电子科技大学 光电科学与工程学院
SCHOOL OF OPTOELECTRONIC SCIENCE AND ENGINEERING OF UESTC

第六章 图像编码与压缩

Chapter06 Image coding and compression

彭真明
E-mail:zmpeng@uestc.edu.cn
pengzm_ioe@163.com

电子科技大学光电信息学院

2018.11.14，二教106/107，沙河校区

内容提要

- 编码及信息论概述
- 行程编码
- 霍夫曼编码
- 预测编码
- 变换编码
- 压缩标准简介*

一、编码及信息论概述

- 图像压缩的必要性

一幅512×512的灰度图像的比特数为：

$$512\times512\times8 = 2,097,152 \text{ bit}$$
$$= 262,144 \text{ byte}$$
$$= 256 \text{ k}$$。

可见，数字图像通常要求很大的比特数，这给图像的传输和存储带来一定的困难。要占用很多的资源，花很高的费用。

一、编码及信息论概述

- 图像压缩的必要性

一部90分钟的彩色电影，每秒放映24帧。把它数字化，每帧512x512象素，每象素的R、G、B三分量分别占8 bit，总比特数为：

$$90 \times 60 \times 24 \times 3 \times 512 \times 512 \times 8\text{bit} = 97{,}200\ (\text{M})$$

若一张CD光盘可存600兆字节数据，这部电影图像（不包括声音）就需要约160张CD光盘用来存储。

一、编码及信息论概述

- 编码

编码是用符号元素（字母、数字、比特和类似的符号等）表示信号、消息或事件的过程。

每个信息或事件被赋予一个编码符号序列，称之为码字。

每个码字中符号的数量就是该码字的长度，简称码长。

一、编码及信息论概述

- 图像压缩的可能性——数据冗余

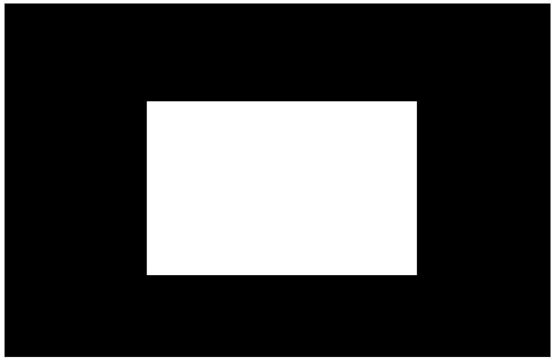
编码冗余：用于表示的8bit编码所包含的比特数要比表示该灰度所需要的比特数多。

空间/时间冗余：空间像素相关性；视频之帧间画面的相关性等；

不相关的信息：视觉可忽略或与用途无关的信息。

例析1：编码冗余

如果用8 bit表示下面图像的像素，可以说该图像存在着编码冗余。



因为该图像的像素只有两个灰度，用1 bit即可表示。

例析2：空间冗余

对于一幅图像，很多单个像素对视觉的贡献是冗余的。这可以建立在对邻域值预测的基础上。

例如：

原图像数据：

250 253 251 252 250 → 40bit

压缩后数据：

250 3 1 2 0 → 14bit

例析3：不相关信息

- 视觉冗余

一些信息在一般视觉处理中比其它信息的相对重要程度要小，这种信息就被称为<u>视觉冗余</u>。

一、编码及信息论概述

- 图像编码及分类

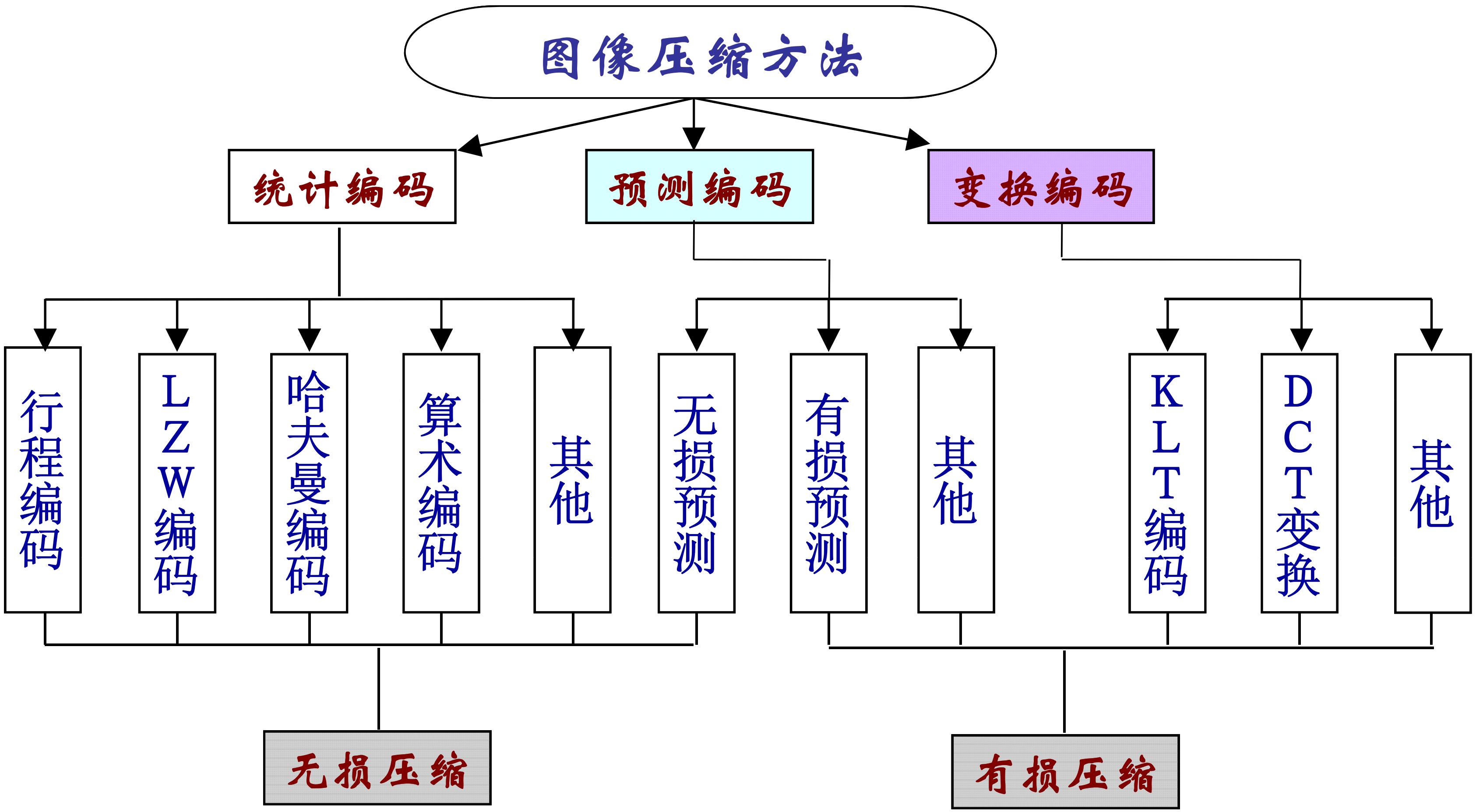
无失真编码（无损压缩、可逆压缩）是一种经编、解码后图像不会产生失真的编码方法。可重建图像，但压缩比不大；

有失真编码（有损压缩、不可逆压缩）解码时无法完全恢复原始图像，压缩比大但有信息损失。

失　真：原图像与解码输出图像之间的随机误差；

压缩比：原图像比特数与压缩后图像比特数之比。

一、编码及信息论概述


图像压缩方法
统计编码
预测编码
变换编码
行程编码
LZW编码
哈夫曼编码
算术编码
其他
无损预测
有损预测
其他
KLT编码
DCT变换
其他
无损压缩
有损压缩

一、编码及信息论概述

- **信源的定义**

信源即能够产生信息的事物。

数学上，信源是一概率场。

若信源可能产生的信息是$s_1, s_2, \cdots, s_n$，这些信息出现的概率分别是 $p(s_1), p(s_2), \cdots, p(s_n)$ 。

则该信源可表示为：

$$\begin{Bmatrix} s_1, & s_2, & \cdots & , s_n \\ p(s_1), & p(s_2), & \cdots, & p(s_n) \end{Bmatrix}$$

一、编码及信息论概述

■ 信息量和熵

信息量：从N个相等可能发生的事件中，选出其中一个事件所需的信息度量，称为信息量。

例如：要辨识1到32中选定的某一个数，可先提问："是否大于16？"，得到回答就消去半数可能事件。每提问一次得到回答，可以得到1bit信息量(二进制位)。

这里共需5次，因此所需的信息量为：

$$\log_2 32 = -\log_2 \frac{1}{32} = -\log_2 p(s) = 5$$

一、编码及信息论概述

- **信息量的计算**

从 N 个数选定一个数 s 的概率为 $p(s)$，且等概率，$p(s)=1/N$。

$$I(s)=\log_2 N=-\log_2\frac{1}{N}=-\log_2 p(s)=I[p(s)]$$

- **熵的计算**

设信源符号表为 $\mathbf{s}=\{s_1, s_2, \ldots, s_q\}$，其概率分布为 $P(\mathbf{s})=\{p(s_1), p(s_2), \ldots, p(s_q)\}$，则信源的熵为：

$$H(\mathbf{s})=-\sum_{i=1}^{q}p(s_i)\log_2 p(s_i)=\sum_{i=1}^{q}p(s_i)I[p(s_i)]$$

一、编码及信息论概述

- **熵的作用**

熵表示信源中消息的**平均信息量**。在不考虑消息间的相关性时，是无失真编码平均长度比特数的下限。

例：信源 $S=\begin{Bmatrix} s_1, s_2, s_3, s_4 \\ 1/2, 1/4, 1/8, 1/8 \end{Bmatrix}$　　$H(s)=$ 

$$H(s)=-\sum_{i=1}^{k} p(s_i)\log_2 p(s_i)=\frac{7}{4}$$

说明：该信源编码平均码长最短情况下为7/4，不能再小，否则就会引起错误。平均码长比此数大许多时，就表明压缩还有待改进。

一、编码及信息论概述

- 熵的性质

1) 熵是一个非负数，即总有$H(s)\geq 0$。

2) 当其中一个符号s_j的出现概率$p(s_j)=1$时，其余符号$s_i(i\neq j)$的出现概率$p(s_i)=0$，$H(s)=0$。

3) 当各个s_i出现的概率相同时，则最大平均信息量为$\log_2 N$，N为可能发生的事件数。

4) 熵值总有，$H(s)\leq \log_2 N$。

一、编码及信息论概述

- 图像的熵

1) s作为灰度，共q级，出现概率均等时，$p(s_i)=1/q$，

$$H(s)=-\sum_{i=1}^{q}\frac{1}{q}\log_2\frac{1}{q}=\log_2 q$$

2) 当灰度只有两级时，即$s_i=0,1$，且0出现概率为p_1，1出现概率为$p_2=1-p_1$，其熵：

$$H(s)=p_1\log_2\frac{1}{p_1}+(1-p_1)\log_2\frac{1}{1-p_1}$$

一、编码及信息论概述

- 图像的熵

3) 对8位图像，s作为灰度，共256级，其熵为：

$$H(\mathbf{s})=-\sum_{i=0}^{255}p(i)\log_2 p(i)$$

4) 当图像由单一灰度级组成时（即灰度均匀分布图像），其熵为：

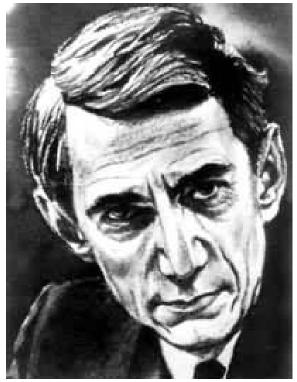
$$H(\mathbf{s})=-\sum_{i=0}^{255}p(i)\log_2 p(i)=0$$

