



Research Institute for Future Media Computing Institute of Computer Vision
未来媒体技术与研究所 计算机视觉研究所



多媒体系统导论

Fundamentals of Multimedia System

授课教师：朱映映教授

Email: zhuyy@szu.edu.cn

第八讲

Lossy Compression Algorithms

第8章



1. 有损压缩

- ◆ 有损压缩产生了较高的压缩比
- ◆ 压缩与解压缩过程导致了信息丢失
- ◆ 从压缩数据中解压缩的文件是源文件的近似
- ◆ 多媒体数据通常联合使用无损与有损算法

2. 失真度量 (Distortion measure)

- 评估编码系统性能的一种方法是失真度量法
- 失真度量是一种在某种失真标准下一个近似值与原值的接近程度的数学量。
- 常用：均方误差 (MSE) , 信噪比 (SNR) , 峰值信噪比 (PSNR)
- 均方误差 (Mean Square Error, MSE)

$$MSE = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} |x(m, n) - \tilde{x}(m, n)|^2$$

式中, $x(m, n)$ 为原始图像的像素值, $\tilde{x}(m, n)$ 为解压缩后的像素值。

2. 失真度量

- 峰值信噪比 (Peak Signal to Noise Ratio, PSNR)

用峰值信号与噪声之比衡量, 定义为最大信号峰值的平方与信号的均方误差(MSE)之比

$$PSNR = 10 \log_{10} \frac{(\text{Peak Signal Value})^2}{MSE} \quad (\text{dB})$$

对 8 位二进制图像

$$PSNR = 10 \log_{10} \frac{255^2}{MSE} \quad (\text{db})$$

2. 失真度量

- 信噪比 (Signal to Noise Ratio, SNR)

$$\text{SNR} = 10 \cdot \log_{10} \left[\frac{\sum_{x=1}^{N_x} \sum_{y=1}^{N_y} f(x, y)^2}{\sum_{x=1}^{N_x} \sum_{y=1}^{N_y} (f(x, y) - \hat{f}(x, y))^2} \right]$$

- PSNR与SNR的关系

$$\text{PSNR(dB)} = \text{SNR(dB)} + 10 \cdot \log_{10} \left[\frac{\sum_{x=1}^{N_x} \sum_{y=1}^{N_y} (255)^2}{\sum_{x=1}^{N_x} \sum_{y=1}^{N_y} (f(x, y))^2} \right]$$

2. 失真度量

- ◆ 例：原信号{12 16 16 12 12 8 8 12 }，压缩后数据{8 12}，解压后数据{12 12 12 12 12 12 12 12}

$$\text{MSE} = (4^2 + 4^2 + (-4)^2 + (-4)^2) / 8 = 8$$

$$\text{SNR} = 10 * \log_{10} \left((12^2 + 16^2 + 16^2 + 12^2 + 12^2 + 8^2 + 8^2 + 12^2) / (4^2 + 4^2 + (-4)^2 + (-4)^2) \right) = 12.79 \text{ db}$$

$$\text{PSNR} = 10 * \log_{10}(16^2 / 8) = 15.05 \text{ db}$$

- ◆ Matlab demo MP.m

3. 比率失真理论

- 有损压缩牵涉到比率与失真之间的权衡
- 比率是重视原始信号所需的平均位数
- 给定源和失真度量， D 是失真容忍量，在 D 允许的失真范围内， $R(D)$ 表示源数据编码的最低比率。

