第8章 图像压缩

- 图像压缩是一种减少描绘一幅图像所需数据量的技术和科学， 它是数字图像处理领域最有用、商业上最成功的技术之一
  - 压缩的本质：用尽可能少的数据表达尽可能多的信息
  - 图像压缩必要性实例
    - ✓ 一部两小时的标清电影，分辨率720×480，全彩色（24bit/pixel）
    - ✓ 如果不压缩，存储需要224GB！
- Web网页图像和高分辨率数字摄像机图片需要进行压缩，以节省**存储**空间和减少**传输**时间
- 图像压缩在视频会议、遥感、文本和医学图像处理领域扮演重要角色

第8章 图像压缩

- 图像视频压缩方向与科大有渊源的IEEE Fellow
  - 前五位均担任过 IEEE T-CSVT主编



李卫平（776）



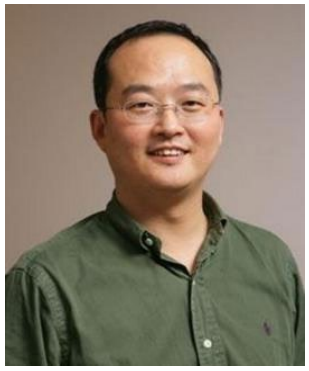
张亚勤（78少）



陈长汶（786）



李世鹏（836）



吴枫



李厚强（876）

第8章 图像压缩

- 8.1 基础知识
  - 编码冗余、空间冗余和时间冗余、不相关的信息量、图像信息的度量、保真度准则、图像压缩模型、图像格式容器和压缩标准
- 8.2 一些基本的压缩方法
  - 霍夫曼编码
  - 算术编码
  - LZW编码
  - 行程编码
  - 基于符号的编码
  - 比特平面编码
  - 块变换编码
  - 预测编码
  - 小波编码
- 8.3 数字图像水印