一、编码及信息论概述

- 无失真编码定理

在无干扰条件下，存在一种无失真的编码方法，使编码的平均长度$\overline{L}$与信源的熵$H(s)$任意地接近，即

$$\overline{L}=H\left(s\right)+\varepsilon$$



Claude Shannon
(1916-2001)

其中，ε为任意小的正数。以$H(s)$为其下限，$\overline{L}\geqslant H(s)$，即香农(Shannon)无干扰编码定理。

一、编码及信息论概述

- 压缩率

若原始图像的平均比特数为b，编码后的平均比特数为b_d，则压缩率/比定义为：

$$C = b/b_d \quad \rightarrow \quad \tilde{C} = \cancel{\frac{b-b_d}{b} = 1-\frac{b_d}{b}}$$

由Shannon定理，无失真编码**最大数据压缩比**为：

$$C_{\max} = \frac{b}{H(s)+\varepsilon} \approx \frac{b}{H(s)}$$

一、编码及信息论概述

- 编码效率

$$\eta=\frac{H(s)}{\overline{L}},\qquad \text{or}\qquad \cancel{\eta=\frac{1}{1+R}}$$

- 冗余度

压缩率/比

$$R_D=1-1/C,\quad \text{or}\ R=1-\eta,\quad \cancel{R=1/\eta-1}$$

R接近于0，或编码效率接近于1的编码称为高效码。

一、编码及信息论概述

- 保真度(Fidelity)

✓ 总误差(Total error between the two images)

$$e = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left[f(x,y)-\hat{f}(x,y)\right]$$

✓ 均方差(MSE)

$$MSE = \sqrt{\frac{1}{MN}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left[f(x,y)-\hat{f}(x,y)\right]^2}$$

✓ 信噪比(SNR)

Signals　　Noises

$$SNR = \sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\hat{f}(x,y)^2 \Big/ \sum_{x=0}^{M-1}\sum_{y=0}^{N-1}\left[f(x,y)-\hat{f}(x,y)\right]^2$$

主观

客观

Next Sect...

内容提要

- ◆编码及信息论概述
- ◆行程编码
- ◆霍夫曼编码
- ◆预测编码
- ◆变换编码
- ◆压缩标准简介*

二、行程编码

- **概念**

行程编码(Run Length Encoding，RLE)是一种消除空间冗余的数据压缩方法。

行程：沿行(或列)具有相同灰度值的像素序列。

- **编码思想**

去除空间像素冗余。即用行程的灰度和行程的长度代替行程本身。

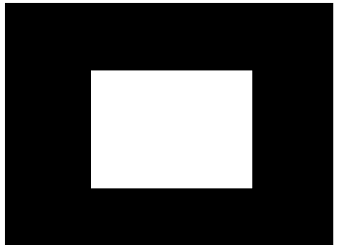
二、行程编码

例：设重复次数为 iC, 重复像素值为 iP，则

编码为：iCiP iCiP iCiP

编码前：aaaaaaabbbbbbcccccccc

编码后：7a6b8c



- 对于有大面积色块的图像，压缩效果很好。
- 对于杂乱图像，压缩效果不好；最坏情况，会加倍数据。

Next Sect...

主要内容

- 编码及信息论概述
- 行程编码
- 霍夫曼编码
- 预测编码
- 变换编码
- 压缩标准简介*

三、霍夫曼(Huffman)编码

- 简介

霍夫曼(Huffman)编码是可变字长编码(Variable length coding,VLC)的一种。

该方法由Huffman(1952)提出，它依据字符出现概率来构造异字头的平均长度最短的码字，被称作Huffman编码，有时称之为最佳编码。

三、霍夫曼(Huffman)编码

- 基本思想

通过减少编码冗余来达到数据压缩的目的。

- 统计各个符号出现的概率。
- 建立一个概率统计表。

一般规定：
最常出现(概率大的)的符号用最短的编码；最少出现的符号用最长的编码。

例析：Huffman编码

例题1： 已知信号源为 $s=\{s_1, s_2, s_3, s_4, s_5, s_6\}$，其概率分布为$p_1$=0.4,$p_2$=0.3,$p_3$=0.1，$p_4$=0.1，$p_5$=0.06，$p_6$=0.04。试分别计算：

(1) 最佳Huffman码；
(2) 该信源的平均码长；
(3) 编码效率η和冗余度R。

例析：Huffman编码

计算步骤：

① 将信源符号按出现概率从大到小排成一列，然后把最末尾的两个符号的概率相加，合成一个概率。

② 把这个符号的概率与其余符号的概率按从大到小排列，然后再把最末两个符号的概率加起来，合成一个概率。

③ 重复上述做法，直到最后剩下两个概率为止。

④ 从最后一步剩下的两个概率开始逐步向前编码，每步只需对两个分支各赋予一个二进制码。例如，对概率大的赋予码元0，对概率小的赋予码元1。

例析：Huffman编码

输入	概率	自然码
S_1	0.4	000
S_2	0.3	001
S_3	0.1	010
S_4	0.1	011
S_5	0.06	100
S_6	0.04	101

例析：Huffman编码

输入	概率	第①步
S_1	0.4	0.4
S_2	0.3	0.3
S_3	0.1	0.1
S_4	0.1	0.1
S_5	0.06	0.1
S_6	0.04	

例析：Huffman编码

输入	概率	第①步	第②步
S_1	0.4	0.4	0.4
S_2	0.3	0.3	0.3
S_3	0.1	0.1	0.2
S_4	0.1	0.1	0.1
S_5	0.06	0.1	
S_6	0.04		

例析：Huffman编码

输入	概率	第①步	第②步	第③步
S_1	0.4	0.4	0.4	0.4
S_2	0.3	0.3	0.3	0.3
S_3	0.1	0.1	0.2	0.3
S_4	0.1	0.1	0.1	
S_5	0.06	0.1		
S_6	0.04			

例析：Huffman编码

输入	概率	第①步	第②步	第③步	第④步
S_1	0.4	0.4	0.4	0.4	0.6
S_2	0.3	0.3	0.3	0.3	0.4
S_3	0.1	0.1	0.2	0.3	
S_4	0.1	0.1	0.1		
S_5	0.06	0.1			
S_6	0.04				

例析：Huffman编码

输入	概率	第①步	第②步	第③步	第④步
S_1	0.4	0.4	0.4	0.4	0.6 0
S_2	0.3	0.3	0.3	0.3 0	0.4 1
S_3	0.1	0.1	0.2 0	0.3 1	
S_4	0.1	0.1 0	0.1 1		
S_5	0.06 0	0.1 1			
S_6	0.04 1				

例析：Huffman编码

输入	概率	第①步	第②步	第③步	第④步
S_1	0.4	0.4	0.4	0.4	0.6 0
S_2	0.3	0.3	0.3	0.3 0	0.4 1
S_3	0.1	0.1	0.2 0	0.3 1	
S_4	0.1	0.1 0	0.1 1		
S_5	0.06 0	0.1 1			
S_6	0.04 1				

$$S_1 = 1$$

例析：Huffman编码

输入	概率	第①步	第②步	第③步	第④步
S_1	0.4	0.4	0.4	0.4	0.6 0
S_2	0.3	0.3	0.3	0.3 0	0.4 1
S_3	0.1	0.1	0.2 0	0.3 1	
S_4	0.1	0.1 0	0.1 1		
S_5	0.06 0	0.1 1			
S_6	0.04 1				

$$S_2 = 00$$

例析：Huffman编码

输入	概率	第①步	第②步	第③步	第④步
S_1	0.4	0.4	0.4	0.4	0.6 0
S_2	0.3	0.3	0.3	0.3 0	0.4 1
S_3	0.1	0.1	0.2 0	0.3 1	
S_4	0.1	0.1 0	0.1 1		
S_5	0.06 0	0.1 1			
S_6	0.04 1				

$$S_3 = 011$$

例析：Huffman编码

输入	概率	第①步	第②步	第③步	第④步
S_1	0.4	0.4	0.4	0.4	0.6 0
S_2	0.3	0.3	0.3	0.3 0	0.4 1
S_3	0.1	0.1	0.2 0	0.3 1	
S_4	0.1	0.1 0	0.1 1		
S_5	0.06 0	0.1 1			
S_6	0.04 1				

$$S_4 = 0100$$

例析：Huffman编码

输入	概率	第①步	第②步	第③步	第④步
S_1	0.4	0.4	0.4	0.4	0.6 0
S_2	0.3	0.3	0.3	0.3 0	0.4 1
S_3	0.1	0.1	0.2 0	0.3 1	
S_4	0.1	0.1 0	0.1 1		
S_5	0.06 0	0.1 1			
S_6	0.04 1				

$$S_5 = 01010$$

例析：Huffman编码

输入	概率	第①步	第②步	第③步	第④步
S_1	0.4	0.4	0.4	0.4	0.6 0
S_2	0.3	0.3	0.3	0.3 0	0.4 1
S_3	0.1	0.1	0.2 0	0.3 1	
S_4	0.1	0.1 0	0.1 1		
S_5	0.06 0	0.1 1			
S_6	0.04 1				

$$S_6 = 01011$$

例析：Huffman编码

- 平均码长 → 自然码长 $m = 3$

$$\underline{L} = 0.4\times1+0.3\times2+0.1\times3+0.1\times4+0.06\times5+0.04\times5$$

$\approx$ **2.2** (bit/像素)

- 信源的熵

$$H(s) = -(0.4\times\log2(0.4)+0.3\times\log2(0.3)+0.1\times\log2(0.1)+0.1\times\log2(0.1)+0.06\times\log2(0.06)+0.04\times\log2(0.04))$$

$\approx$ **2.14** (bit/像素)