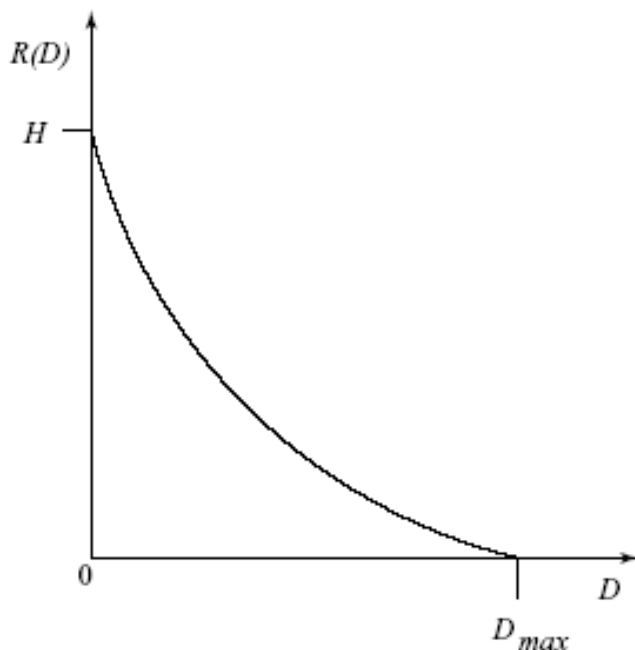


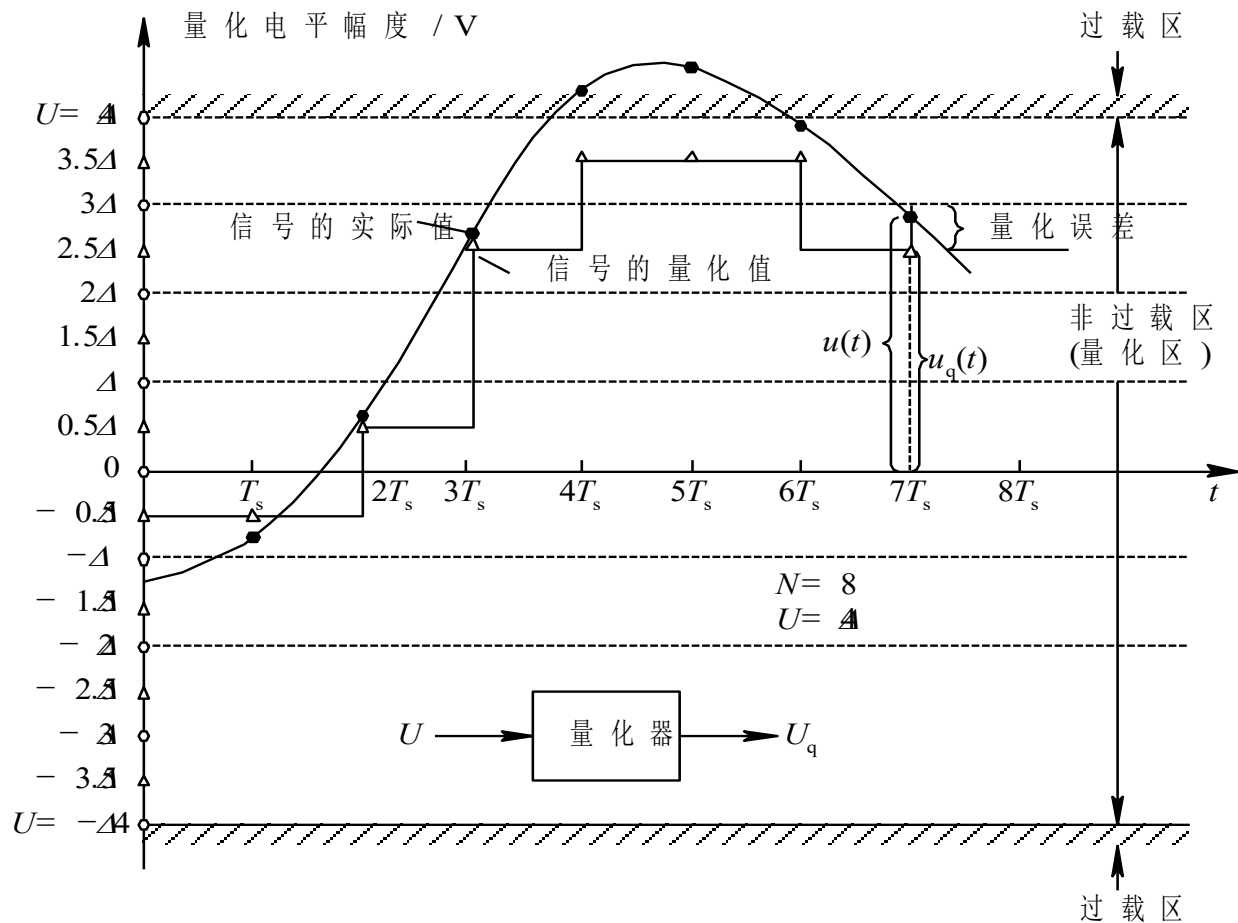
图 典型的比率-失真函数

$D=0$ 时，表示没有损失时的最小比率，是源数据的熵；

$R(D)=0$ 时，失真达到最大值，表示没有编码。

4. 量化

◆ 均匀量化



4. 量化

量化间隔计算

量化级间隔（级差） $\Delta = \frac{2U}{N}$

N : 量化级数（等于离散值的数目）

U : 临界过载电压。

过载电压：幅度大于 U （本图为大于 4Δ ）的电压。

量化区： $|u(t)| \leq U$ 的区域

量化误差 { 量化区内，都不会超过 $\Delta/2$

量化过载区内，都超过 $\Delta/2$

4. 量化

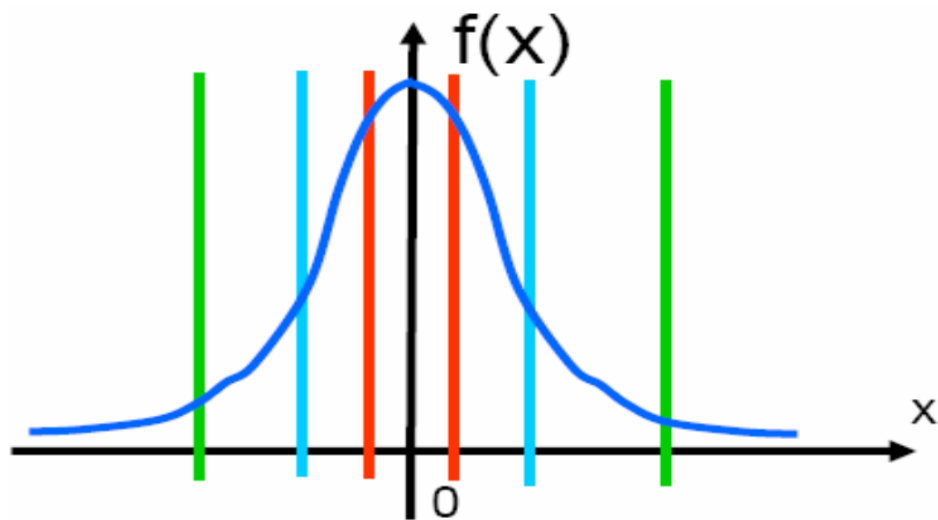
- 均匀量化, 其量化间隔 $\Delta = \frac{2U}{N}$
- 码字位数 n 与量化级数 N 的关系是 $N=2^n$
