Types of Coding

• Source Coding - Code data to more efficiently represent the information
  – Reduces “size” of data
  – Analog - Encode analog source data into a binary format
  – Digital - Reduce the “size” of digital source data

• Channel Coding - Code data for transmission over a noisy communication channel
  – Increases “size” of data
  – Digital - add redundancy to identify and correct errors
  – Analog - represent digital values by analog signals

• Complete “Information Theory” was developed by Claude Shannon
Digital Image Coding

• Images from a 6 MPixel digital camera are 18 MBytes each
• Input and output images are digital
• Output image must be smaller (i.e. $\approx 500$ kBytes)
• This is a digital source coding problem
Two Types of Source (Image) Coding

• Lossless coding (entropy coding)
  – Data can be decoded to form exactly the same bits
  – Used in “zip”
  – Can only achieve moderate compression (e.g. 2:1 - 3:1) for natural images
  – Can be important in certain applications such as medical imaging

• Lossy source coding
  – Decompressed image is visually similar, but has been changed
  – Used in “JPEG” and “MPEG”
  – Can achieve much greater compression (e.g. 20:1 - 40:1) for natural images
  – Uses entropy coding
Entropy

- Let $X$ be a random variables taking values in the set $\{0, \cdots, M-1\}$ such that 
  \[ p_i = P\{X = i\} \]
- Then we define the entropy of $X$ as 
  \[ H(X) = - \sum_{i=0}^{M-1} p_i \log_2 p_i \]
  \[ = -E[\log_2 p_X] \]

$H(X)$ has units of bits
Conditional Entropy and Mutual Information

- Let \((X, Y)\) be a random variables taking values in the set \(\{0, \cdots, M - 1\}^2\) such that
  \[
p(i, j) = P\{X = i, Y = j\}
  \]
  \[
p(i|j) = \frac{p(i, j)}{\sum_{k=0}^{M-1} p(k, j)}
  \]
- Then we define the conditional entropy of \(X\) given \(Y\) as
  \[
  H(X|Y) = - \sum_{i=0}^{M-1} \sum_{j=0}^{M-1} p(i, j) \log_2 p(i|j)
  \]
  \[
  = -E[\log_2 p(X|Y)]
  \]
- The mutual information between \(X\) and \(Y\) is given by
  \[
  I(X;Y) = H(X) - H(X|Y)
  \]
  The mutual information is the reduction in uncertainty of \(X\) given \(Y\).
Entropy (Lossless) Coding of a Sequence

• Let $X_n$ be an i.i.d. sequence of random variables taking values in the set \( \{0, \cdots, M - 1\} \) such that

\[
P\{X_n = m\} = p_m
\]

– $X_n$ for each $n$ is known as a symbol

• How do we represent $X_n$ with a minimum number of bits per symbol?
A Code

- **Definition:** A code is a mapping from the discrete set of symbols \{0, \cdots, M - 1\} to finite binary sequences
  - For each symbol, \( m \) there is a corresponding finite binary sequence \( \sigma_m \)
  - \( |\sigma_m| \) is the length of the binary sequence

- Expected number of bits per symbol (bit rate)
  \[
  \bar{n} = E[|\sigma_{X_n}|] = \sum_{m=0}^{M-1} |\sigma_m| p_m
  \]

- Example for \( M = 4 \)

| \( m \) | \( \sigma_m \) | \( |\sigma_m| \) |
|-------|-------------|--------|
| 0     | 0 1         | 2      |
| 1     | 1 0         | 2      |
| 2     | 0           | 1      |
| 3     | 1 0 1 0 0   | 6      |

- Encoded bit stream
  \[(0, 2, 1, 3, 2) \rightarrow (01|0|10|100100|0)\]
Fixed versus Variable Length Codes

• Fixed Length Code - $|\sigma_m|$ is constant for all $m$

• Variable Length Code - $|\sigma_m|$ varies with $m$

• Problem
  – Variable length codes may not be uniquely decodable
  – Example: Using code from previous page

    $\begin{align*}
    (6) & \rightarrow (100100) \\
    (1, 0, 2, 2) & \rightarrow (10|01|0|0)
    \end{align*}$

    – Different symbol sequences can yield the same code

• **Definition:** A code is *Uniquely Decodable* if there exists only a single unique decoding of each coded sequence.