- 压缩结果

压缩比(C)： $= m / \underline{L} = 3/2.2 \approx$ **1.364**

编码效率(η)： $= H(s) / \underline{L} = 2.14/2.2 \approx$ **97.3%**

冗余度(R)： $= 1 - \eta \approx 0.027 =$ **2.7 %**

例析：Huffman编码

例题2： 对下面的一首英文歌词进行Huffman编码。

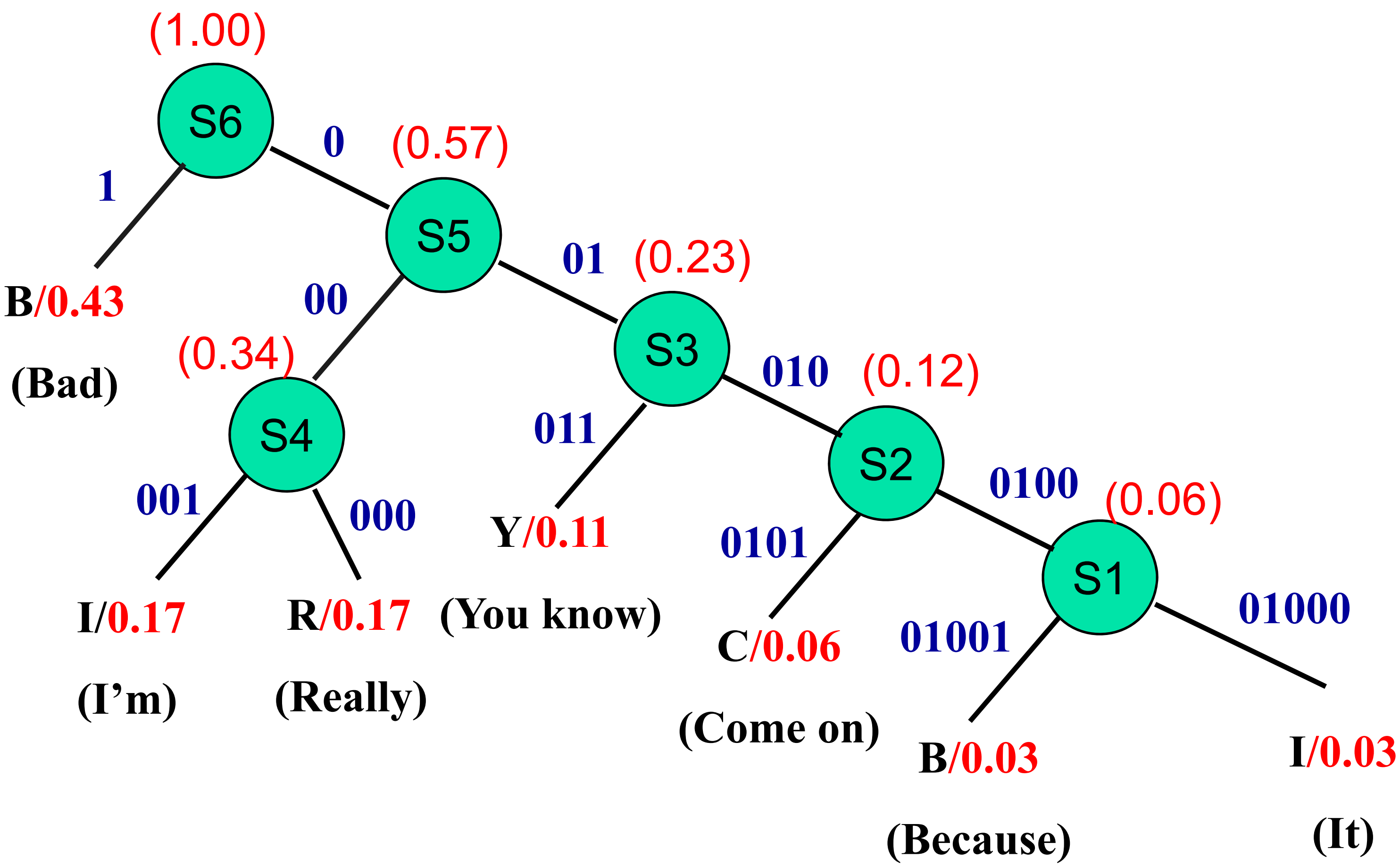
Because I’m bad, I am bad — come on;
Bad, bad — really, really bad;
You know I’m bad, I'm bad —
Bad, bad — really, really bad;
You know I’m bad, I'm bad — come on, you know;
Bad, bad — really, really bad.

例析：Huffman编码

表1 歌词短语及概率统计

短语	符号(首字母)	频率(次)	概率
Because	B	1	1/35 = 0.03
I’m	I	6	6/35 = 0.17
Bad	B	15	15/35 = 0.43
Come on	C	2	2/35 = 0.06
It	I	1	1/35 = 0.03
Really	R	6	6/35 = 0.17
You know	Y	4	4/35 = 0.11

编码的树结构表示


(1.00)
S6
1
0
(0.57)
S5
B/0.43
(Bad)
00
01
(0.23)
S3
(0.34)
S4
010
(0.12)
011
S2
001
000
Y/0.11
0100
(0.06)
0101
S1
I/0.17
R/0.17
(You know)
C/0.06
01001
01000
(I'm)
(Really)
(Come on)
B/0.03
I/0.03
(Because)
(It)

例析：Huffman编码

表2 短语、码长及编码结果统计

短语	符号	频率	概率	码长	编码结果
Because	B	1	0.03	5	01001
I’m	I	6	0.17	3	001
Bad	B	15	0.43	1	1
Come on	C	2	0.06	4	0101
It	I	1	0.03	5	01000
Really	R	6	0.17	3	000
You know	Y	4	0.11	3	011

例析：Huffman编码

平均码长：

$\underline{L}$ = 0.03*5+0.17*3+0.43*1+0.06*4+0.03*5+0.17*3 + 0.11*3 = 2.32

信源的熵：

$H(s)$ = -(0.03*log2(0.03)+0.17*log2(0.17) +0.43*log2(0.43) + 0.06*log2(0.06)+0.03*log2(0.03) +0.17*log2(0.17) + 0.11*log2(0.11)) = 2.29

$C, \eta, R =$

编码前:35×3 = 105,编码后: 81位; C=105 /81 ≈ 1.296

$\eta = ?$； $R = ?$

Next Sect...

内容提要

- 编码及信息论概述
- 行程编码
- 霍夫曼编码
- 预测编码
- 变换编码
- 压缩标准简介*

四、预测编码

- **预测编码**

时间域(Time domain)编码方法。

- 思路：利用前面已经出现过的符号来预测当前的符号，然后将实际上的符号与预测相减得到预测误差值。
- 预测编码的好处在于预测误差值的范围比原信号的数字范围小，如果预测足够精确的话。
- 通常,会对预测误差值进一步编码（Quantization）。

<u>**可分为有损和无损预测，但预测本身不会造成失真**</u>。

四、预测编码

1.无损预测编码(Lossless Predictive Coding)

- 基本思想
  - 相邻像素的信息有冗余，当前像素值可以用先前的像素值来获得。
  - 用当前像素值f_n，通过预测器得到一个预测值$\hat{f}_n$，对当前值和预测值求差。对差编码，作为压缩数据流中的下一个元素。

五、预测编码

例题1： 利用前1个像素进行预测(1阶)

100	102	101	100	100
101	100	102	102	100
100	103	100	102	101
100	100	100	102	100
101	101	100	100	100

原像素值

100	2	-1	-1	0
1	-1	2	0	-2
-1	3	-3	2	-1
0	0	0	2	-2
1	0	-1	0	0

预测值

四、预测编码

- 预测原理

一般情况下，利用前m个像素的线性组合来预测，即

$$f_n = round\left[\sum_{i=1}^{m}\alpha_i f_{n-i}\right]$$

预测器为：

$$\hat{f}_n(x,y) = round\left[\sum_{i=1}^{m}\alpha_i f(x\text{-}i,y)\right]$$

其中，*round*为取最近整数，α_i为预测系数(可为1/m)，*x*,*y*分别是列/行变量。

注意：前m个像素不能用此法编码，可用哈夫曼编码。

例析：预测编码

例题2： $f = \{154,159,151,149,139,121,112,109,129\}$

取 $m = 2,\ a = 1/2$

预测：

$$f2 = 1/2 \times (154+159) \approx 156,\quad e2 = 151-156 = \text{-}5$$
$$f3 = 1/2 \times (159+151) = 155,\quad e3 = 149-155 = \text{-}6$$
$$f4 = 1/2 \times (151+149) = 150,\quad e4 = 139-150 = \text{-}11$$
$$f5 = 1/2 \times (149+139) = 144,\quad e5 = 121-144 = \text{-}23$$
$$f6 = 1/2 \times (139+121) = 130,\quad e6 = 112-130 = \text{-}18$$
$$f7 = 1/2 \times (121+112) \approx 116,\quad e7 = 109-116 = \text{-}7$$
$$f8 = 1/2 \times (112+109) \approx 110,\quad e8 = 129-110 = 19$$

四、预测编码

- 编码步骤

第①步：压缩头处理。 → 如Huffman编码

第②步：预测值。 → 预测器设计

第③步：求预测误差值。

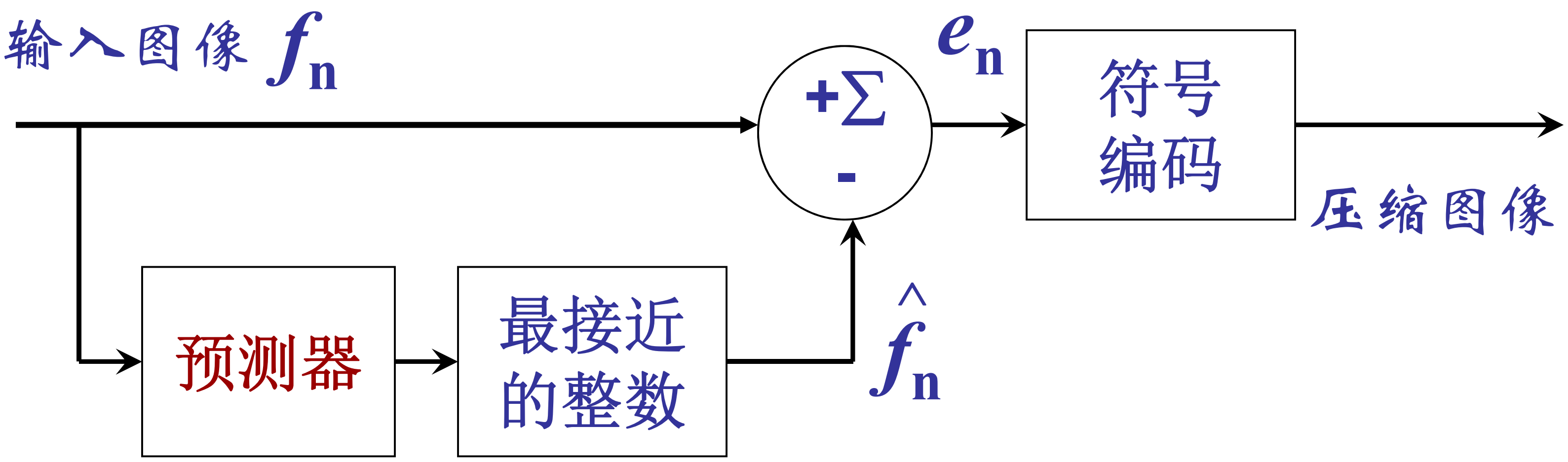
$$e(x,y)=f(x,y)-\hat{f}(x,y)$$

第④步：对误差e(x,y)编码，作为压缩值。

重复执行②、③、④步。

四、预测编码

- 编码原理图



四、预测编码

2. 有损预测编码

有损预测编码（Lossy Predictive Coding，LPC）

- ✓ 基本概念
- ✓ 量化器
- ✓ 编码原理

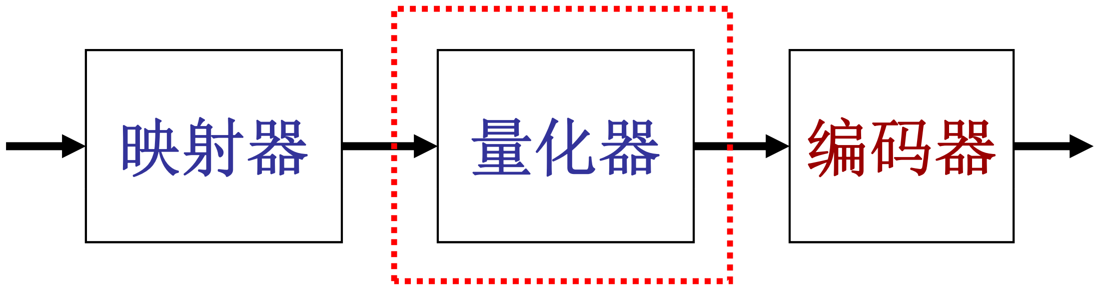
四、预测编码

- 基本概念
  - 通过牺牲图像的准确率来达到增大压缩率的目的。
  - 如果容忍解压后的结果中有一定的误差，那么压缩率可以显著提高。
  - 有损与无损压缩的本质区别在于：是否有量化器模块。

