2} + a^{n_1} + n_2 = a^{n_1}(b^{n_2} + a^{n_2}) \).

• Periodic sequences:
  \[
  x(n_1, n_2) = x(n_1 + N_1, n_2) = x(n_1, n_2 + N_2)
  \]
  (4)

• Every sequence \( x(n_1, n_2) \) can be expressed as:
  \[
  x(n_1, n_2) = \sum_{k_1=-\infty}^{+\infty} \sum_{k_2=-\infty}^{+\infty} x(k_1, k_2)\delta(n_1-k_1, n_2-k_2)
  \]
  (5)
SYSTEMS

• Transformation of the input signal \( x(n_1, n_2) \) into the output signal \( y(n_1, n_2) \):

\[
T[x(n_1, n_2)] = y(n_1, n_2) \quad (6)
\]

• Linearity:

\[
T[ax_1(n_1, n_2) + bx_2(n_1, n_2)] = aT[x_1(n_1, n_2)] + bT[x_2(n_1, n_2)] \quad (7)
\]

• System is Shift Invariance if

\[
T[x(n_1, n_2)] = y(n_1, n_2) \quad (8)
\]

implies:

\[
T[x(n_1 - k_1, n_2 - k_2)] = y(n_1 - k_1, n_2 - k_2) \quad (9)
\]

• LINEAR SHIFT INVARIANT (LSI) SYSTEMS ARE OF UTMOST IMPORTANCE.
LSI SYSTEMS

- LSI systems can be uniquely specified by their *impulse response*. That is:
  \[ T[\delta(n_1,n_2)] = h(n_1,n_2) \]

- Knowing the impulse response enables one to determine the output uniquely for any given input.

- Input/output relationship for an LSI system with impulse response \( h(n_1,n_2) \) is given by the Convolution sum:
  \[
  y(n_1,n_2) = \sum_{k_1} \sum_{k_2} x(k_1,k_2) T[\delta(n_1 - k_1, n_2 - k_2)] \\
  \sum_{k_1} \sum_{k_2} x(k_1,k_2) h(n_1 - k_1, n_2 - k_2) 
  \]
  \[ (10) \]

- Notation for convolution:
  \[
  y(n_1,n_2) = x(n_1,n_2) \ast h(n_1,n_2) 
  \]
  \[ (11) \]

- An example of convolution.
SEPARABLE SYSTEMS

• A separable system is an LSI system whose impulse response is separable:

\[ h(n_1, n_2) = h_1(n_1)h_2(n_2) \quad (12) \]

• Consider the number of multiplies involved in convolving an \( N \times N \) input sequence \( x(n_1, n_2) \) with an \( M \times M \) impulse response of an LSI system \( h(n_1, n_2) \). Assume \( M << N \).

1. If \( x \) and \( h \) are not separable, then total number of multiplies goes as \( M^2N^2 \).
2. If \( h \) is separable, then the number of multiplies goes as \( 2MN^2 \).
3. If both \( h \) and \( x \) are separable, then we need \( N^2 \) multiplies.
Figure 1.15 Convolution of $x(n_1, n_2)$ with a separable sequence $h(n_1, n_2)$. 
Figure 1.16 Example of convolving $x(n_1, n_2)$ with a separable sequence $h(n_1, n_2)$. 
STABILITY

- System is BIBO stable if all bounded inputs result in bounded outputs.
- Hard to show stability for general systems.
- Can show that necessary and sufficient condition for BIBO stability of LSI system is absolute summability of the impulse response.

\[
\sum_{n_1} \sum_{n_2} |h(n_1, n_2)| < \infty \quad (13)
\]
Special Support Systems

- First quadrant sequence is one whose non zero values are in the first quadrant.
- First quadrant system is one whose impulse response is a first quadrant sequence.
- Wedge support system is one whose impulse response has a wedge support.

- Linear transformation of variables can be used to convert a wedge support system into a first quadrant system.
  - Important since checking stability of wedge support system reduces to that of FQ systems.
For a point along \( N_2 \):

**Example:**

Let \( m_1 = n_1 \)

\[
\begin{align*}
m_1 &= N_2 = N_{12} = N_{12} = 0 \\
m_2 &= -N_2 \cdot N_{12} + N_{11} \cdot N_{22}
\end{align*}
\]
Without loss of generality assume no common factor between \( N_{i1} \) and \( N_{i2} \).

Also \( \nu = \nu_{21} N_{21} \) and \( \nu_{22} \).