• **Definition:** A *Prefix Code* is a specific type of uniquely decodable code in which no code is a prefix of another code.
Lower Bound on Bit Rate

• **Theorem:** Let \( C \) be a uniquely decodable code for the i.i.d. symbol sequence \( X_n \). Then the mean code length is greater than \( H(X_n) \).

\[
\bar{n} \triangleq E[|\sigma_{X_n}|] = \sum_{m=0}^{M-1} |\sigma_m| p_m \geq H(X_n)
\]

• Question: Can we achieve this bound?

• Answer: Yes! Constructive proof using Huffman codes
Huffman Codes

• Variable length prefix code ⇒ Uniquely decodable

• Basic idea:
  – Low probability symbols ⇒ Long codes
  – High probability symbols ⇒ short codes

• Basic algorithm:
  – Low probability symbols ⇒ Long codes
  – High probability symbols ⇒ short codes
Huffman Coding Algorithm

1. Initialize list of probabilities with the probability of each symbol
2. Search list of probabilities for two smallest probabilities, $p_k^*$ and $p_l^*$.
3. Add two smallest probabilities to form a new probability, $p_m = p_k^* + p_l^*$.
4. Remove $p_k^*$ and $p_l^*$ from the list.
5. Add $p_m$ to the list.
6. Go to step 2 until the list only contains 1 entry
Recursive Merging for Huffman Code

- Example for $M = 8$ code

```
  p0  p1  p2  p3  p4  p5  p6  p7
  0.4  0.08  0.08  0.2  0.12  0.07  0.04  0.01

  0.4  0.08  0.08  0.2  0.12  0.07  0.05

  0.4  0.08  0.08  0.2  0.12  0.12

  0.4  0.16  0.2  0.12  0.12

  0.4  0.16  0.2  0.24

  0.4  0.36  0.24

  0.4  0.6

  1.0
```
Resulting Huffman Code

- Binary codes given by path through tree

\[
\begin{align*}
p_0 & \quad p_1 & \quad p_2 & \quad p_3 & \quad p_4 & \quad p_5 & \quad p_6 & \quad p_7 \\
1 & \quad 0111 & \quad 0110 & \quad 010 & \quad 001 & \quad 0001 & \quad 00001 & \quad 00000
\end{align*}
\]
Upper Bound on Bit Rate of Huffman Code

• **Theorem:** For a Huffman code, \( \bar{n} \) has the property that

\[
H(X_n) \leq \bar{n} < H(X_n) + 1
\]

• A Huffman code is within 1 bit of optimal efficiency

• Can we do better?
Coding in Blocks

- We can code blocks of symbols to achieve a bit rate that approaches the entropy of the source symbols.

\[ \cdots, \underbrace{X_0, \cdots, X_{m-1}}_{Y_0}, \underbrace{X_m, \cdots, X_{2m-1}}_{Y_1}, \cdots \]

So we have that

\[ Y_n = [X_{nm}, \cdots, X_{(n+1)m-1}] \]

where

\[ Y_n \in \{0, \cdots, M^m - 1\} \]
Bit Rate Bounds for Coding in Blocks

• It is easily shown that $H(Y_n) = mH(X_n)$ and the number of bits per symbol $X_n$ is given by $\bar{n}_x = \frac{n_y}{m}$ where $n_y$ is the number of bits per symbol for a Huffman code of $Y_n$.

• Then we have that

\[
H(Y_n) \leq \bar{n}_y < H(Y_n) + 1
\]
\[
\frac{1}{m}H(Y_n) \leq \frac{n_y}{m} < \frac{1}{m}H(Y_n) + \frac{1}{m}
\]
\[
H(X_n) \leq \frac{n_y}{m} < H(X_n) + \frac{1}{m}
\]
\[
H(X_n) \leq \bar{n}_x < H(X_n) + \frac{1}{m}
\]

• As the block size grows, we have

\[
\lim_{m \to \infty} H(X_n) \leq \lim_{m \to \infty} \bar{n}_x \leq H(X_n) + \lim_{m \to \infty} \frac{1}{m}
\]
\[
H(X_n) \leq \lim_{m \to \infty} \bar{n}_x \leq H(X_n)
\]

• So we see that for a Huffman code of blocks with length $m$

\[
\lim_{m \to \infty} \bar{n}_x = H(X_n)
\]
Comments on Entropy Coding

• As the block size goes to infinity the bit rate approaches the entropy of the source

\[ \lim_{m \to \infty} \bar{n}_x = H(X_n) \]

• A Huffman coder can achieve this performance, but it requires a large block size.