- 在一定的量化区内, 级数越多, 量化间隔越小, 噪声越小, 信噪比越大, 但编码位数越多。

4. 量化

非均匀量化

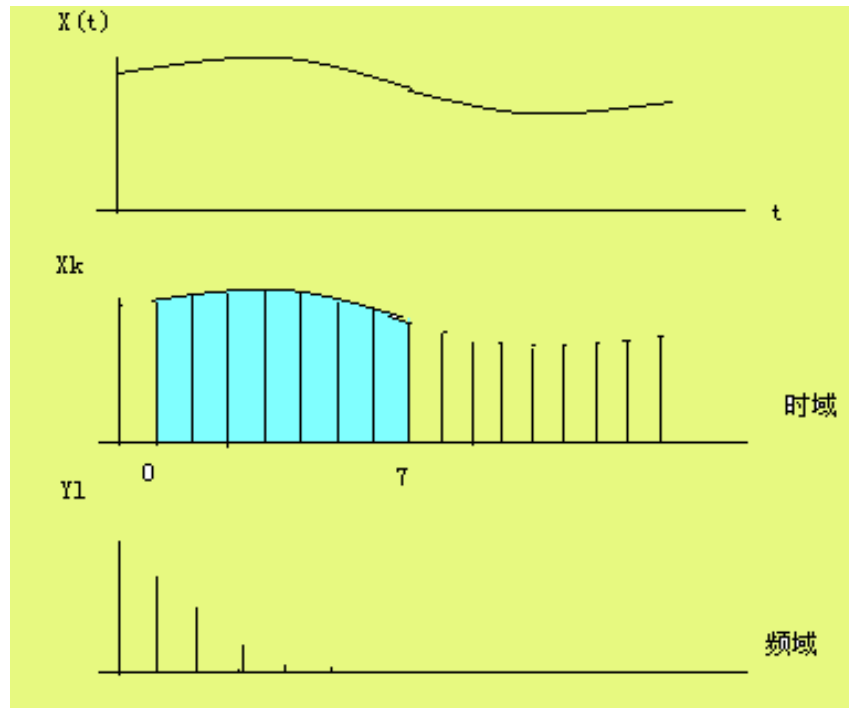
- 均匀量化器只对均匀分布信源是最佳的
- 对给定的 M ，为了减小 $MSQE$ ，我们应该在概率 $f_X(x)$ 较大时缩小步长，而在 $f_X(x)$ 较小时增大步长

$$D = \sigma_q^2 = \int_{-\infty}^{\infty} (x - Q(x))^2 f(x) dx = \sum_{i=1}^M \int_{b_{i-1}}^{b_i} (x - y_i)^2 f(x) dx$$



5. 变换编码

例1： $x(t)$ 为模拟信号， $x(k)$ 为取样后的样本序列 $x(k)$ ， 已 $n=8$ 为例， 对 x_0, \dots, x_7 进行正交变换， 得到YI的8个输出值 y_0, \dots, y_7 。 在该坐标系中， 信息集中在 y_0, y_1, y_2 三个分量上。