Since \( N_{i1} \) and \( N_{i2} \) not in the same direction,

\[
D = N_{i1} N_{22} - N_{i2} N_{21} \neq 0
\]

Do a change of variable

\[
m_1 = N_{22} n_1 - N_{12} n_2
\]

\[
m_2 = -N_{21} n_1 + N_{11} n_2
\]

For a point along \( N_{i1} \) we let

\[
n_1 = N_{i1}, \quad n_2 = N_{i2} \text{ in }
\]

we get

\[
m_1 = N_{22} N_{i1} - N_{12} N_{21}
\]

\[
m_2 = -N_{21} N_{i1} + N_{11} N_{21} = 0
\]
Fourier Transform (cont'd)

- More generally, Discrete Time Fourier Transform (DTFT) of a sequence \( x(n_1, n_2) \) is given by:

\[
X(\omega_1, \omega_2) = \sum_{n_1} \sum_{n_2} x(n_1, n_2) e^{-j\omega_1 n_1} e^{-j\omega_2 n_2} \quad (14)
\]

- Observations:

1. \( X(\omega_1, \omega_2) \) is a complex function, has a real and imaginary part.
2. It is a continuous function of \( \omega_1 \) and \( \omega_2 \).
3. It is doubly periodic with period \( 2\pi \) in \( \omega_1 \) and \( \omega_2 \).
4. If \( x(n_1, n_2) \) is separable, so is its Fourier Transform

\[
x(n_1, n_2) = x_1(n_1)x_2(n_2)
\]

\[
X(\omega_1, \omega_2) = X_1(\omega_1)X_2(\omega_2) \quad (15)
\]

- Inverse Fourier Transform:

\[
x(n_1, n_2) = \left(\frac{1}{2\pi}\right)^2 \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} X(\omega_1, \omega_2) e^{j\omega_1 n_1} e^{j\omega_2 n_2} d\omega_1 d\omega_2 \quad (16)
\]
Properties of Fourier Transform

- linearity.
  \[ ax_1 + bx_2 \leftrightarrow aX_1 + bX_2 \]

- Spatial Shift:
  \[ x(n_1-m_1, n_2-m_2) \leftrightarrow e^{-j\omega_1m_1-j\omega_2m_2}X(\omega_1, \omega_2) \]

- Multiplication in time corresponds to convolution in frequency domain:
  \[ c(n_1, n_2)x(n_1, n_2) \leftrightarrow \]
  \[ \left(\frac{1}{2\pi}\right)^2 \int_{-\pi}^{\pi} X(\theta_1, \theta_2)C(\omega_1-\theta_1, \omega_2-\theta_2)d\theta_1d\theta_2 \]

- Differentiation in frequency domain corresponds to multiplication by \( j \) in the space domain:
  \[ -jn_1x(n_1, n_2) \leftrightarrow \partial X(\omega_1, \omega_2)/\partial \omega_1 \]
  \[ -jn_2x(n_1, n_2) \leftrightarrow \partial X(\omega_1, \omega_2)/\partial \omega_2 \]
Properties of Fourier Transform (cont'd)

- Transposition:
  \[ x(n_2, n_1) \leftrightarrow X(\omega_2, \omega_1) \]

- Reflection:
  \[ x(-n_1, n_2) \leftrightarrow X(-\omega_1, \omega_2) \]
  \[ x(n_1, -n_2) \leftrightarrow X(\omega_1, -\omega_2) \]
  \[ x(-n_1, -n_2) \leftrightarrow X(-\omega_1, -\omega_2) \]

- Complex Conjugate:
  \[ x^*(n_1, n_2) \leftrightarrow X^*(-\omega_1, -\omega_2) \]

- Real and Imaginary Parts:
  \[ Re[x] \leftrightarrow \frac{1}{2}[X(\omega_1, \omega_2) + X^*(-\omega_1, -\omega_2)] \]
  \[ Im[x] \leftrightarrow \frac{1}{2}[X(\omega_1, \omega_2) - X^*(-\omega_1, -\omega_2)] \]
  \[ \frac{1}{2}[x(n_1, n_2) + x^*(-n_1, -n_2)] \leftrightarrow Re[X(\omega_1, \omega_2)] \]
  \[ \frac{1}{2}[x(n_1, n_2) - x^*(-n_1, -n_2)] \leftrightarrow Im[X(\omega_1, \omega_2)] \]
Properties of Fourier Transform (cont'd)

- Parseval's theorem:

\[ \sum_{n_1} \sum_{n_2} x(n_1, n_2) w^*(n_1, n_2) = \]
\[ (\frac{1}{2\pi})^2 \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} X(\omega_1, \omega_2) W^*(\omega_1, \omega_2) d\omega_1 d\omega_2 \]