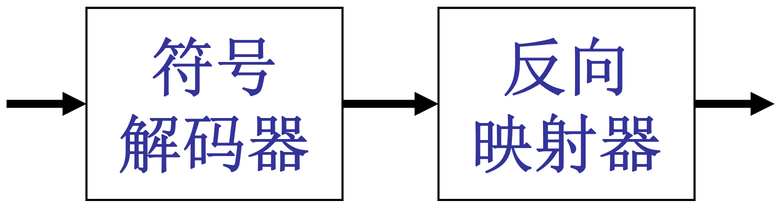
四、预测编码

数据源编/解码的一般模型

- 编码模型



- 解码模型



四、预测编码

- **量化器**
  - 减少数据量的最简单的办法是将图像量化成较少的灰度级，通过减少图像的灰度级来实现图像的压缩；
  - 该量化过程是不可逆的，因而解码时图像有损失。

例如：输入是256个灰度级，对灰度级量化后输出，只剩下4个层次，则数据量被大大减少。然而，这个过程是不可逆的。

四、预测编码

- 量化器定义

- 阶梯形量化函数 $t = q(s)$，是一个s的奇函数（即$q(-s) = -q(s)$），它可以通过$L/2$、s_i和t_i来完全描述。

- s_i 被称为量化器的决策级（阈值）；

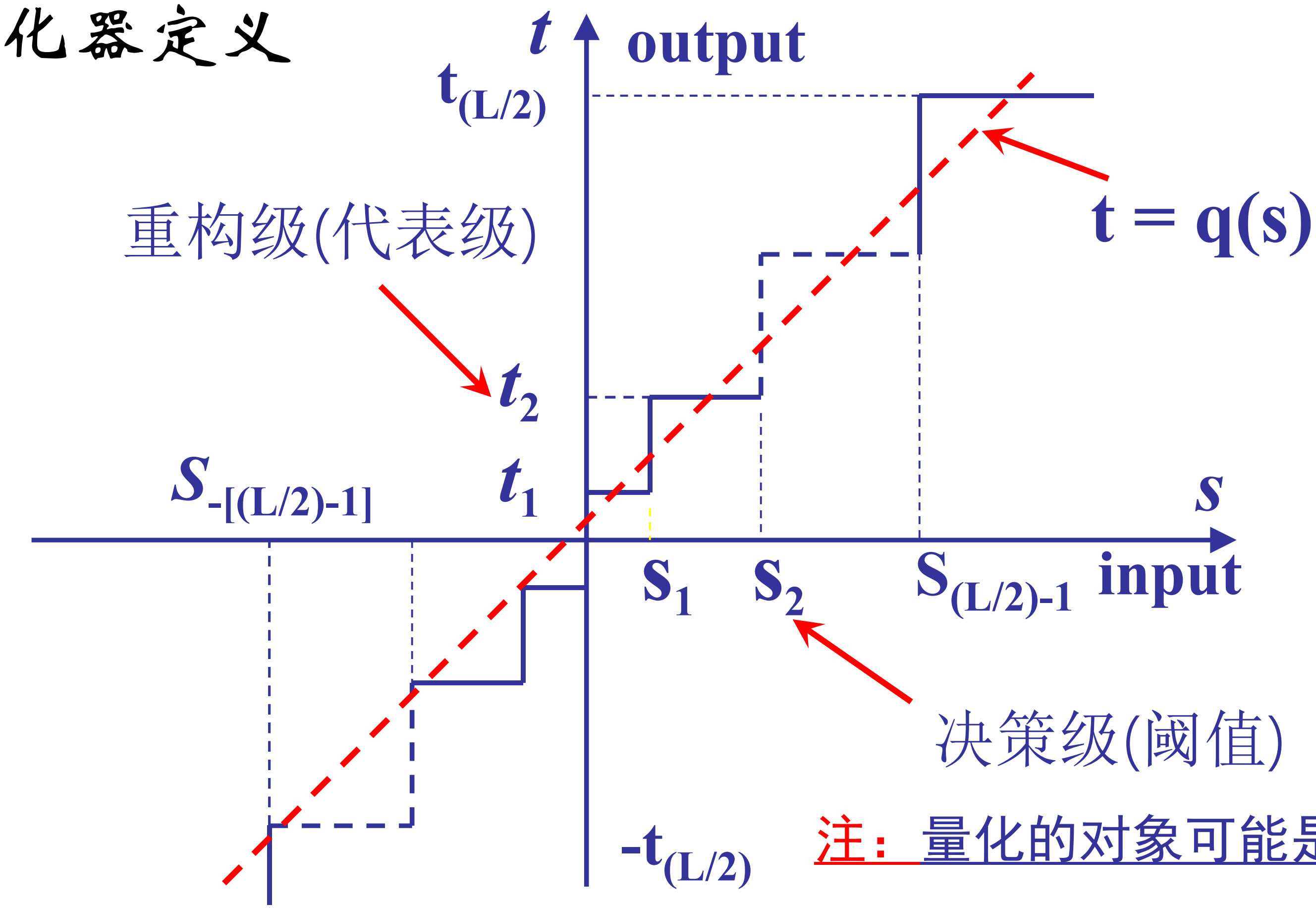
  t_i 被称为量化器的重构级（代表级）。

  L：是量化器的级数。

- 由于习惯的原因，s_i被认为是映射到t_i，如果它在半开区间$(s_i,s_{i+1}]$。

四、预测编码

- 量化器定义



注：量化的对象可能是负数

四、预测编码

- 算法演变——无损到有损

基本思想：对无损预测压缩的误差进行量化，通过消除视觉心理冗余，达到对图像进一步压缩的目的。

—— 引入量化 (Quantification)

四、预测编码

- 编/解码原理及过程

  - 将en量化： $\hat{e}_n = Q(e_n)$
  - 用： $f_n' = \boxed{\hat{e}_n} + \hat{f}_n$
  - 近似： $f_n \approx f_n'$
  - 编码： $\hat{e}_n = Q(f_n - \hat{f}_n)$
  - 解码： $f_n' = \hat{e}_n + \hat{f}_n$

- 无损预测

$$f_n' = e_n + \hat{f}_n$$

$$f_n = f_n'$$

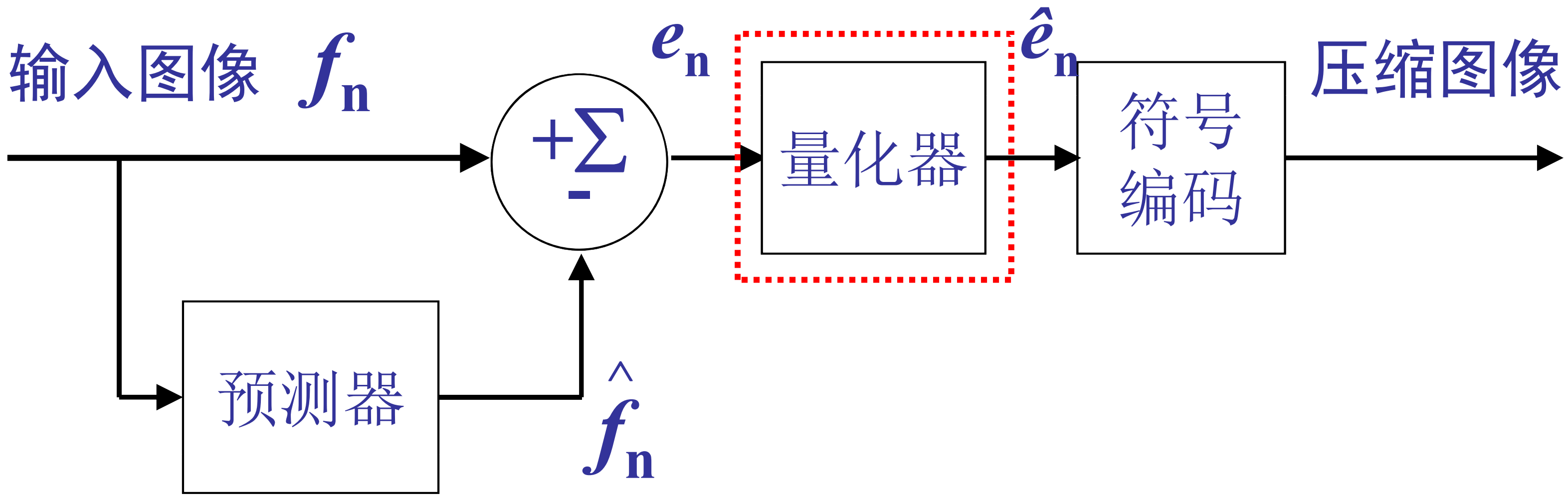
$$e_n = f_n - \hat{f}_n$$

$$f_n' = e_n + \hat{f}_n$$

四、预测编码

- 编码流程

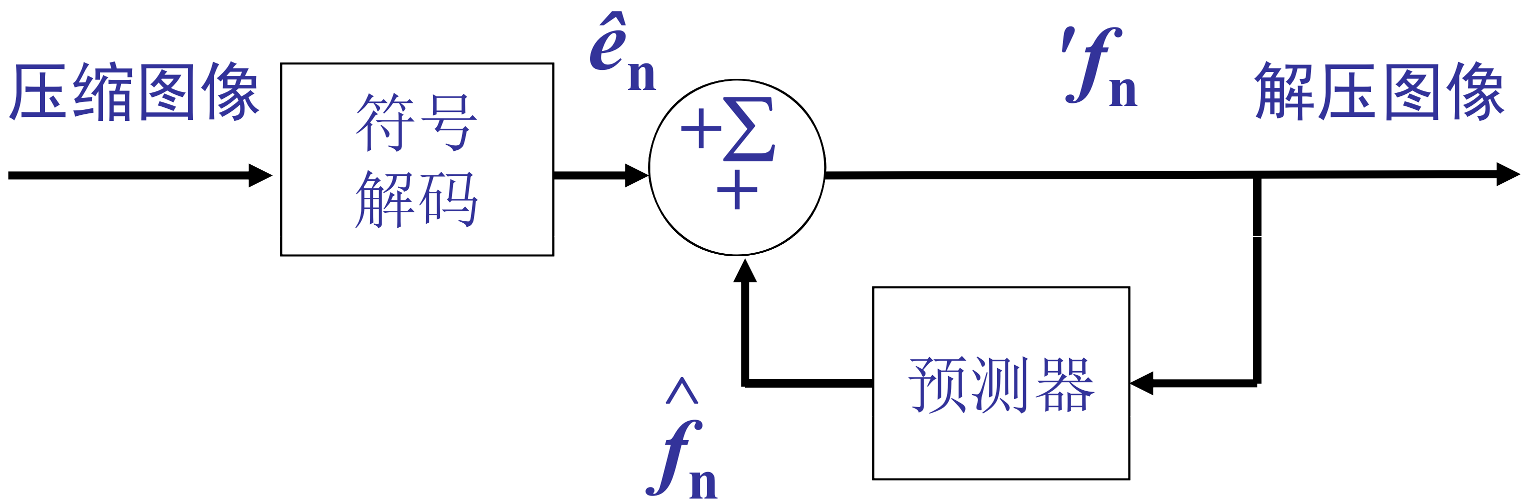
$$\hat{e}_n = Q\left(f_n - \hat{f}_n\right)$$