5. 变换编码

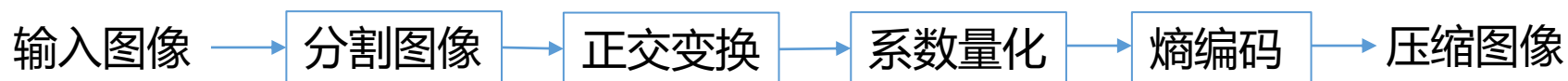
- ◆ 变换编码不是直接对空域信号（图像）和时域信号（语音）进行编码，而是首先将空域（时域）信号映射变换到另一个正交矢量空间（变换域或频域），产生一批变换系数，然后对这些变换系数进行编码处理。
- ◆ 变换编码是一种间接编码方法，利用 (1)信号在时域和空域上数据之间相关性大，数据冗余度大，经过变换在变换域中描述，数据相关性大大减少，数据冗余量减少，参数独立，数据量少，这样再进行量化，编码就能得到较大的压缩比；(2)利用人的视觉特性对高频细节不敏感，可以滤除高频信号，保留低频信号，实现压缩。
- ◆ “卡—洛变换”（Karhunen-Loeve）是一种线性正交变换，称为最佳变换，并将其性能作为一种标准，用以比较其它变换的性能。但它不存在快速变换。
- ◆ 典型的准最佳变换有DCT（离散余弦变换）、DFT(离散傅里叶变换)、WHT（Walsh Hadama 变换）、HrT(Haar 变换)等。其中，最常用的是离散余弦变换。

5. 变换编码

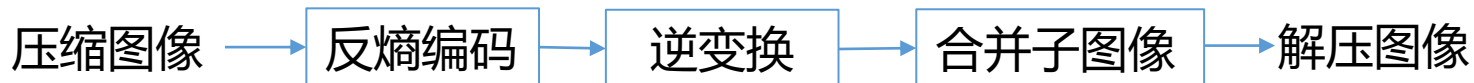
- ◆ 基本原理：如果 Y 是对输入向量 X 进行线性变换 T 的结果，线性变换 T 使得 Y 的元素间的相关性比 X 中元素的相关性更弱，那么对 Y 的编码效率就比对 X 编码的效率高。
- ◆ 如果大部分的信息可以用变换后的向量 Y 中的前几个分量精确描述，那么对剩余的分量可以只进行量化或者设为零。
- ◆ 一般而言，变换 T 不对数据进行压缩，压缩是通过对 Y 的分量的量化和处理完成的。
- ◆ 任何信号都可以表示为多个不同振幅和频率的正弦波或余弦波信号的叠加，这一过程就叫做傅里叶分析。
- ◆ 一个信号通常由一个直流(DC)分量和多个交流(AC)分量组成。