- Examples of Fourier Transform.
and along the $\omega_1$ and $\omega_2$ axes. One reason for the energy concentration near the origin is that images typically have large regions where the intensities change slowly. Furthermore, sharp discontinuities such as edges contribute to low-frequency as well as high-frequency components. The energy concentration along the $\omega_1$ and $\omega_2$ axes is in part due to a rectangular window used to obtain a finite-extent image. The rectangular window creates artificial sharp discontinuities at the four boundaries. Discontinuities at the top and bottom of the image contribute energy along the $\omega_2$ axis and discontinuities at the two sides contribute energy along the $\omega_1$ axis. Figure 1.33 illustrates this property. Figure 1.33(a) shows an original image of 512 × 512 pixels, and Figure 1.33(b) shows $|X(\omega_1, \omega_2)|^{1/4}$ of the image in Figure 1.33(a). The operation $(\cdot)^{1/4}$ has the effect of compressing large amplitudes while expanding small amplitudes, and therefore shows $|X(\omega_1, \omega_2)|$ more clearly for higher-frequency regions. In this particular example, energy concentration along approximately diagonal directions is also visible. This is because of the many sharp discontinuities in the image along approximately diagonal directions. This example shows that most of the energy is concentrated in a small region in the frequency plane.

Since most of the signal energy is concentrated in a small frequency region, an image can be reconstructed without significant loss of quality and intelligibility from a small fraction of the transform coefficients. Figure 1.34 shows images that were obtained by inverse Fourier transforming the Fourier transform of the image in Figure 1.33(a) after setting most of the Fourier transform coefficients to zero. The percentages of the Fourier transform coefficients that have been preserved in

![Figure 1.33 Example of the Fourier transform magnitude of an image. (a) Original image $x[n_1, n_2]$ of 512 × 512 pixels. (b) $|X(\omega_1, \omega_2)|^{1/4}$, scaled such that the smallest value maps to the darkest level and the largest value maps to the brightest level. The operation $(\cdot)^{1/4}$ has the effect of compressing large amplitudes while expanding small amplitudes, and therefore shows $|X(\omega_1, \omega_2)|$ more clearly for higher-frequency regions.](image)
Figure 1.36 Illustration of energy concentration in the Fourier transform domain for a typical image. (a) Image obtained by preserving 12.4% of Fourier transform coefficients of the image in Figure 1.33(a). All other coefficients are set to 0. (b) Same as (a) with 10% of Fourier transform coefficients preserved. (c) Same as (a) with 4.8% of Fourier transform coefficients preserved.
$P_0(t)$ is all the projection of $f_c(t_1, t_2)$ along $\theta$. 
Applications of Tomography

- Medical: X-ray Tomography ➔ CAT scan
  - Reconstruct sections of human body via X-ray taken at different orientations
  - Detect Tumors
  - Cinematic reconstruction of a beating heart
  - Gamma rays or positrons can also be used.

- Geologic measurement from boreholes

- Acoustic or microwave transmitter.
  - Assume: only signal received is the one that propagates in a straight line from transmitter.

- Used for mapping hydrocarbon deposits, probing underground caverns for nuclear waste disposal, measuring the burn front of an in-situ coal gasification process.
Other Applications

- Electron microscopes produce
  projections of their specimen

- Radio telescopes measure
  projections of interstellar space

- Response of an LSI optical
  system to a line
  is projection of the point
  spread function (impulse response)
Projection Slice Theorem:

Consider continuous 3D space signal \( f_c(t_1, t_2) \) with Fourier Transform:

\[
F_c(R_1, R_2) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f_c(t_1, t_2) e^{-j\pi R_1 t_1} e^{-j\pi R_2 t_2} dt_1 dt_2
\]

\[
f_c(t_1, t_2) = \frac{1}{(2\pi)^2} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} F_c(R_1, R_2) e^{j\pi R_1 t_1} e^{j\pi R_2 t_2} dR_1 dR_2
\]

Integrate \( f_c(t_1, t_2) \) along parallel lines.
\[ p(t) = \text{result of integration} \]

\[ p(t) = \int_{-\infty}^{+\infty} f_c(t_1, t_2) \, dt_1 \left| \begin{array}{c} t_1 = t\cos\theta - u\sin\theta \\ u = -\infty \end{array} \right. \]

\[ t_2 = t\sin\theta + u\cos\theta \]

Continuum FT of \( p(t) \):

\[ P_0(r) = \int_{-\infty}^{+\infty} p(t) e^{-j\omega t} \, dt \]