四、预测编码

- 解码流程

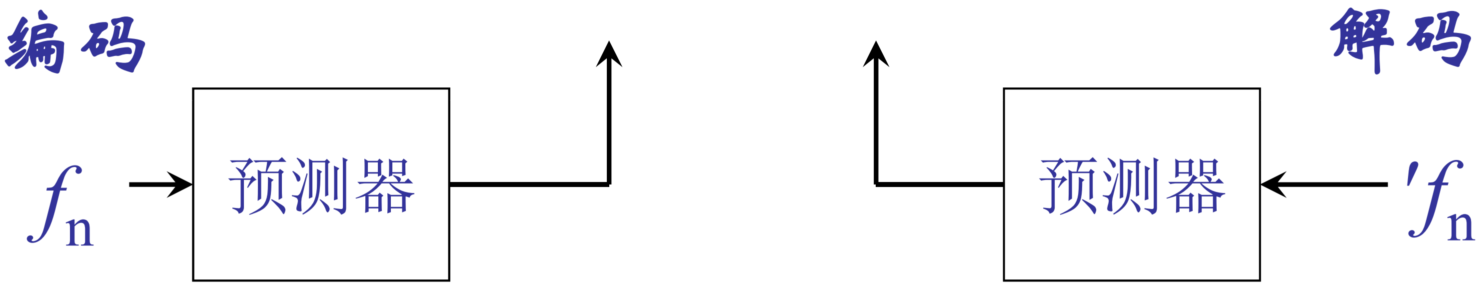
$$f_n' = \hat{e}_n + \hat{f}_n$$



四、预测编码

- **编码方案的修正**

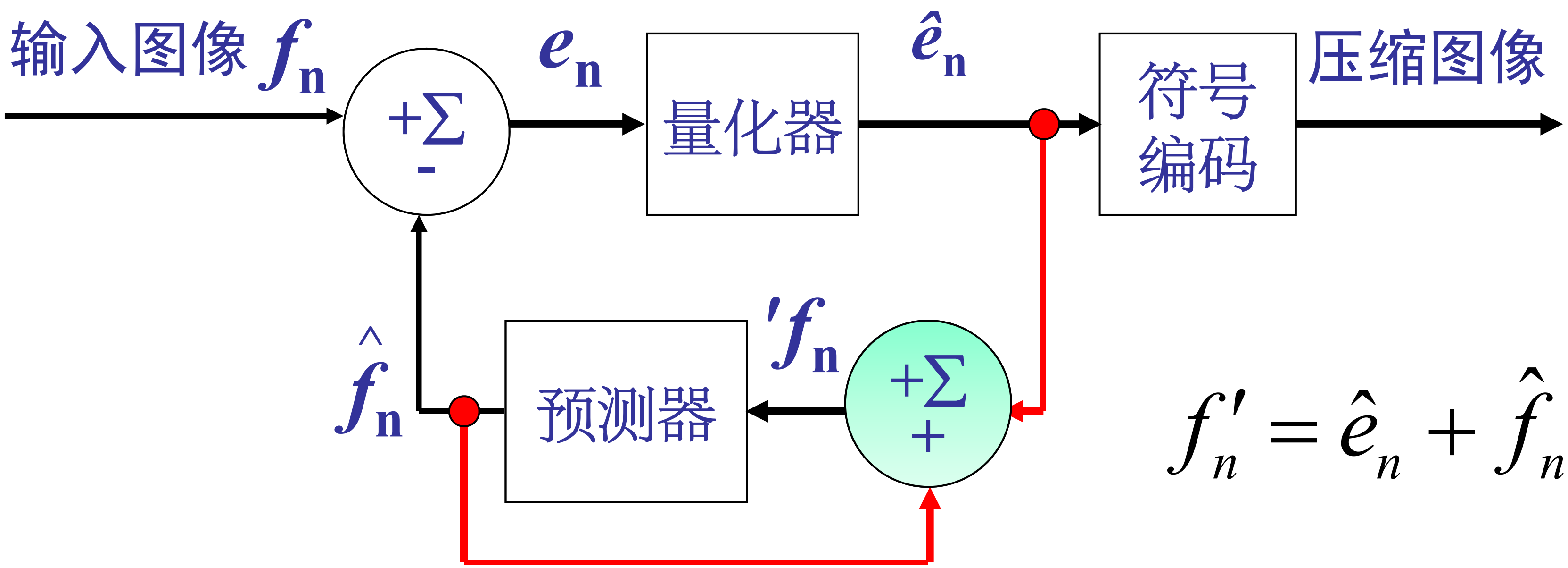
由于预测器的输入是f_n，而解压中的预测器的输入是f_n'，要使用相同的预测器，编码方案要进行修改。



四、预测编码

- 编码方案的修正

$$\hat{e}_n = Q\left(f_n - \hat{f}_n\right)$$



$$f'_n = \hat{e}_n + \hat{f}_n$$

四、预测编码

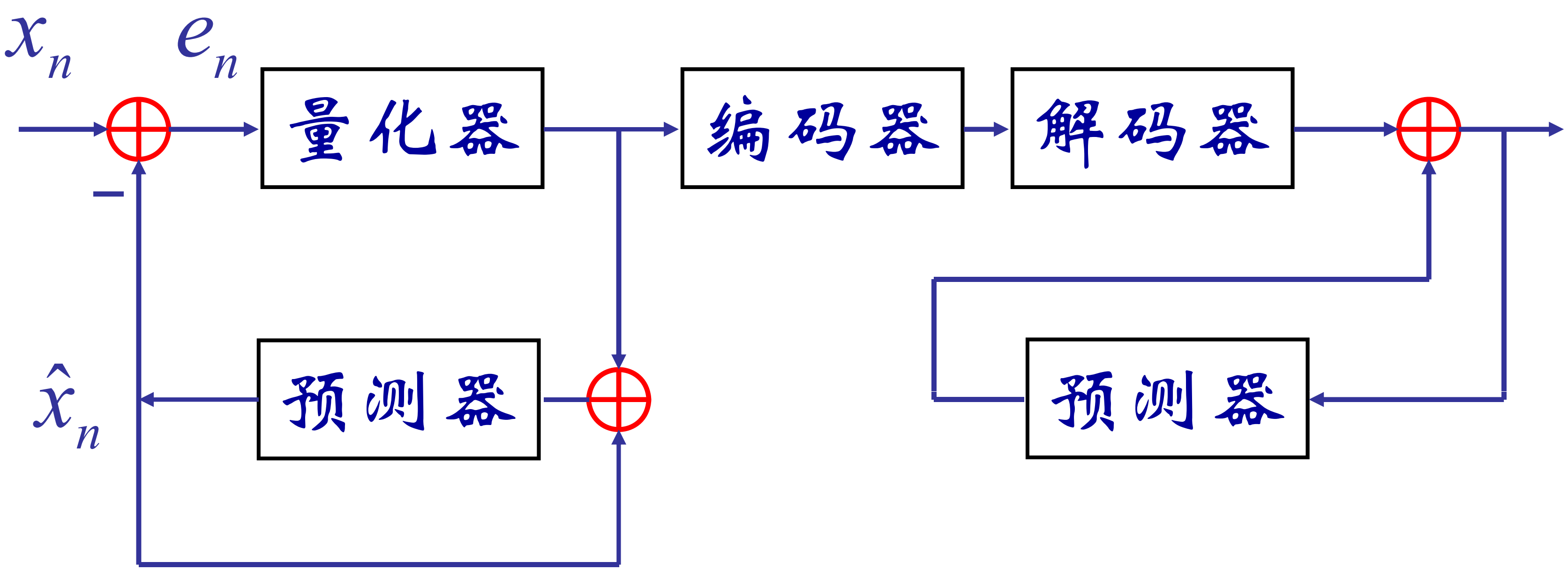
- DPCM简介

差分脉冲编码调制(**Differential Pulse Code Modulation, DPCM**)，采用反馈方法预测估值。

- 1950年，卡特勒申请专利;
- 1952年，获得批准;
- 1958年，格雷厄姆（Graham)开始计算机模拟研究编码方法;
- 1966年，奥尼尔（O’Neal)对电视信号预测编码进行理论分析以及计算机模拟;
- 1971年投入使用。

四、预测编码

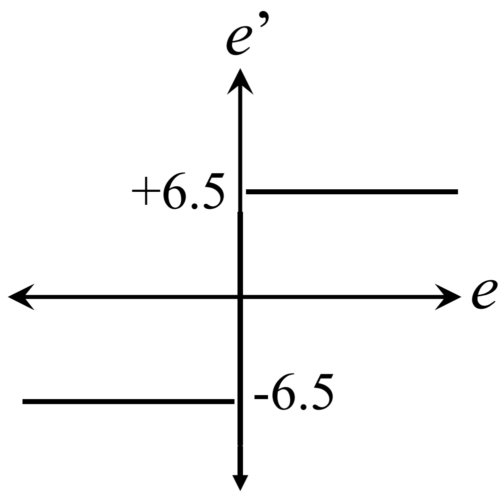
- DPCM编码原理图



四、预测编码

- **量化器**

$$e_n' = \begin{cases} +\zeta, & e_n > 0, \quad \zeta\text{是一个正常数} \\ -\zeta, & else, \quad e_n'\text{用 1 位编码} \end{cases}$$



- **预测器**

$$\hat{f}_n = \alpha f_{n-1}'$$

α一般是一个小于1的预测系数。

例析：DPCM

例题3： 已知：$\alpha = 1$，$\zeta = 6.5$，两个像素分别为 $f_0 = 14$、$f_1 = 15$，则：

$$n = 0,\quad \hat{f}_0 = f_0 = 14$$

$$n = 1,\quad \hat{f}_1 = \alpha f_0 = 1 \times 14$$ → （预测结果）

编码：$e_1 = 15 - 14 = 1$ → （预测误差）

$$e_1' = +6.5\left(\because e_1 > 0\right)$$ → （预测误差）

解码：$f_1' = e_1' + \hat{f}_1 = 6.5 + 14 = 20.5$ → （重构结果）

$$f_1 - f_1' = \left(15 - 20.5\right) = -5.5$$ → （重构误差）

例析：DPCM

输入		编码				解码		误差
n	f	^f	e	′e	′f	^f	′f	f-′f
0	14	-	-	-	14.0	-	14.0	0.0
1	15	14.0	1.0	6.5	20.5	14.0	20.5	-5.5
2	14	20.5	-6.5	-6.5	14.0	20.5	14.0	0.0
3	15	14.0	1.0	6.5	20.5	14.0	20.5	-5.5
.	.	.	.	.	.	.	.	
14	29	20.5	8.5	6.5	27.0	20.5	27.0	2.0
15	37	27.0	10.0	6.5	33.5	27.0	33.5	3.5
16	47	33.5	13.5	6.5	40.0	33.5	40.0	7.0
17	62	40.0	22.0	6.5	46.5	40.0	46.5	15.5