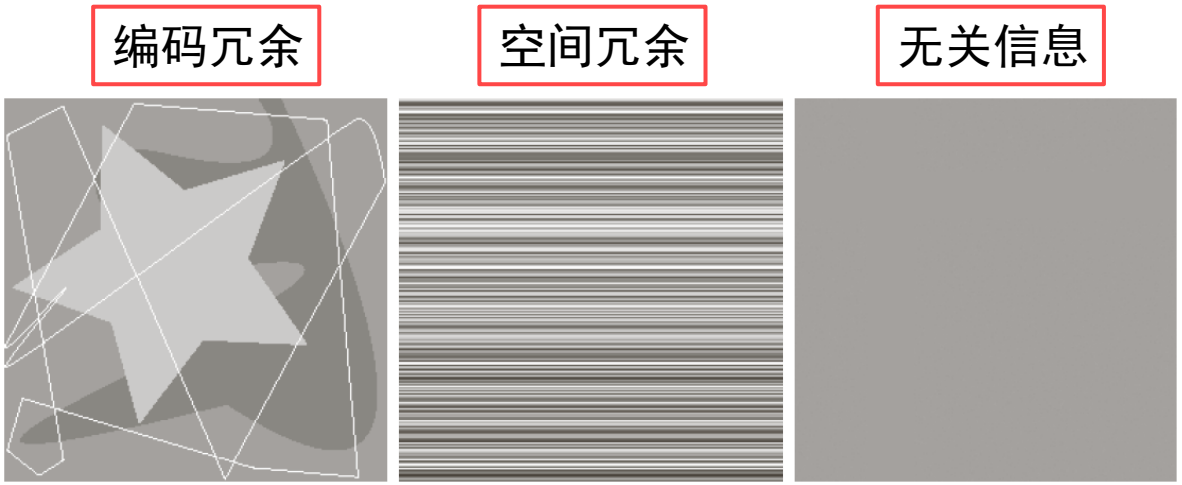
8.1 基础知识

- 相对数据冗余R

$$R = 1 - \frac{1}{C} = \frac{b - b'}{b'}$$

压缩率$C = b/b'$

- 三种主要类型的数据冗余：
  - 编码冗余：统计冗余
  - 空间和时间冗余
  - 不相关的信息：感知冗余



计算机产生的256 × 256 × 8比特图像

编码冗余

- r_k表示一副$M \times N$图像的灰度，r_k出现概率$p_r(r_k)$

$$p_r(r_k) = \frac{n_k}{MN}, k = 0,1,2,\ldots,L-1$$

$$L_{avg} = \sum_{k=0}^{L-1} l(r_k)p_r(r_k)$$

- 变长编码：L_{avg}=1.81 bit
- 定长编码：L_{avg}=8 bit
- C=8/1.81=4.42　R=1-1/C=0.774

r_k	$p_r(r_k)$	Code 1	$l_1(r_k)$	Code 2	$l_2(r_k)$
r_{87} = 87	0.25	01010111	8	01	2
r_{128} = 128	0.47	10000000	8	1	1
r_{186} = 186	0.25	11000100	8	000	3
r_{255} = 255	0.03	11111111	8	001	3
r_k for $k \neq 87, 128, 186, 255$	0	—	8	—	0

TABLE 8.1 Example of variable-length coding.

空间冗余和时间冗余

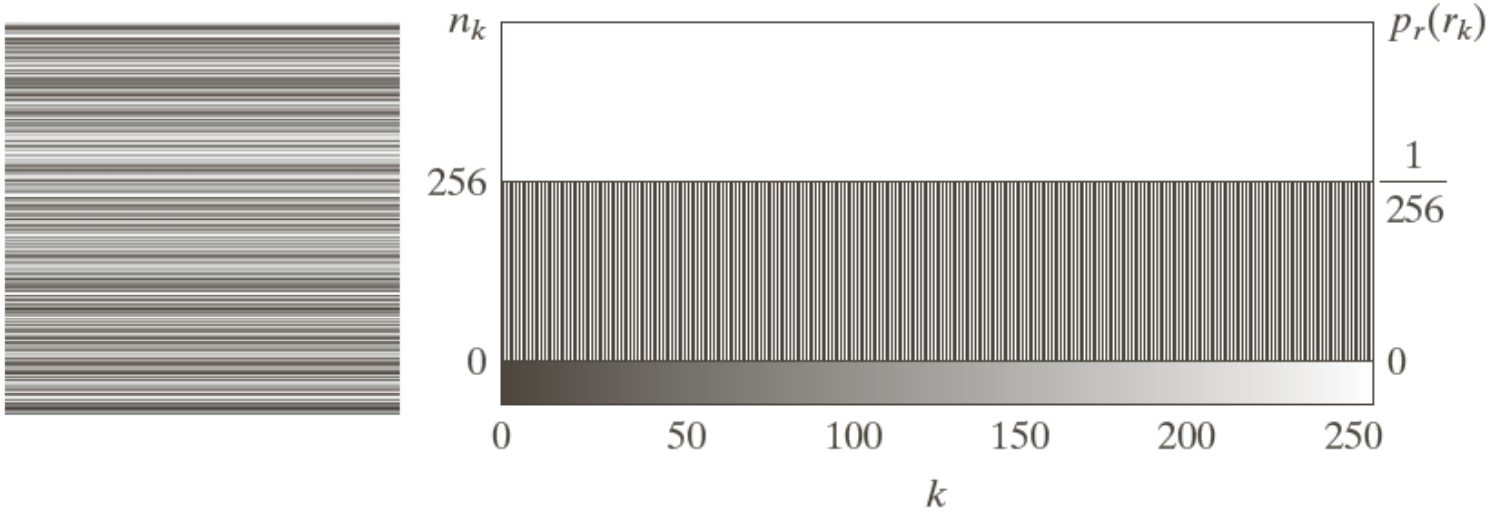
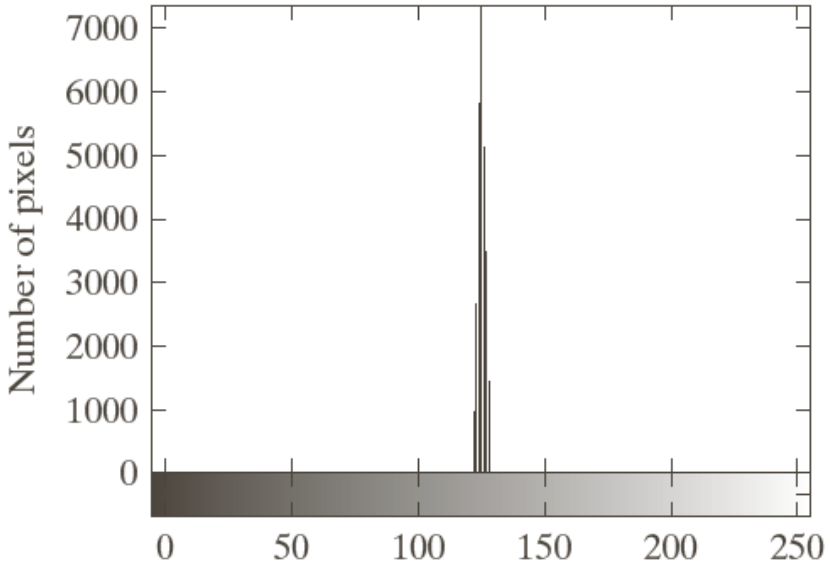