Comparing \( P_0(r) \) and \( F_c(r_1, r_2) \) show:

\[ P_0(r) = F_c(r_1, r_2) \left| \begin{array}{c} r_1 = r\cos\theta \\ r_2 = r\sin\theta \end{array} \right. \]
Radar Inversion Formula

\[ \phi_c(t, t_2) = \frac{1}{4\pi^2} \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} F_c(r_1, r_2) \exp \left[ j r_1 t + j r_2 t_2 \right] \, dr_1 \, dr_2 \]

Describe 2D Fourier plane in POLAR coordinates \((r, \theta)\)

\[ f_c(t, t_2) = \frac{1}{4\pi^2} \int_{0}^{\infty} \int_{-\infty}^{+\infty} F_c(w \cos \theta, w \sin \theta) \exp \left[ j w (t \cos \theta + t_2 \sin \theta) \right] \, |w| \, dw \, d\theta \]

\[ G(w) = \frac{1}{4\pi^2} \int_{0}^{\infty} \int_{-\infty}^{+\infty} P_\theta(w) \exp \left[ j w (t \cos \theta + t_2 \sin \theta) \right] \, |w| \, dw \, d\theta \]

Inverse 1D Fourier Transform of \( P_\theta(w) |w| = F \{ g_\theta(t) \} = G(w) \)

where \( g_\theta(t) = p_\theta(t) * h(t) \rightarrow \text{robust kernel generalization} \)

\[ R(t) = F \{ \int_{-\infty}^{+\infty} |w|^2 \} \]

\[ \Rightarrow g_\theta(t) = \frac{d}{dt} \int_{-\infty}^{+\infty} \frac{P_\theta(t)}{t - \tau} \, d\tau \]
\[ f_c(t_1, t_2) = \frac{1}{2\pi} \int_0^{2\pi} g_0(t, \cos \theta + t \sin \theta) \, d\theta \]

**Explanation:**

1. **Start with the projection** 
   \( P_0(t) \)

2. **Filter it** 
   \( g_0(t) = P_0(t) + b(t) \)

3. **Back project it:**
   \[ f_c(t_1, t_2) = \frac{1}{2\pi} \int_0^{2\pi} g_0(t, \cos \theta + t \sin \theta) \, d\theta \]

- Start from \( f_c \) of one variable \( g_0(t) \) and get a \( f_c \) of two variables \( g_0(t, \cos \theta, t / \sin \theta) \). 
- Pictorially \( g_0(t, \cos \theta, t / \sin \theta) \) is obtained from \( g_0(t) \) by back projecting along \( u \), i.e. parallel to the original lines of integration that defined the projection.
Assume N projections at equally spaced angles \( \theta_i = \frac{\pi}{N} \) for \( i = 0, 1, \ldots, N-1 \).

Assume 2D FT of \( f_c \) is circularly band-limited.

Assume each projection has been sampled at the same sampling rate.

Take \( M \) point DFT of each sampled projection.

Get polar samples.

Assume \( f_c \) is BL to \( R_0 \).

\( \Rightarrow \) Samples of \( f_c \) \((r, \theta)\) enough.

\( \Rightarrow \) Sample in the domain and project \( \frac{r}{R_0} \).

\( \text{For } R_1 \leq r \leq R_2 \) for \( R_1 + R_2 > R_0 \).
Concentric Sampling

Assume 2D signal has F.T. confined to square (instead of circle)

Each projection is a Bandlimited function whose bandwidth is a function of projection angle. Change sampling rate as a function of angle.

\[ P_\theta(t) = \sum_{n=-\infty}^{\infty} P_\theta \left( \frac{n\pi}{W_0} \right) \text{sinc} \left[ \frac{W_0 (t-n\pi)}{W_0} \right] \]

Where \( W_0 \gg W \)

Take \( N \) point D.F.T. \( \Rightarrow \) Concentric grid
Reconstruction Algorithm

1. Assume \( f(t, r, z) \) adequately represented by
   NYU point Discrete Fourier Transform
   
   \[ \text{zero order interpolation} \Rightarrow \text{nearest neighbor} \]

2. Linear interpolation
   weighted average of near by samples

3. Convolution Back Projection [KAK]
   
   Define \( \Delta \theta_i = \theta_i - \theta_{i-1} \)
   Then from Rader formulation
   
   \[ f(t, r, z) = \sum_{i=0}^{N} \Delta \theta_i \cdot q_i(t) \cdot g_i(t, \cos \theta_i + t \cdot \sin \theta_i) \]
   \[ q_i(t) = \rho_i(t) \ast k(t) \]
   \[ F\{k(t)\} = 1 \]