四、预测编码

3. 预测器的一般模型

$$\hat{x}_n = f(x_{n-m}, x_{n-m+1}, \cdots, x_{n-1})$$

可以是固定的，也可以是自适应的；可以是线性的，也可以是非线性的。

预测器设计得越好，对输入的数据压缩就越多。

预测器模型

- **一维线性预测**

若$\hat{x}_n$是$x_{n-m}, x_{n-m+1}, \cdots, x_{n-1}$的线性组合，则称为线性预测，

$$\hat{x}(n) = \sum_{k=1}^{m} a_k x(n-k)$$

$$e(n) = x(n) - \hat{x}(n) = x(n) - \sum_{k=1}^{m} a_k x(n-k)$$

其中，a_k预测系数，m预测的阶数。

预测器模型

- 最佳线性预测

选a_k，使$\sigma^2_{d(n)}$最小。设$x(n)$是广义平稳的，且$e(n)$均值为0，则

$$\sigma^2_{e(n)} = E\left\{e^2(n)\right\} = E\left\{\left[x(n) - \sum_{k=1}^{m} a_k x(n-k)\right]^2\right\}$$

令$\dfrac{\partial \sigma^2_{e(n)}}{\partial a_i} = 0$，可得：

$$R(i) - \sum_{k=1}^{m} a_k R(k-i) = 0 \qquad 1 \le i \le m$$

预测器模型

假设$x(n)$是各态遍历的，且训练样本数N相当大，则$x(n)$的自相关函数

$$R(-k)=R(k)\approx\frac{1}{N}\sum_{n=1}^{N}x(n)x(n-k)$$

上式写成矩阵形式：

$$\begin{bmatrix} R(0) & R(1) & \cdots & R(m-1) \\ R(1) & R(0) & \cdots & R(m-2) \\ \vdots & \vdots & \ddots & \vdots \\ R(m-1) & R(m-2) & \cdots & R(0) \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_m \end{bmatrix}=\begin{bmatrix} R(1) \\ R(2) \\ \vdots \\ R(m) \end{bmatrix}$$

预测器模型

- 自适应线性预测

若$x(n)$是非平稳的，或是分段近似广义平稳，则可采用边送数据边测量与估计$x(n)$的自相关函数，求出相应的最佳预测系数。

随之，相应的最佳预测系数随着$x(n)$的统计特性的变化而变化，这就是自适应线性预测。

预测器模型

- 二维线性预测

原始图像用$f(m,n)$表示：

$$\hat{f}(m,n)=\sum_{(k,l)\in Z}\sum a_{k,l}f(m-k,n-l)$$

预测的差值定义为：

$$e(m,n)=f(m,n)-\hat{f}(m,n)$$

这里Z为对像素$f(m,n)$进行预测的相关点的集合。

预测器模型

方程的解 $a_1, a_2, \ldots, a_m$ 是一组最佳的预测系数。压缩效果可用方差$\sigma^2_{e(n)}$来衡量，即：

$$\sigma^2_{e(n)} = \sigma^2_{x(n)} - \sum_{k=1}^{m} a_k R(k)$$

原始序列相关性越强，$R(k)$越大，$\sigma^2_{e(n)}$越小，压缩效果越显著；原始序列互不相关，即$R(k)=0$，$k\neq0$，则，$\sigma^2_{e(n)}=\sigma^2_{x(n)}$一点也不能压缩。

Next Sect...

内容提要

- ◆编码及信息论概述
- ◆行程编码
- ◆霍夫曼编码
- ◆预测编码
- ◆变换编码
- ◆压缩标准简介*

五、变换编码

1. 变换编码的基本思想

- 用一个可逆的、线性的变换(如FFT、DCT)，把图像映射到变换系数集合;
- 对该系数集合进行量化和编码;
- 对于大多数自然图像，重要系数的数量是比较少的。因而可以用量化(或完全抛弃)，且仅以较小的图像失真为代价。

例析：DCT编码

例如：

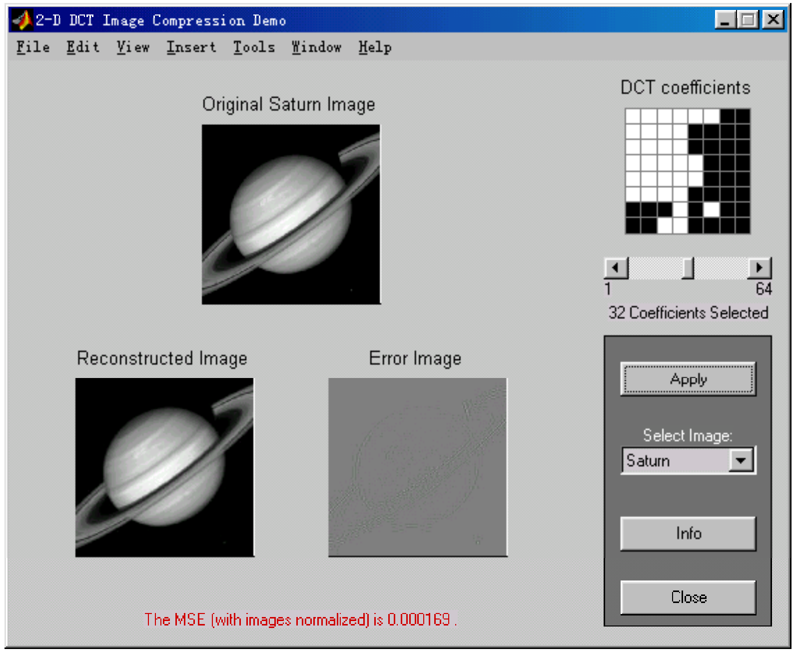
52	55	61	66	70	61	64	73
63	59	66	90	109	85	69	72
62	59	68	113	144	104	66	73
63	58	71	122	154	106	70	69
67	61	68	104	126	88	68	70
79	65	60	70	77	68	58	75
85	71	64	59	55	61	65	83
87	79	69	68	65	76	78	94

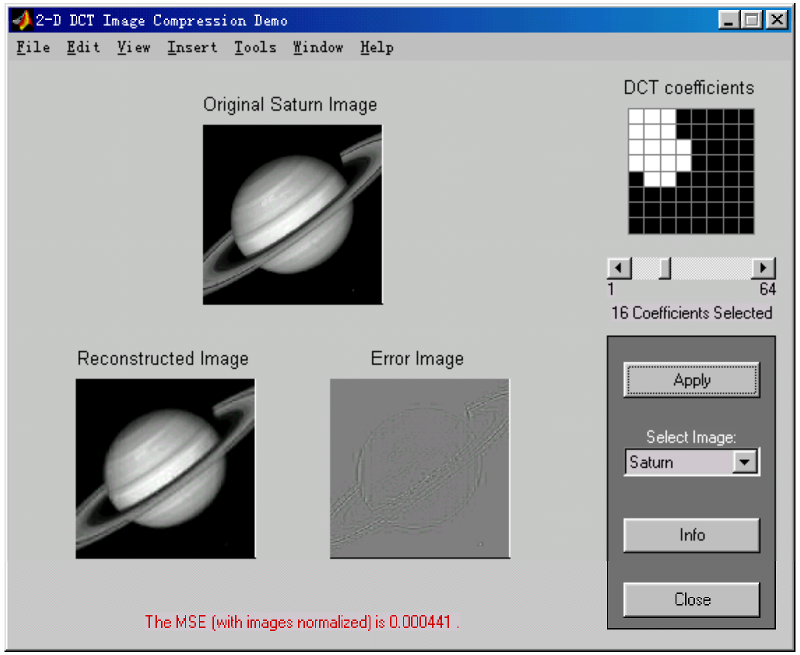
原图像

610	-29	-61	26	55	-19	0	3
7	-20	-61	9	12	-6	-6	7
-46	8	77	-25	-29	11	7	-4
-48	12	35	-14	-9	7	2	2
11	-7	-12	-2	0	2	-4	2
–9	2	4	–3	0	1	2	0
–2	–1	2	1	1	–3	2	–2
–1	0	0	–2	0	0	0	1