5. 变换编码

◆ 变换编码系统如下图



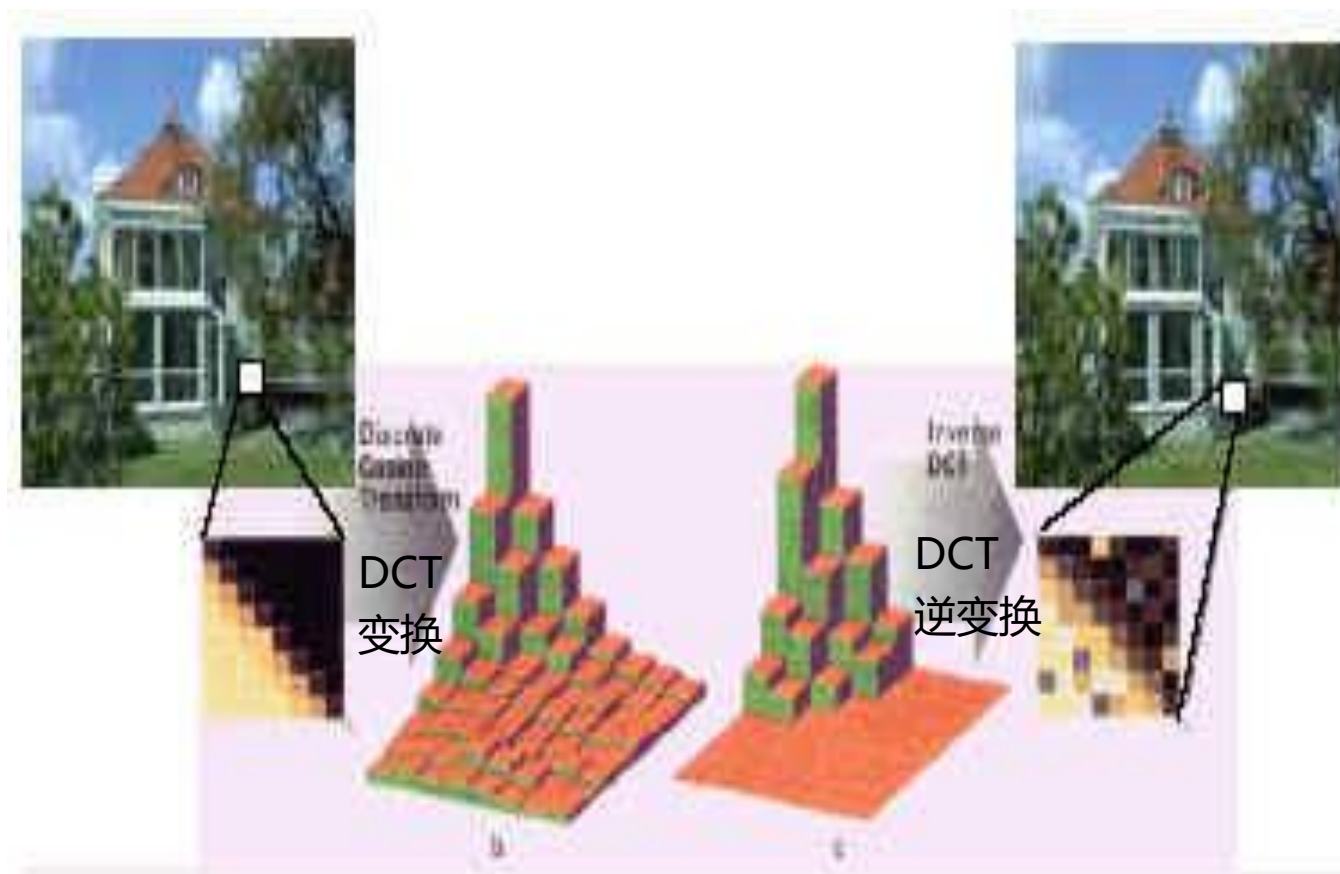
(a)编码器框图



(b)解码器框图

5. 变换编码

例2



DCT能够以数据无关的方式解除输入信号之间的相关性

1D Discrete Cosine Transform

- Given an input function $f(i)$ over one integer variable i (a piece of a vector), the 1D DCT transforms it into a new function $F(u)$, with integer u running over the same range as i . The general definition of the transform is:

$$F(u) = \sqrt{\frac{2}{M}} C(u) \sum_{i=0}^{M-1} \cos \frac{(2i+1)u\pi}{2M} f(i)$$

- $i, u = 0, 1 \dots \dots M-1$;
- The constants $C(u)$ are determined by

$$C(u) = \begin{cases} \frac{\sqrt{2}}{2} & \text{if } u = 0, \\ 1 & \text{otherwise.} \end{cases}$$

1D Discrete Cosine Transform

- ◆ When $M = 8$, 1D Inverse Discrete Cosine Transform (1D-IDCT)

$$\tilde{f}(i) = \sum_{u=0}^7 \frac{C(u)}{2} \cos \frac{(2i+1)u\pi}{16} F(u)$$

- ◆ In short, the role of the DCT is to decompose the original signal into its DC and AC components; the role of the IDCT is to reconstruct (recompose) the signal.

1D DCT and IDCT with 8 numbers

- Consider a data sequence with 8 numbers
- 1D DCT

$$F(u) = \frac{C(u)}{2} \sum_{i=0}^7 \cos \frac{(2i+1)u\pi}{16} f(i)$$

- 1D IDCT

$$\tilde{f}(i) = \sum_{u=0}^7 \frac{C(u)}{2} \cos \frac{(2i+1)u\pi}{16} F(u)$$

- The constants $C(u)$ are determined by

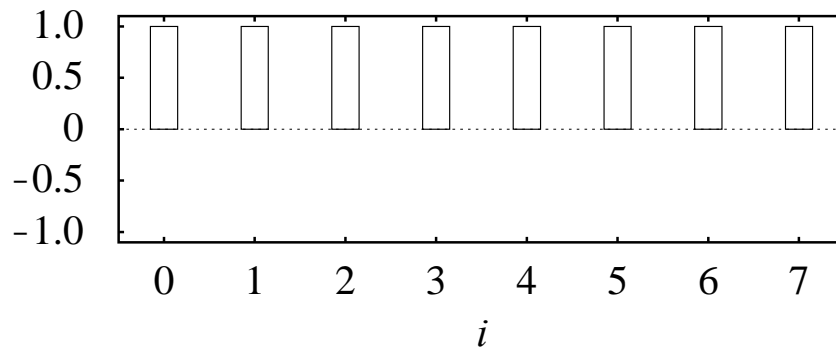
$$C(u) = \begin{cases} \frac{\sqrt{2}}{2} & \text{if } u = 0, \\ 1 & \text{otherwise.} \end{cases}$$

The 1D DCT Basis Functions

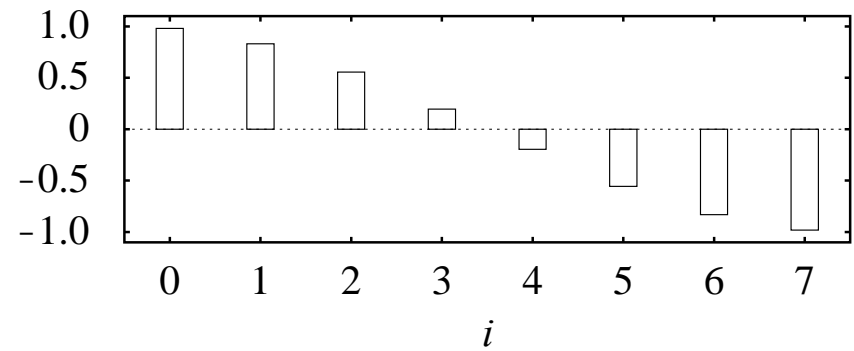
DCT and IDCT use the same set of cosine functions; they are known as basis functions:

$$\cos \frac{(2i + 1)u\pi}{16}$$

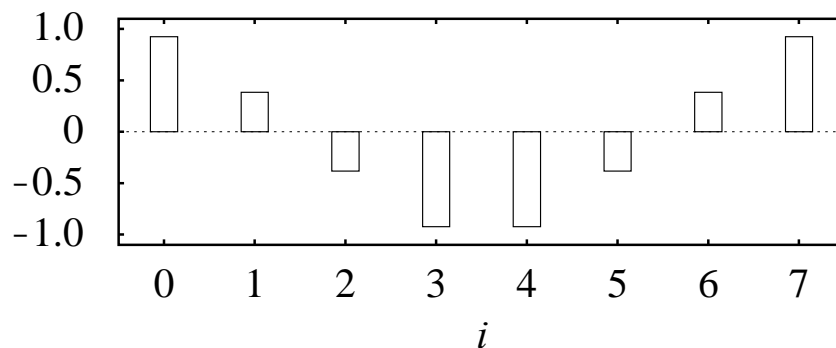
The 0th basis function ($u = 0$)



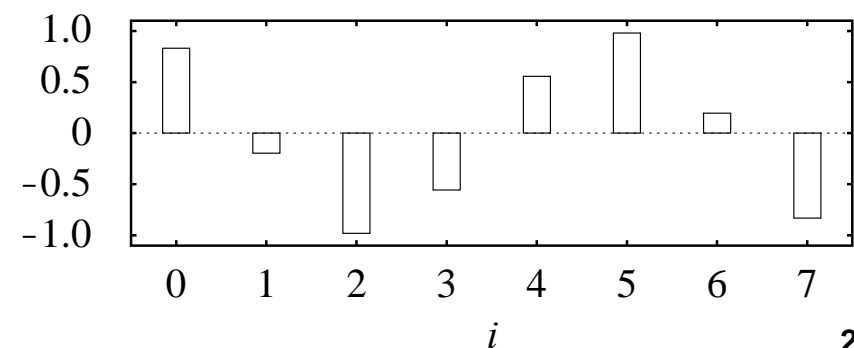
The 1st basis function ($u = 1$)



The 2nd basis function ($u = 2$)

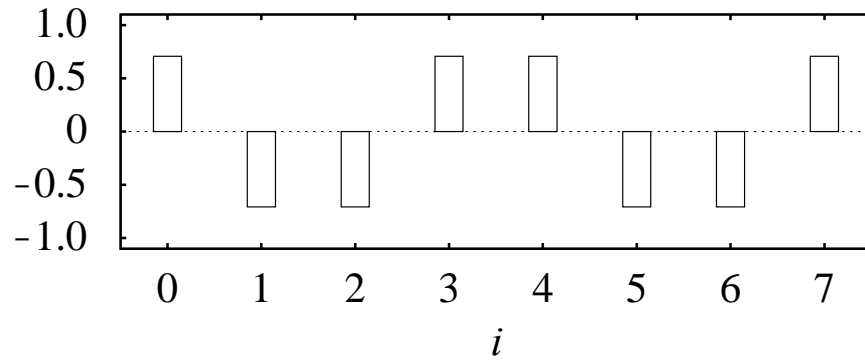


The 3rd basis function ($u = 3$)

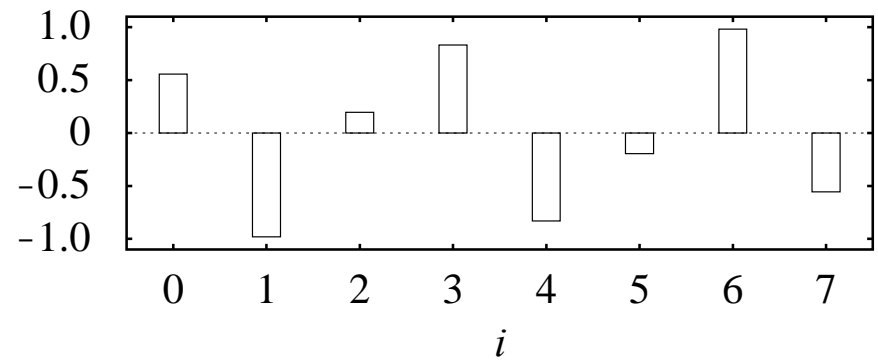


The 1D DCT Basis Functions

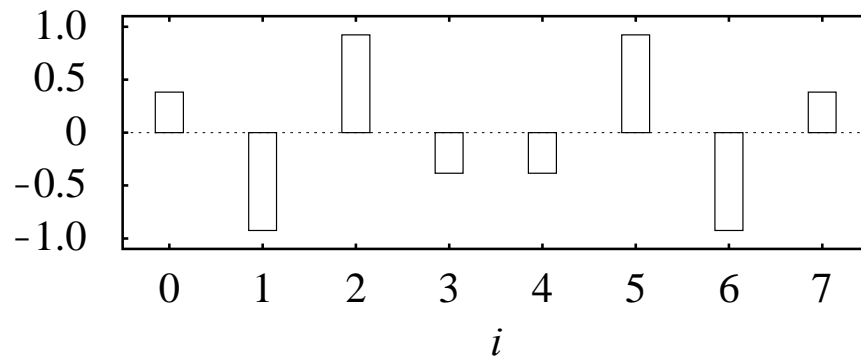
The 4th basis function ($u = 4$)