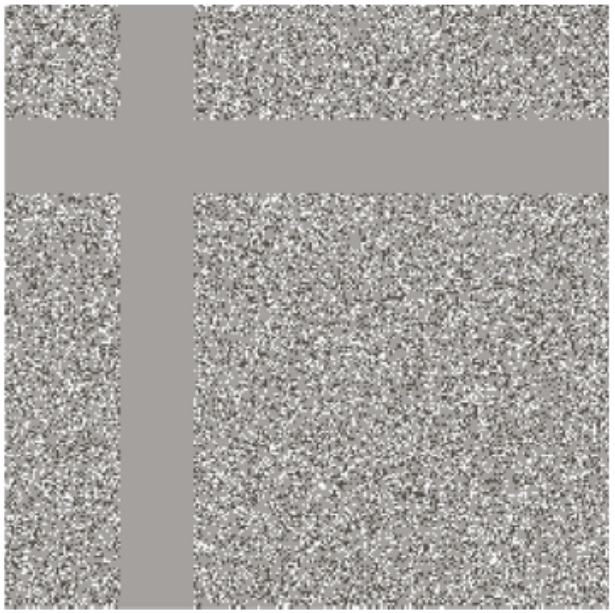
FIGURE 8.2 The intensity histogram of the image in Fig. 8.1(b).

- 空间相关（x和y方向）：多数像素可根据相邻像素灰度进行合理预测
  - 行程编码：指定一个新的灰度开始点，以及该灰度持续像素的数量
- 时间相关：当该图像是视频序列的一部分时

不相关信息

- 数字图像压缩中，被人类视觉系统忽略的信息或与图像与其的应用无关的信息需要被删除





a b

FIGURE 8.3
(a) Histogram of the image in Fig. 8.1(c) and (b) a histogram equalized version of the image.

图像信息的度量

- 概率为$P(E)$的随机事件包含的信息

$$I(E) = -\log P(E)$$

- 信源的平均信息称为该信源的熵

$$H = -\sum_{j=1}^{J} P(a_j) \log P(a_j)$$

- 灰度信源的熵

$$\tilde{H} = -\sum_{k=0}^{L-1} P_r(r_k) \log_2 p_r(r_k)$$

- 香农第一定理（无噪声编码定理）

$$\lim_{n\to\infty} \left[\frac{L_{avg,n}}{n} \right] = H$$

保真度准则：度量压缩质量

- $f(x,y)$是输入图象，其近似图象为$\hat{f}(x,y)$
- $f(x,y)$和$\hat{f}(x,y)$之间误差$e(x,y)$

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

- 平均误差的平方根

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

- 均方信噪比

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

保真度准则

Value	Rating	Description
1	Excellent	An image of extremely high quality, as good as you could desire.
2	Fine	An image of high quality, providing enjoyable viewing. Interference is not objectionable.
3	Passable	An image of acceptable quality. Interference is not objectionable.
4	Marginal	An image of poor quality; you wish you could improve it. Interference is somewhat objectionable.
5	Inferior	A very poor image, but you could watch it. Objectionable interference is definitely present.
6	Unusable	An image so bad that you could not watch it.