• As \( m \) becomes large \( M^m \) becomes very large \( \Rightarrow \) large blocks are not practical.

• This assumes that \( X_n \) are i.i.d., but a similar result holds for stationary and ergodic sources.

• Arithmetic coders can be used to achieve this bitrate in practical situations.
Run Length Coding

• In some cases, long runs of symbols may occur. In this case, run length coding can be effective as a preprocessor to an entropy coder.

• Typical run length coder uses

  \[ \cdot\cdot\cdot, (\text{value}, \# \text{ of repetitions}), (\text{value}, \# \text{ of repetitions}+1), \cdot\cdot\cdot \]

  where \(2^b\) is the maximum number of repetitions

• Example: Let \(X_n \in \{0, 1, 2\}\)

  \[ \cdot\cdot\cdot | \overbrace{0000000}^{07} | \underline{111} | \overbrace{222222}^{26} | \cdot\cdot\cdot \]

  \[ \cdot\cdot\cdot | \overbrace{0000000}^{08} | \underline{00} | \underline{111} | \cdot\cdot\cdot \]

  • If more than \(2^b\) repetitions occur, then the repetition is broken into segments

  • Many other variations are possible.
Predictive Entropy Coder for Binary Images

- Uses in transmission of Fax images (CCITT G4 standard)
- Framework
  - Let $X_s$ be a binary image on a rectangular lattice $s = (s_1, s_2) \in S$
  - Let $W$ be a causal window in raster order
  - Determine a model for $p(x_s | x_{s+r} r \in W)$
- Algorithm
  1. For each pixel in raster order
     (a) Predict
        $$\hat{X}_s = \begin{cases} 
        1 & \text{if } p(1 | X_{s+r} r \in W) > p(0 | X_{s+r} r \in W) \\
        0 & \text{otherwise}
        \end{cases}$$
     (b) If $X_s = \hat{X}_s$ send 0; otherwise send 1
  2. Run length code the result
  3. Entropy code the result
Predictive Entropy Coder Flow Diagram

Encoder

Decoder

C. A. Bouman: Digital Image Processing - January 7, 2024
How to Choose $p(x_s | x_{s+r} \ r \in W)$?

• Non-adaptive method
  – Select typical set of training images
  – Design predictor based on training images

• Adaptive method
  – Allow predictor to adapt to images being coded
  – Design decoder so it adapts in same manner as encoder
Non-Adaptive Predictive Coder

- Method for estimating predictor
  1. Select typical set of training images
  2. For each pixel in each image, form \( z_s = (x_{s+r_0}, \cdots, x_{s+r_{p-1}}) \) where \( \{r_0, \cdots, r_{p-1}\} \in W \).
  3. Index the values of \( z_s \) from \( j = 0 \) to \( j = 2^p - 1 \)
  4. For each pixel in each image, compute
     \[
     h_s(i, j) = \delta(x_s = i)\delta(z_s = j)
     \]
     and the histogram
     \[
     h(i, j) = \sum_{s \in S} h_s(i, j)
     \]
  5. Estimate \( p(x_s | x_{s+r} \in W) = p(x_s | z_s) \) as
     \[
     \hat{p}(x_s = i | z_s = j) = \frac{h(i, j)}{\sum_{k=0}^{1} h(k, j)}
     \]
Adaptive Predictive Coder

- Adapt predictor at each pixel
- Update value of $h(i, j)$ at each pixel using equations
  
  $$
  h(i, j) \leftarrow h(i, j) + \delta(x_s = i)\delta(z_s = j)
  $$

  $$
  N(j) \leftarrow N(j) + 1
  $$

- Use updated values of $h(i, j)$ to compute new predictor at each pixel

  $$
  \hat{p}(i|j) \leftarrow \frac{h(i, j)}{N(j)}
  $$

- Design decoder to track encoder
Adaptive Predictive Entropy Coder Flow Diagram

Encoder

Xs → XOR → Run Length Encoding → Huffman Coding

Causal Predictor

Causal Histogram Estimation

Decoder

Huffman Decoding → Run Length Decoding → XOR

Causal Predictor

Causal Histogram Estimation

Xs
Lossy Source Coding

• Method for representing discrete-space signals with minimum distortion and bit-rate

• Outline
  – Rate-distortion theory
  – Karhunen-Loeve decorrelating Transform
  – Practical coder structures
Distortion