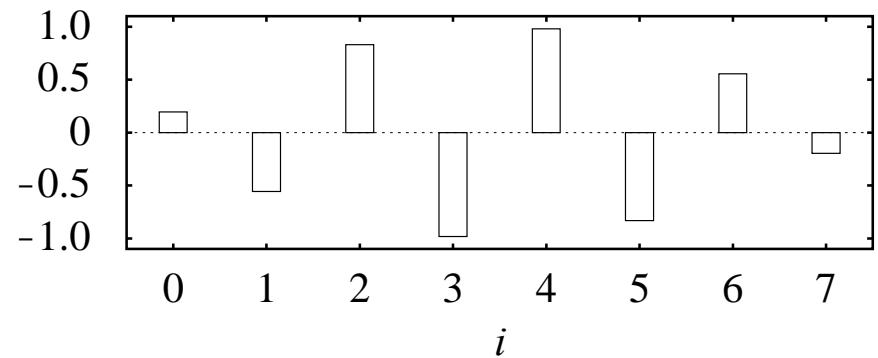
The 5th basis function ($u = 5$)



The 6th basis function ($u = 6$)



The 7th basis function ($u = 7$)



Example of 1D DCT

- Discrete Cosine Transform (DCT)
 - ✓ F is the result of a linear transform T of the input vector f in such a way that the components of F are much less correlated
- Example 3(textbook-p159-Ex8.1) of transform coding
 - ✓ Input vector X : $f \{100 \ 100 \ 100 \ 100 \ 100 \ 100 \ 100 \ 100\}$
 - ✓ Transform coding T : 1D DCT

$$F(u) = \frac{C(u)}{2} \sum_{i=0}^7 \cos \frac{(2i+1)u\pi}{16} f(i)$$

- ✓ Transformed vector Y : $F \{283 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0\}$

Example of 1D Invert DCT

- ◆ Input vector $X: f$
 - {100 100 100 100 100 100 100 100}
- ◆ Transformed vector $Y: F$
 - {283 0 0 0 0 0 0 0}
- ◆ Invert DCT

$$\tilde{f}(i) = \sum_{u=0}^7 \frac{C(u)}{2} \cos \frac{(2i+1)u\pi}{16} F(u)$$

- ◆ Output data $X': f'$
 - {100 100 100 100 100 100 100 100}

Definition of 2D DCT

- ◆ Given an input function $f(i, j)$ over two integer variables i and j , the 2D DCT transforms it into a new function $F(u, v)$, with integer u and v running over the same range as i and j . The general definition of the transform is:

$$F(u, v) = \frac{2 C(u) C(v)}{\sqrt{MN}} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \cos \frac{(2i+1) \cdot u\pi}{2M} \cdot \cos \frac{(2j+1) \cdot v\pi}{2N} \cdot f(i, j)$$

- $i, u = 0, 1 \dots M-1$;
- $j, v = 0, 1 \dots N-1$;
- The constants $C(u)$ and $C(v)$ are determined by:

$$C(\xi) = \begin{cases} \frac{\sqrt{2}}{2} & \text{if } \xi = 0, \\ 1 & \text{otherwise.} \end{cases}$$

Definition of 2D DCT

- ◆ 2D Inverse Discrete Cosine Transform (2D IDCT)

$$\tilde{f}(i, j) = \sum_{u=0}^7 \sum_{v=0}^7 \frac{C(u) C(v)}{4} \cos \frac{(2i+1)u\pi}{16} \cos \frac{(2j+1)v\pi}{16} F(u, v)$$

2D DCT and IDCT with 8*8 Numbers

- ◆ Consider an image of (8*8)
- ◆ 2D DCT

$$F(u, v) = \frac{C(u)C(v)}{4} \sum_{i=0}^7 \sum_{j=0}^7 \cos \frac{(2i+1)u\pi}{16} \cos \frac{(2j+1)v\pi}{16} f(i, j)$$

- ◆ 2D IDCT

$$\tilde{f}(i, j) = \sum_{u=0}^7 \sum_{v=0}^7 \frac{C(u)C(v)}{4} \cos \frac{(2i+1)u\pi}{16} \cos \frac{(2j+1)v\pi}{16} F(u, v)$$