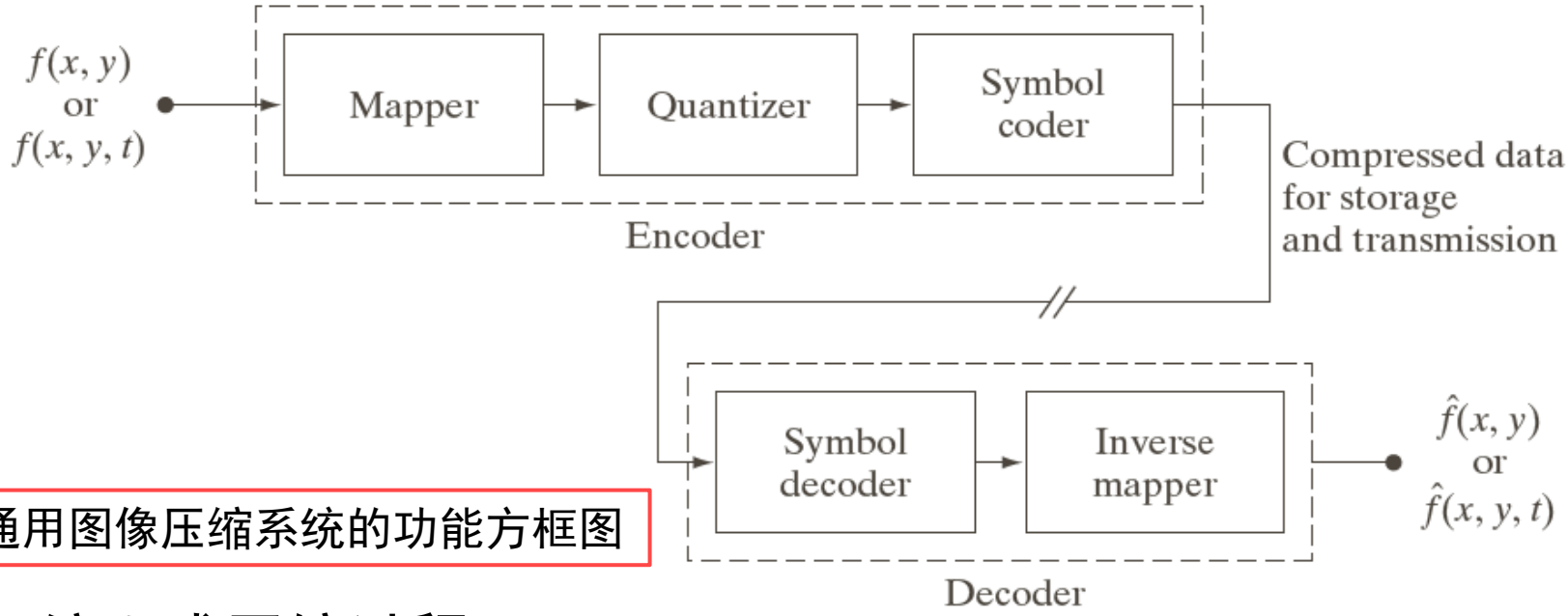
电视配置研究组织的等级尺度

压缩重建的典型图像

人工生成的图像



图像压缩模型

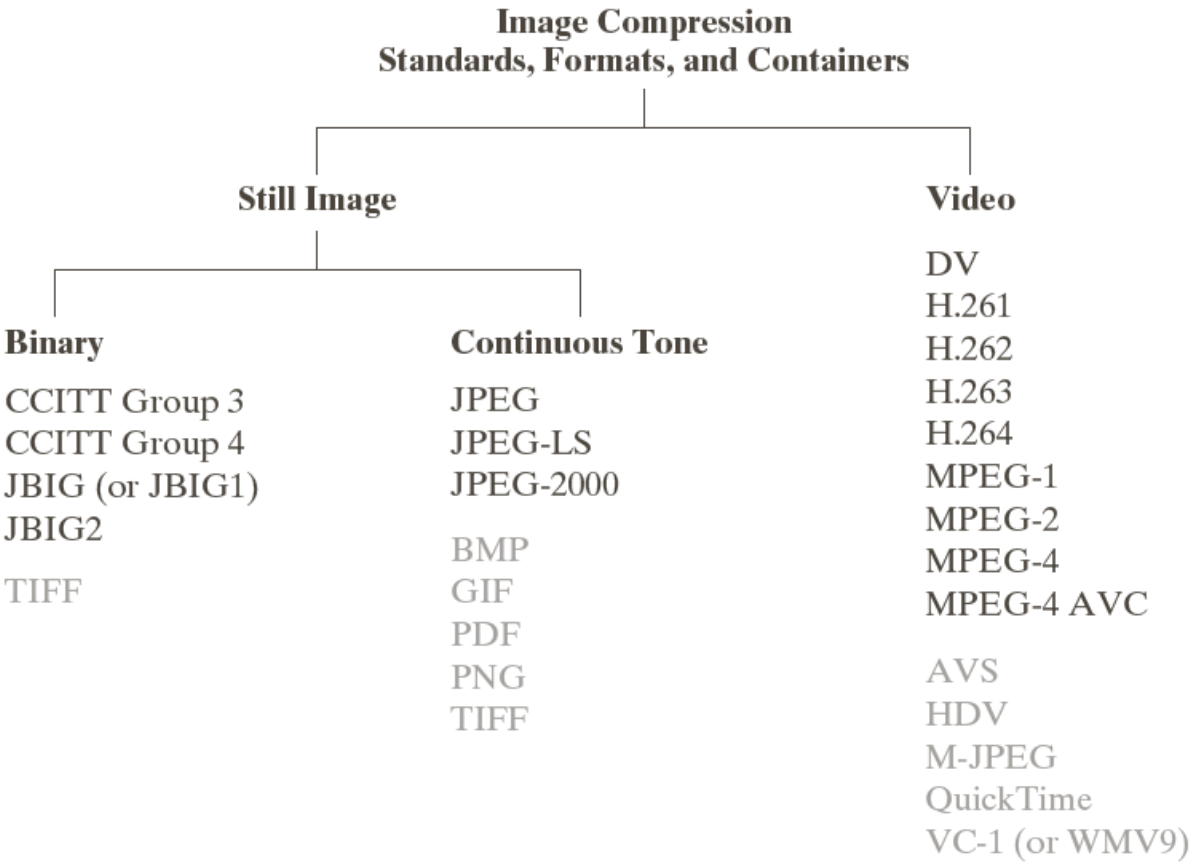


一个通用图像压缩系统的功能方框图

- 编码或压缩过程
  - 映射器：降低空间和时间冗余；可逆
  - 量化器：根据预设保真度准则，排除压缩表示无关信息；不可逆
  - 符号编码器：生成定长编码或变长编码，最小化编码冗余；可逆
- 解码或解压缩过程
  - 符号解码器和反映射器

图像格式、容器和压缩标准

- 文件格式：相当于高层的信息，定义了数据类型、压缩方法等
- 容器：相当于一个存放压缩以后的数据的东西
- 压缩方法：处在更底层，规定了数据的具体压缩的过程



8.2 一些基本的压缩方法

- 霍夫曼编码
- 算数编码
- LZW编码
- 行程编码
- 基于符号的编码
- 比特平面编码
- 块变换编码
- 预测编码
- 小波编码

霍夫曼编码

霍夫曼编码步骤：

（1）概率排序，缩减和简化信源符号数量

Original source		Source reduction			
Symbol	Probability	1	2	3	4
a_2	0.4	0.4	0.4	0.4	0.6
a_6	0.3	0.3	0.3	0.3	0.4
a_1	0.1	0.1	0.2	0.3	
a_4	0.1	0.1	0.1		
a_3	0.06	0.1			
a_5	0.04				