DCT变换系数

例析：DCT编码



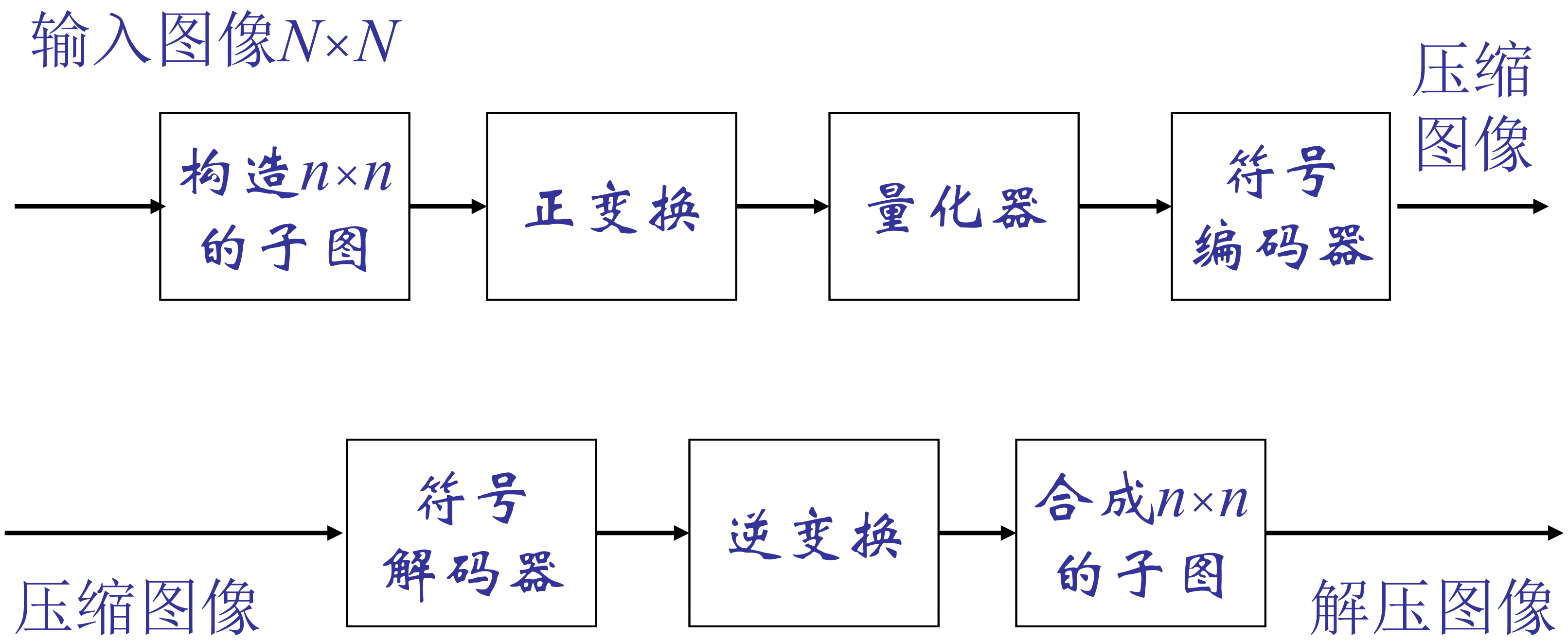


dctdemo.m

DEMO

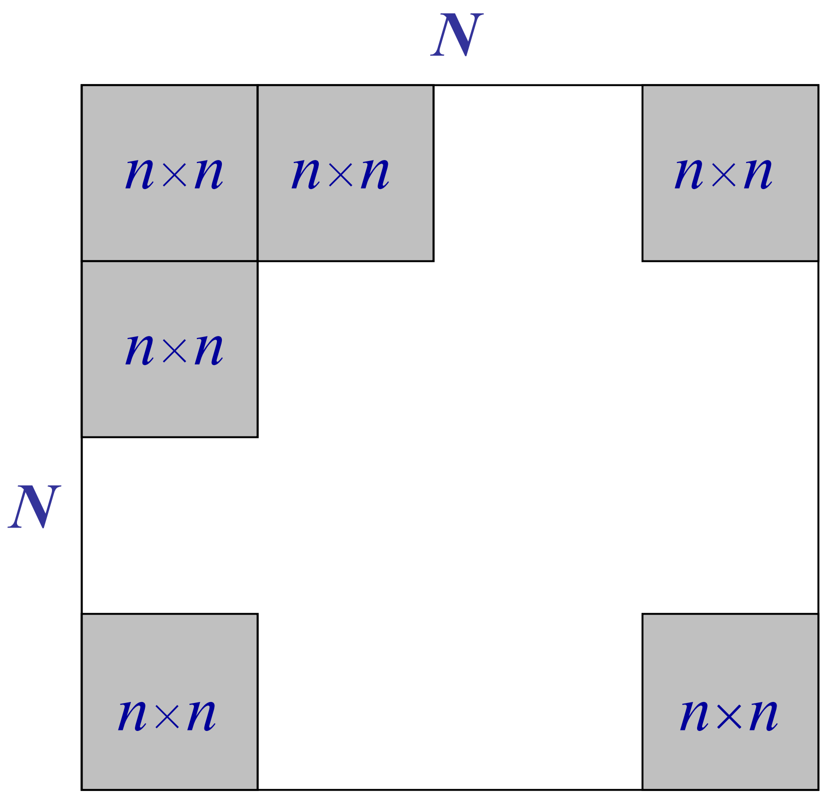
五、变换编码

- 编码/解码流程图



五、变换编码

- 构造$n\times n$的子图



五、变换编码

2.变换编码的基本原理

将FFT逆变换表达式进行改写：

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

F(u,v) 记为：T(u,v)

exp[j2π(ux+vy)/N] 记为：H(x,y,u,v)

$$f(x,y)=\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}T(u,v)H(x,y,u,v)$$

变换编码，即要用等式的右部近似原图像。

五、变换编码——基本理论

进一步改写：

$$F=\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}T(u,v)H_{uv}$$

其中：

(1) F是一个包含了$f(x,y)$的象素的n×n的矩阵；

(2) H_{uv}的值只依赖坐标变量x,y,u,v，与T(u,v)和$f(x,y)$ 的值无关，被称为基图像。

五、变换编码——基本理论

- 基图像H_{uv}

$$H_{uv}=\begin{bmatrix} h(0,0,u,v) & h(0,1,u,v) & \cdots & h(0,n-1,u,v) \\ h(1,0,u,v) & h(1,1,u,v) & \cdots & h(0,n-1,u,v) \\ \cdots & \cdots & \cdots & \cdots \\ h(n-1,0,u,v) & h(n-1,1,u,v) & \cdots & h(n-1,n-1,u,v) \end{bmatrix}$$

H_{uv}可以在变换前一次生成，对每一个n×n的子图变换都可以使用。

五、变换编码——基本理论

- 变换系数截取模板

通过定义变换系数截取模板函数，消去冗余。

$$m(u,v)=\begin{cases}0, & \text{如果}\,T(u,v)\text{满足特定的截断条件}\\ 1, & \text{otherwise}\end{cases}$$

$$\begin{matrix}1&1&1&1&0&0&0&0\\1&1&1&1&0&0&0&0\\1&1&1&0&0&0&0&0\\1&1&0&0&0&0&0&0\\1&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\\0&0&0&0&0&0&0&0\end{matrix}$$

对于：$F=\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}T(u,v)H_{uv}$

近似：$\hat{F}=\sum_{u=0}^{N-1}\sum_{v=0}^{N-1}T(u,v)m(u,v)H_{uv}$

五、变换编码——基本理论

- **误差评估**

$$e_{ms} = E\left\{\left\| F - \hat{F} \right\|^2\right\}$$

$$= E\left\{\left\| \sum_{u=0}^{n-1}\sum_{v=0}^{n-1} T(u,v)H_{uv} - \sum_{u=0}^{n-1}\sum_{v=0}^{n-1} T(u,v)m(u,v)H_{uv} \right\|^2\right\}$$

$$= E\left\{\left\| \sum_{u=0}^{n-1}\sum_{v=0}^{n-1} T(u,v)H_{uv}\left[1-m(u,v)\right] \right\|^2\right\} = \sum_{u=0}^{n-1}\sum_{v=0}^{n-1} \sigma^2_{T(u,v)}\left[1-m(u,v)\right]$$

其中，||F –^F||是矩阵范数，$\sigma_{2T(u,v)}$是变换在(u,v)位置上的系数方差。最后的简化是基图像的规范正交，并假设F的像素是通过一个具有0均值和已知协方差的随机处理产生的。

五、变换编码——基本理论

- **小结**

(1) 总的近似均方误差是丢弃的变换系数的方差之和（也即对于m(u,v) ＝0的系数方差之和）。

(2) 能把大多数信息封装到最少的系数里去的变换，可得到最好的子图像的近似，同时重构误差也最小。

(3) 在等式成立的假设下，一个N×N的图像的$(N/n)^2$个子图像的均方误差是相同的。因此，N×N图像的均方误差（平均误差的测量）等于一个子图像的均方误差。

五、变换编码

3. 几个关键问题

- 变换的选择
- 变换的评价
- 子图尺寸选择
- 压缩位分配（编码）

五、变换编码——基本理论

- 变换的选择
  - Karhunen-Loeve变换（KLT）
  - 离散傅立叶变换（DFT）
  - 离散余弦变换（DCT）
  - Walsh-Hadamard变换（WHT）
  - 离散小波变换（DWT）

五、变换编码——几个关键问题

- **变换的评价**

  按信息封装能力排序：

  KLT，DCT，DFT，WHT，HaarT

- KLT的基图像是数据依赖的，每次都要重新计算Huv，因而很少使用。
- DFT的块效应严重。
- DCT最常用，现已被国际标准采纳，作成芯片。其优点有：a) 基本没有块效应；b) 信息封装能力强，把最多的信息封装在最少的系数中。

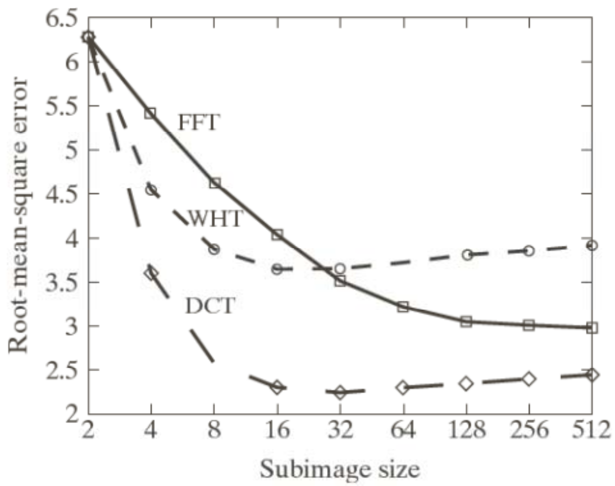
五、变换编码——几个关键问题

- 子图尺寸选择

选择原则：

(1) 如果n是子图的维数，**n应是2的整数次方**，便于降低计算复杂度；

(2) n一般选为8×8或16×16，具体可由试验得到。



一般而言，随着n的增加，块效应相应减少。

五、变换编码——几个关键问题

- **压缩位分配（编码）**

定义：截取、量化、系数编码统称为位分配。
主要解决m(u,v)的设计、编码问题。

截取和量化一般有两种方法：
✓子带编码（固定模板）；
✓阈值编码（自适应编码）。

五、变换编码——几个关键问题

1) 子带编码

基本思想：所有子图像使用相同的编码模板。

1	1	1	1	1	0	0	0
1	1	1	1	0	0	0	0
1	1	1	0	0	0	0	0
1	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

可消去87.5%的系数的模板

五、变换编码——几个关键问题

(a) 大部分的信息包含在方差较大的变换系数中。

- 每一个DCT变换系数被认为是一个随机变量；
- 该变量的分布可以在所有变换子图像的集合上进行计算。

(b) 在$(N/n)^2$个子图找出取方差较大的m个系数的位置。

- 同时确定系数的坐标u和v；
- 对所有子图像，这m个系数的T(u,v)值是保留的，其他的T值被抛弃；
- m是一个可选常数。

五、变换编码——几个关键问题

- **算法实现**
  - **计算模板**：方差较大的地方置1，其它地方置0;
  - **量化系数**：如最优Lloyd-Max量化器等;
  - **系数编码**：有以下2种二进制位分配方法，即

① 系数被赋予相同数量的二进制位；

② 系数之间固定地分配一定的二进制位。

8	7	6	4	3	2	1	0
7	6	5	4	3	2	1	0
6	5	4	3	3	1	1	0
4	4	3	3	2	1	0	0
3	3	3	2	1	1	0	0
2	2	1	1	1	0	0	0
1	1	1	0	0	0	0	0
0	0	0	0	0	0	0	0

五、变换编码——几个关键问题

2) 阈值编码(自适应编码)

基本思想：没有一个取舍系数的固定模板。不同的子图保留不同的系数，即通过一个阈值T，来决定一个系数的去留。

$$\text{if}\quad a(\text{系数}) > T(\text{阈值})\quad m(u,v)=1$$
$$\text{else}\quad m(u,v)=0$$

由于其简单性，阈值编码是实际应用中经常使用的编码方法。

五、变换编码——几个关键问题

理论依据

- ✓ 取值较大的变换系数，在重构子图的质量中起的作用也最重要；
- ✓ 较大系数的分布随子图的不同而不同。

1	1	0	1	0	0	0	0
1	1	1	1	0	0	0	0
1	1	0	0	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