- Let $X$ and $Z$ be random vectors in $\mathbb{R}^M$. Intuitively, $X$ is the original image/data and $Z$ is the decoded image/data. Assume we use the squared error distortion measure given by

$$d(X, Y) = \| X - Z \|^2$$

Then the distortion is given by

$$D = E \left[ d(X, Y) \right] = E \left[ \| X - Z \|^2 \right]$$

- This actually applies to any quadratic norm error distortion measure since we can define

$$\tilde{X} = AX \quad \text{and} \quad \tilde{Z} = AZ$$

So

$$\tilde{D} = E \left[ \| \tilde{X} - \tilde{Z} \|^2 \right] = E \left[ \| X - Z \|_B^2 \right]$$

where $B = A^t A$. 
Lossy Source Coding: Theoretical Framework

- Notation for source coding

\[ X_n \in \mathbb{R}^M \text{ for } 0 \leq n < N \text{ - a sequence of i.i.d. random vectors} \]
\[ Y \in \{0, 1\}^K \text{ - a } K \text{ bit random binary vector.} \]
\[ Z_n \in \mathbb{R}^M \text{ for } 0 \leq n < N \text{ - the decoded sequence of random vectors.} \]

\[ X^{(N)} = (X_0, \cdots, X_{N-1}) \]
\[ Z^{(N)} = (Z_0, \cdots, Z_{N-1}) \]

- Encoder function: \( Y = Q(X_0, \cdots, X_{N-1}) \)
- Decoder function: \((Z_0, \cdots, Z_{N-1}) = f(Y)\)

- Resulting quantities

Bit-rate = \( \frac{K}{N} \)

Distortion = \( \frac{1}{N} \sum_{n=0}^{N-1} E \left[ \|X_n - Z_n\|^2 \right] \)

- How do we choose \( Q(\cdot) \) to minimize the bit-rate and distortion?
Differential Entropy

• Notice that the information contained in a Gaussian random variable is infinite, so the conventional entropy $H(X)$ is not defined.

• Let $X$ be a random vector taking values in $\mathbb{R}^M$ with density function $p(x)$. Then we define the differential entropy of $X$ as

$$h(X) = -\int_{x\in\mathbb{R}^M} p(x) \log_2 p(x) \, dx$$

$$= -\mathbb{E} \left[ \log_2 p(X) \right]$$

$h(X)$ has units of bits
Conditional Entropy and Mutual Information

• Let \( X \) and \( Y \) be a random vectors taking values in \( \mathbb{R}^M \) with density function \( p(x, y) \) and conditional density \( p(x|y) \).

• Then we define the differential conditional entropy of \( X \) given \( Y \) as

\[
\begin{align*}
    h(X|Y) &= -\int_{x \in \mathbb{R}^M} \int_{y \in \mathbb{R}^M} p(x, y) \log_2 p(x|y) \\
           &= -E[\log_2 p(X|Y)]
\end{align*}
\]

• The mutual information between \( X \) and \( Y \) is given by

\[
    I(X; Y) = h(X) - h(X|Y) = I(Y; X)
\]

• **Important:** The mutual information is well defined for both continuous and discrete random variables, and it represents the reduction in uncertainty of \( X \) given \( Y \).
The Rate-Distortion Function

• Define/Remember:
  – Let $X_0$ be the first element of the i.i.d. sequence.
  – Let $D \geq 0$ be the allowed distortion.

• For a specific distortion, $D$, the rate is given by

$$R(D) = \inf_{Z} \{ I(X_0; Z) : E[\|X_0 - Z\|^2] \leq D \}$$

where the infimum (i.e. minimum) over $Z$ is taken over all random variables $Z$.

• Later, we will show that for a given distortion we can find a code that gets arbitrarily close to this optimum bit-rate.
Properties of the Rate-Distortion Function

• Properties of $R(D)$
  
  – $R(D)$ is a monotone decreasing function of $D$.
  
  – If $D \geq E \left[ \|X_0\|^2 \right]$, then $R(D) = 0$
  
  – $R(D)$ is a convex function of $D$
Shannon’s Source-Coding Theorem

• Shannon’s Source-Coding Theorem:
For any $R' > R(D)$ and $D' > D$ there exists a sufficiently large $N$ such that there is an encoder

$$Y = Q(X_0, \cdots, X_{N-1})$$

which achieves

$$\text{Rate} = \frac{K}{N} \leq R'$$

and

$$\text{Distortion} = \frac{1}{N} \sum_{n=0}^{N-1} E \left[ \|X_n - Z_n\|^2 \right] \leq D'$$

• Comments:

– One can achieve a bit rate arbitrarily close to $R(D)$ at a distortion $D$.