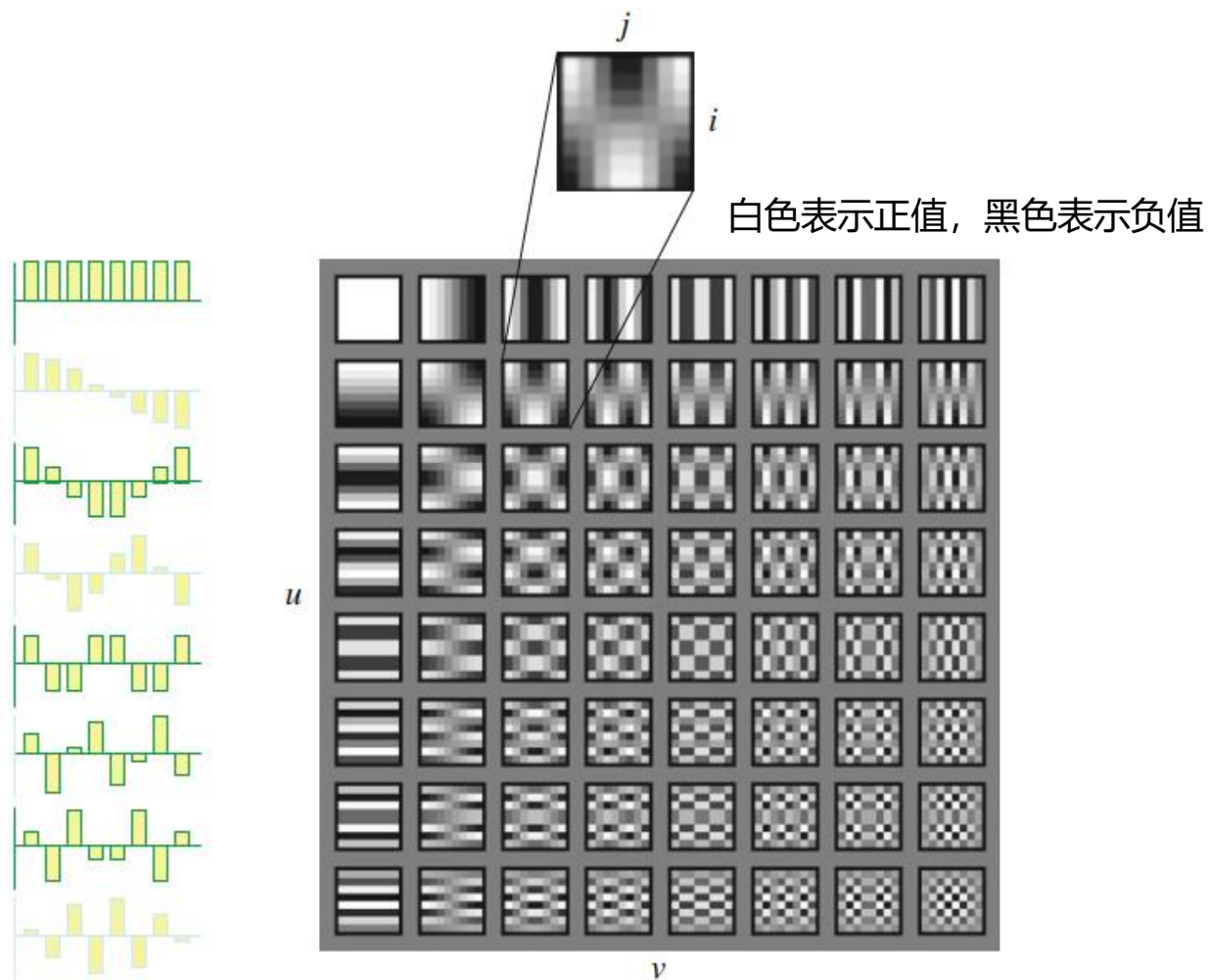


图 一维和二维DCT基示意图

Example of 2D DCT



8*8

Matlab demo: DCTTrans.m

Example of 2D DCT

例4

139	144	149	153	155	155	155	155
144	151	153	156	159	156	156	156
150	155	160	163	158	156	156	156
159	161	162	160	160	159	159	159
159	160	161	162	162	155	155	155
161	161	161	161	160	157	157	157
162	162	161	163	162	157	157	157
162	162	161	161	163	158	158	158

一个实际8*8图像块

236	-1.0	-12	-5.2	2.1	-1.7	-2.7	-1.3
-22	-18	-6.2	-3.2	-2.9	-0.1	0.4	-1.2
-11	9.3	-1.6	1.5	0.2	-0.9	-0.6	-0.1
-7.1	-1.9	0.2	1.5	1.6	-0.1	0	0.3
-0.6	-0.8	1.5	1.6	-0.1	-0.7	0.6	1.3
-1.8	-0.2	1.6	-0.1	-0.8	1.5	1	-1
-1.3	-0.4	-0.3	-1.5	0.5	1.7	1.1	-0.8
-2.6	1.6	-3.8	-1.8	1.9	1.2	-0.6	-0.4

图像经过DCT变换后的系数