五、变换编码——几个关键问题

- 算法实现

① 阈值选取，一般按以下3种途径：

a. 所有子图使用相同全局阈值。

压缩率的大小由大于全局阈值的系数个数所决定。

b. 每个子图使用不同的阈值。

每个子图保留的系数的个数事先确定，即总保留N个最大的，称为***N*-最大化编码**。对于每个子图同样多的系数被丢弃。因此，每个子图的压缩率是相同的，并且是预先知道的。

五、变换编码——几个关键问题

c. 阈值作为子图系数位置的函数

所有子图使用同一个全局阈值模板，但阈值的选取，与系数的位置相关，阈值模板给出了不同位置上系数的相应阈值。

例如：

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

五、变换编码——几个关键问题

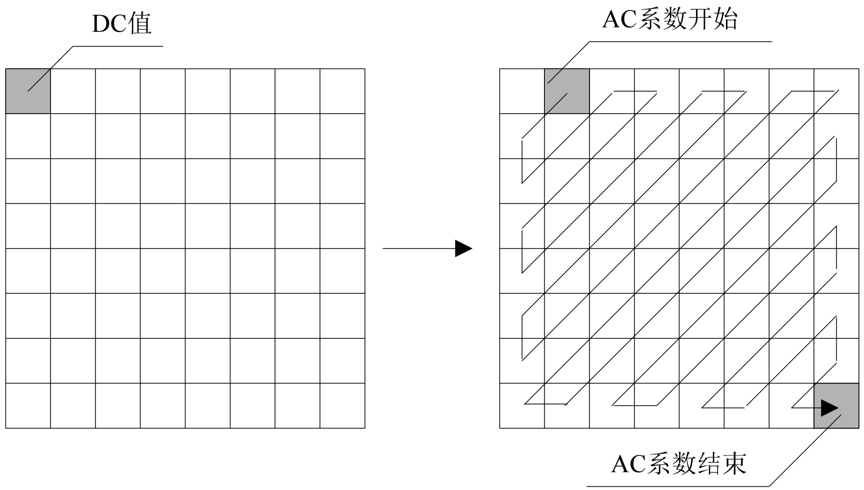
② 系数编码

将系数按45度对角顺序展开成序列，得到有一个有长串为零的序列。

例如： -19 -20 5 21 6 0 0 0 0 0 0 0 0 0 0

用RLE编码对上述序列编码。

0	1	5	6	14	15	27	28
2	4	7	13	16	26	29	42
3	8	12	17	25	30	41	43
9	11	18	24	31	40	44	53
10	19	23	32	39	45	52	54
20	22	33	38	46	51	55	60
21	34	37	47	50	56	59	61
35	36	48	49	57	58	62	63



zig-zag排序

Next Sect...

内容提要

- ◆编码及信息论概述
- ◆行程编码
- ◆霍夫曼编码
- ◆预测编码
- ◆变换编码
- ◆压缩标准简介*

六、压缩标准简介

- 图像压缩标准

静止帧黑白、彩色压缩(JPEG)

连续帧单色、彩色压缩(MPEG)

JPEG —— Joint Photographic Experts Group

MPEG —— Moving Picture Experts Group

（1） JPEG压缩标准

有三种压缩系统：

(1)基线编码系统：面向大多数有损压缩的应用，采用DCT变换压缩。

(2) 扩展编码系统：面向递进式应用，从低分辨率到高分辨率逐步递进传递的应用。

(3)独立编码系统：面向无损压缩的应用，采用无损预测压缩，符号编码采用哈夫曼或算术编码。

一个产品或系统必须包括对基线系统的支持。

JPEG压缩标准

- JPEG压缩流程

输入图像 → 构造8x8的子图 → 颜色空间转换 → 零偏置转换 → DCT正变换 → 量化器 → 符号编码器 → 压缩图像

压缩图像 → 符号解码器 → DCT逆变换 → 颜色空间转换 → 零偏置转换 → 合成8x8的子图 → 解压图像

MPEG1/2/4/7标准

MPEG1/2/4/7标准由ISO/IEC制定的。

国际标准化组织 (International Organization for Standardization，ISO) 。

国际电工委员会 (International Electrotechnical Commission, IEC)，正式成立于1906 年，属于非政府性国际组织，是世界上成立最早的专门国际标准化机构。

MPEG1/2/4/7标准

- 简介
- MPEG的第一个成果MPEG-1于1992年推出，是VCD的基础。由于有限的352×288像素分辨率，MPEG-1只适用于家庭环境，获得的视频质量及数据率相当低。
- 1995推出MPEG-2。720×576的像素以及更高的分辨率大大提高了视频质量。
- 1999年12月发布了MPEG-4。
- MPEG-7为多媒体内容描述接口标准。

从MPEG组织成立至今，其任务和方向都发生了很多变化。MPEG-1和MPEG-2已经是成熟的编码标准，现在的热点主要集中在MPEG-4 和 MPEG-7上。

MPEG1/2/4/7

- MPEG1

应用范围：

视频CD_ROM存储、视频消费。

编码技术

- DCT变换
- 前向、双向运动补偿预测
- Zig-zag排序
- 霍夫曼编码、算术编码
- 每15帧至少要有一个I帧

MPEG1/2/4/7

- MPEG2

应用范围：数字电视、高质量视频、有线电视、视频编辑、视频存储。

编码技术
- DCT变换
- 前向、双向运动补偿预测
- zig-zag排序
- 霍夫曼编码、算术编码
- 每15帧至少要有一个I帧

MPEG1/2/4/7

- MPEG4

应用范围：互联网、交互视频、移动通信。

编码技术

- DCT变换、小波变换
- 前向、双向运动补偿预测
- zig-zag排序
- 脸部动画、背景编码
- 霍夫曼编码、算术编码
- 每15帧至少要有一个I帧

MPEG1/2/4/7

- MPEG7

1998年10月被提出，它的正式名称是“多媒体内容描述借口”。其用途非常广泛：即可用于存储(在线或离线)，也可应用于流式结构(如广播、将模型假如Internet)。还可用于实时或非实时环境下。

MPEG-7的应用领域

- 音视数据库的存储和检索；
- 广播媒体的选择（广播、电视节目）；
- 因特网上的个性化新闻服务；
- 智能多媒体；
- 教育领域的应用；
- 远程购物；
- 社会和文化服务；
- 调查服务
- 遥感；
- 监视；
- 生物医学应用；
- 建筑、不动产及内部设计等。

H. 261/263/264

- H.261/263

H.261/263标准是由CCITT(国际电话与电报咨询委员会)制定的。CCITT现在被称为ITU-T(国际标准化组织电讯标准化分部)，是世界上主要的制定和推广电讯设备和系统标准的国际组织.它位于瑞士的geneva。

H. 261/263/264

- H.261

应用范围：ISDN视频会议。

编码技术
- DCT变换
- 前向运动补偿预测
- zig-zag排序
- 脸部动画、背景编码
- 霍夫曼编码

运动补偿

H. 261/263/264

- H.263

应用范围：可视电话。

编码技术
- DCT变换
- 双向运动补偿预测
- zig-zag排序
- 脸部动画、背景编码
- 霍夫曼编码

H. 261/263/264

- H.264

H.264是ISO/IEC MPEG(运动图像专家组)与ITU-T(视频编码专家组)组成的JVT(联合视频组)提出的新视频编码标准，并于2003年5月确定为国际标准。

同时，还作为MPEG-4的PART10(称为为MPEG4 AVC)。

H. 261/263/264

- H.264

H. 264视频编解码标准的压缩效率可以达到以往标准(如H.263 或MPEG-4) 的1.5～2 倍， 与H. 263标准相比, H.264 采用了许多新技术， 如对4×4 点残差数据块进行整数变换L采用4×4 点整数变换, 计算复杂度低, 峰值信噪比(PSNR) 却只降低0. 02dB。

will be over…

内容提要

- 编码及信息论概述
- 行程编码
- 霍夫曼编码
- 预测编码
- 变换编码
- 压缩标准简介*

Open the next …

Thank you !

Optoelectronic Image Processing

Curriculum Group

2017.11.16

本章作业

1、已知符号 A , E , I , O , U , V 其出现的概率分别是0.1, 0.4, 0.06,0.1,0.04,0.3,对其进行霍夫曼编码，给出码字、码字的平均长度和编码效率。

2、考虑如下大小为4 ×8的图像：

21	21	95	95	169	169	243	243
21	21	95	95	169	169	243	243
21	21	95	95	169	169	243	243
21	21	95	95	169	169	243	243

(1) 计算该图像的熵； (2) 用霍夫曼压缩该图像；
(3)计算用霍夫曼编码能达到的压缩率和效率。

本章作业

3、编程练习：读取一幅512 ×512 ×8比特的单色Lena图像，完成以下步骤：

(1) 统计该图像的概率直方图，并画出直方图；

(2) 计算该图像的熵；

(3) 对其进行霍夫曼编码；

(4) 分别计算压缩率和冗余度。

请使用原图(lena512.jpg)

Terms

- Image compression: 图像压缩
- Image coding: 图像编码
- Encoding: 编码
- Decoding: 解码，译码
- Cryptography: 密码学
- Decompression: 解压
- Encoder: 编码器
- Decoder: 解码器
- Redundant: 冗余的
- Irrelevant: 不相干的

Terms

- Compression ratio: 压缩比
- Dictionary-based encoding techniques: 基于字典的编码技术
- Statistical encoding method: 统计编码方法
- Lossless image compression: 无损图像压缩
- Reversible encoding: 可逆编码
- Error-free encoding: 无误差编码
- Information preserving encoding: 信息保持编码
- Encoding model: 编码模型
- Information: 信息

Terms

- Source of messages: 信源， 消息源
- Memoryless source of messages: 无记忆信源
- Memory source of messages: 有记忆信源
- Entropy: 熵
- Lossy image compression: 有损图像压缩
- Fidelity: 保真度
- Huffman coding:霍夫曼编码
- B code: B码
- Shift code: 移位码
- Run: 行程

Terms

- Run length encoding: 行程编码
- Contour encoding: 轮廓编码
- LZW algorithm: Lemple-Ziv-Welch编码
- Isoprefrence curves: 等值线，等优线
- Freeman’s chain code: Freeman链码
- Mapping: 映射
- Scalar quantization: 标量量化
- Rate distortion theory: 率失真理论
- Date rate: 数据率
- Distortion: 失真度

Terms

- Rate distortion function:率失真函数
- Reconstruction error: 重构误差
- Transform image encoding: 变换图像编码
- Block encoding: 块编码
- Bit allocation: 位分配，比特分配
- Differential encoding:微分编码
- Predictive encoding: 预测编码
- Predictor: 预测器
- Predictive coefficient: 预测系数

Terms

- Differential pulse code modulation (DPCM): 差分脉冲编码调制
- Hybrid encoding: 混合编码
- Subband coding: 子带编码
- Karhunen-Loeve transform: 卡洛变换
- K-L transform: 卡洛变换
- Singular value decomposition transform: 奇异值分解变换
- SVD transform:奇异值分解变换
- Eigenvalue: 特征值

Terms

- Eigenvector: 特征向量
- Hartley transform: 哈特雷变换
- Haar transform: 哈尔变换
- Markov process: 马尔科夫过程
- Covariance matrix: 协方差矩阵
- Hadamard transform: 哈达玛变换
- Walsh transform: 沃尔什变换
- Slant transform: 斜变换