– Proof is constructive (but not practical), and uses codes that are randomly distributed in the space $IR^{MN}$ of source symbols.
Example 1: Coding of Gaussian Random Variables

• Let $X \sim N(0, \sigma^2)$ with distortion function $E[|X - Z|^2]$, then it can be shown that the rate-distortion function has the form

$$
R(\delta) = \max \left\{ \frac{1}{2} \log \left( \frac{\sigma^2}{\delta} \right), 0 \right\}
$$

$$
D(\delta) = \min \{ \sigma^2, \delta \}
$$

• Intuition:
  – $\delta$ is a parameter which represents the $\sqrt{\text{quantization step}}$
  – $\frac{1}{2} \log \left( \frac{\sigma^2}{\delta} \right)$ represents the number of bits required to encode the quantized scalar value.
  – Minimum number of bits must be $\geq 0$.
  – Maximum distortion must be $\leq \sigma^2$. 

![](Distortion versus Rate Curve)
Example 2: Coding of $N$ Independent Gaussian Random Variables

- Let $X = [X_1, \cdots, X_{N-1}]^t$ with independent components such that $X_n \sim N(0, \sigma_n^2)$, and define the distortion function to be

$$\text{Distortion} = E \left[ ||X - Z||^2 \right]$$

Then it can be shown that the rate-distortion function has the form

$$R(\delta) = \sum_{n=0}^{N-1} \max \left\{ \frac{1}{2} \log \left( \frac{\sigma_n^2}{\delta} \right), 0 \right\}$$

$$D(\delta) = \sum_{n=0}^{N-1} \min \{ \sigma_n^2, \delta \}$$

- Intuition:
  - In an optimal coder, the quantization step should be approximately equal for each random variable being quantized.
  - The bit rate and distortion both add for each component.
  - It can be proved that this solution is optimal.
Example 3: Coding of Gaussian Random Vector

• Let \( X \sim N(0, R) \) be a \( N \) dimensional Gaussian random vector, and define the distortion function to be

\[
\text{Distortion} = E[\|X - Z\|^2]
\]

• Analysis: We know that we can always represent the covariance in the form

\[
R = T^t \Lambda T
\]

where the columns of \( T \) are the eigenvectors of \( R \), and \( \Lambda = diag\{\sigma_0^2, \cdots, \sigma_{N-1}^2\} \) is a diagonal matrix of eigenvalues. We can then decorrelate the Gaussian random vector with the following transformation.

\[
\tilde{X} = T^t X
\]

From this we can see that \( \tilde{X} \) has the covariance matrix given by

\[
E[\tilde{X}\tilde{X}^t] = E[T^t XX^t T] = T^t E[XX^t] T = T^t RT = \Lambda
\]

So therefore, \( \tilde{X} \) meets the conditions of Example 2. Also, we see that

\[
E[\|X - Z\|^2] = E[\|\tilde{X} - \tilde{Z}\|^2]
\]

where \( \tilde{X} = T^t X \) and \( \tilde{Z} = T^t Z \) because \( T \) is an orthonormal transform.
Example 3: Coding of Gaussian Random Vector (Result)

• Let \( X \sim N(0, R) \) be a \( N \) dimensional Gaussian random vector, and define the distortion function to be

\[
\text{Distortion} = E[\|X - Z\|^2]
\]

Then it can be shown that the rate-distortion function has the form

\[
R(\delta) = \sum_{n=0}^{N-1} \max \left\{ \frac{1}{2} \log \left( \frac{\sigma_n^2}{\delta} \right), 0 \right\}
\]

\[
D(\delta) = \sum_{n=0}^{N-1} \min \{ \sigma_n^2, \delta \}
\]

where \( \sigma_0^2, \ldots, \sigma_{N-1}^2 \) are the eigenvalues of \( R \).