Show Picture:

Conclusion: 1. Concentric - Better Than polar
   2. Convolution Back projection
      Better Than linear better than second
Iterative Reconstruction

Main idea:
Assume an initial estimate of 
\( f_c(t_1, t_2) \) - Call it 
\( f^*_c(t_1, t_2) \)

\( f^*_c(t_1, t_2) \) estimate at
\( Kth \) iteration

Compute projections of \( \Theta_i \) \( i = 1, \ldots, N \)
Compare with actual projections observed
Adjust \( f^*_c \) to \( f_c \) accordingly
to have much off the computed
and real projections
\( f^*_c(t_1, t_2) = f_c(t_1, t_2) + \sum_{i=1}^{N} \lambda_i \left[ \theta_i(t_1, t_2 + \pi, \sin(\theta)) - D_i f_c(t_1, t_2) \right] \)
\( D_i \) = evaluate projection and then backproject

Herman, Lent, Computers in

Biology Medicine

Vol 6, 1976 pp 273-94

Iterative Reconstruction Algorithm
Before we had projections with collimated beam validation, projection was made by integrating over a series of parallel line.

\[ \text{long collection time} \]

\[ \text{motion artifacts} \]

For beam: straight line but not parallel
Reconstruction from Projections

- Applications to tomography problems in medicine, radar imaging, geophysics. See Expansion
- Projection Slice Theorem. Reverse side
- Algorithms used:
  1. Fourier domain reconstruction algorithms.
  3. Iterative Reconstruction Algorithms.
- Tutorial reference:
Projection slice theorem

\[ p_\theta(t) = \int_{-\infty}^{\infty} f(x, y) |_{x=\cos \theta, y=\sin \theta} \, dx. \]

\[ P_e(\Omega) = \int_{-\infty}^{\infty} p_\theta(t) e^{-j\Omega t} \, dt. \]

\[ P_e(\Omega) = F_e(\Omega, \Omega_2) |_{\Omega=\cos \theta, \Omega_2=\sin \theta} = F_e(\Omega \cos \theta, \Omega \sin \theta). \]
transform by $X_r(\Omega_1, \Omega_2)$. Suppose we obtain a discrete-space signal $x(n_1, n_2)$ by sampling the analog signal $x_r(t_1, t_2)$ with sampling period $(T_1, T_2)$ as follows:

$$x(n_1, n_2) = x_r(t_1, t_2)|_{t_1=nT_1, t_2=nT_2}$$  \hspace{1cm} (1.52)

Equation (1.52) represents the input-output relationship of an ideal analog-to-digital (A/D) converter. The relationship between $X(\omega_1, \omega_2)$, the discrete-space Fourier transform of $x(n_1, n_2)$, and $X_r(\Omega_1, \Omega_2)$, the continuous-space Fourier transform of $x_r(t_1, t_2)$, is given by

$$X(\omega_1, \omega_2) = \frac{1}{T_1T_2} \sum_{\Delta=\pm} \sum_{\Delta=\pm} X_r \left( \omega_1 = \frac{2\pi n_1}{T_1}, \omega_2 = \frac{2\pi n_2}{T_2} \right)$$  \hspace{1cm} (1.55)

Two examples of $X_r(\Omega_1, \Omega_2)$ and $X(\omega_1, \omega_2)$ are shown in Figure 1.40. Figure 1.40(a) shows a case in which $1/T_1 > \Omega_1/\pi$ and $1/T_2 > \Omega_2/\pi$, where $\Omega_1$ and $\Omega_2$ are the cutoff frequencies of $X_r(\Omega_1, \Omega_2)$, as shown in the figure. Figure 1.40(b) shows a case in which $1/T_1 < \Omega_1/\pi$ and $1/T_2 < \Omega_2/\pi$. From the figure, when $1/T_1 > \Omega_1/\pi$ and $1/T_2 > \Omega_2/\pi$, $x_r(t_1, t_2)$ can be recovered from $x(n_1, n_2)$. Otherwise, $x_r(t_1, t_2)$ cannot be exactly recovered from $x(n_1, n_2)$ without additional information on $x_r(t_1, t_2)$. This is the 2-D sampling theorem, and it is a straightforward extension of the 1-D result.

An ideal digital-to-analog (D/A) converter recovers $x_r(t_1, t_2)$ from $x(n_1, n_2)$ when the sampling frequencies $1/T_1$ and $1/T_2$ are high enough to satisfy the require-