• Intuition:

  – An optimal code requires that the components of a vector be decorrelated before source coding.
Example 4: Coding of Stationary Gaussian Random Process

- Let $X_n$ be a stationary Gaussian random process with power spectrum $S_x(\omega)$, and define the distortion function to be

$$\text{Distortion} = E[|X_n - Z|^2]$$

Then it can be shown that the rate-distortion function has the form

$$R(\delta) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \max \left\{ \frac{1}{2} \log \left( \frac{S_x(\omega)}{\delta} \right), 0 \right\} d\omega$$

$$D(\delta) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \min \{ S_x(\omega), \delta \} d\omega$$

- Intuition:
  - The Fourier transform decorrelates a stationary Gaussian random process.
  - Frequencies with amplitude below $\delta$ are clipped to zero.
The Discrete Cosine Transform (DCT)

- DCT (There is more than one version)

\[
F(k) = \frac{1}{\sqrt{N}} \sum_{n=0}^{N-1} f(n) c(k) \cos \left( \frac{\pi(2n + 1)k}{2N} \right)
\]

where

\[
c(k) = \begin{cases} 
1 & k = 0 \\
\sqrt{2} & k = 1, \ldots, N - 1
\end{cases}
\]

- Inverse DCT (IDCT)

\[
f(n) = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} F(k) c(k) \cos \left( \frac{\pi(2n + 1)k}{2N} \right)
\]

- Comments:

  - In this form, the DCT is an orthonormal transform. So if we define the matrix \( F \) such that

\[
F_{n,k} = c(k) \cos \left( \frac{\pi(2n + 1)k}{2N} \right),
\]

then \( F^{-1} = F^H \) where

\[
F^{-1} = [F^t]^* = F^H
\]

  - Takes and \( N \)-point real valued signal to an \( N \)-point real valued signal.
Relationship Between DCT and DFT

• Let us define the padded version of \( f(n) \) as

\[
f_p(n) = \begin{cases} f(n) & 0 \leq n \leq N - 1 \\ 0 & N \leq n \leq 2N - 1 \end{cases}
\]

and its \( 2N \)-point DFT denoted by \( F_p(k) \). Then the DCT can be written as

\[
F(k) = \frac{c(k)}{\sqrt{N}} \sum_{n=0}^{N-1} f(n) \cos \left( \frac{\pi(2n+1)k}{2N} \right)
\]

\[
= \frac{c(k)}{\sqrt{N}} \sum_{n=0}^{N-1} f(n) \text{Re} \left\{ e^{-j\frac{2\pi(2n+1)k}{2N}} e^{-j\frac{\pi k}{2N}} \right\}
\]

\[
= \frac{c(k)}{\sqrt{N}} \text{Re} \left\{ e^{-j\frac{\pi k}{2N}} \sum_{n=0}^{2N-1} f_p(n) e^{-j\frac{2\pi(2n+1)k}{2N}} \right\}
\]

\[
= \frac{c(k)}{\sqrt{N}} \text{Re} \left\{ e^{-j\frac{\pi k}{2N}} F_p(k) \right\}
\]

\[
= \frac{c(k)}{\sqrt{N}} \left( F_p(k) e^{-j\frac{\pi k}{2N}} + F_p(k) e^{+j\frac{\pi k}{2N}} \right)
\]

\[
= \frac{c(k)e^{-j\frac{\pi k}{2N}}}{\sqrt{N}} \left( F_p(k) + \left( F_p(k) e^{-j\frac{2\pi k}{2N}} \right)^* \right)
\]
Interpretation of DFT Terms

• Consider the inverse DCT for each of the two terms

\[ F_p(k) \iff f_p(n) \]

• \( DCT\{f_p(n)\} \Rightarrow F_p(k) \)

• \( DCT\{f_p(-n + N - 1)\} \Rightarrow \left( F_p(k) e^{-j\frac{2\pi k}{2N}} \right)^* \)

• Simple example for \( N = 4 \)

\[
\begin{align*}
  f(n) &= [f(0), f(1), f(2), f(3)] \\
  f_p(n) &= [f(0), f(1), f(2), f(3), 0, 0, 0, 0] \\
  f_p(-n+N-1) &= [0, 0, 0, 0, f(3), f(2), f(1), f(0)] \\
  f(n) + f_p(-n+N-1) &= [f(0), f(1), f(2), f(3), f(3), f(2), f(1), f(0)]
\end{align*}
\]
Relationship Between DCT and DFT (Continued)

So the DCT is formed by $2N$-point DFT of $f(n) + f(-n + N - 1)$.

$$F(k) = \frac{c(k)e^{-j\frac{\pi k}{2N}}}{\sqrt{N}} DFT_{2N} \{f(n) + f(-n + N - 1)\}$$