霍夫曼编码

霍夫曼编码步骤：

（2）对每个简化后的信源进行编码，从最小信源开始，直到遍历原始信源

Original source			Source reduction							
Symbol	Probability	Code	1		2		3		4	
a_2	0.4	1	0.4	1	0.4	1	0.4	1	0.6	0
a_6	0.3	00	0.3	00	0.3	00	0.3	00	0.4	1
a_1	0.1	011	0.1	011	0.2	010	0.3	01		
a_4	0.1	0100	0.1	0100	0.1	011				
a_3	0.06	01010	0.1	0101						
a_5	0.04	01011								

平均长度

$$L_{avg} = 0.4 \times 1 + 0.3 \times 2 + 0.1 \times 3 + 0.1 \times 4 + 0.06 \times 5 + 0.04 \times 5 = 2.2$$

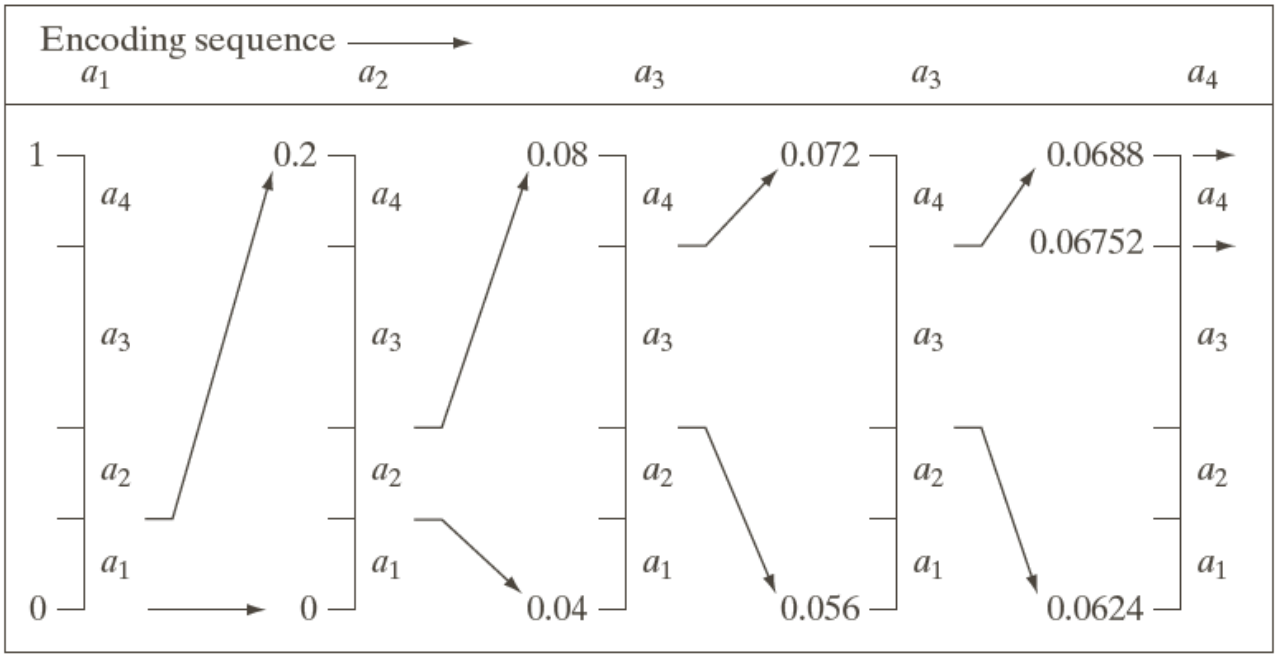
算术编码

- 将一个信源符号序列编码为一个介于0到1之间的实数
- 示例：对来自一个4符号信源的5符号序列$a_1\ a_2\ a_3\ a_3\ a_4$进行编码

Source Symbol	Probability	Initial Subinterval
a_1	0.2	[0.0, 0.2)
a_2	0.2	[0.2, 0.4)
a_3	0.4	[0.4, 0.8)
a_4	0.2	[0.8, 1.0)

信源

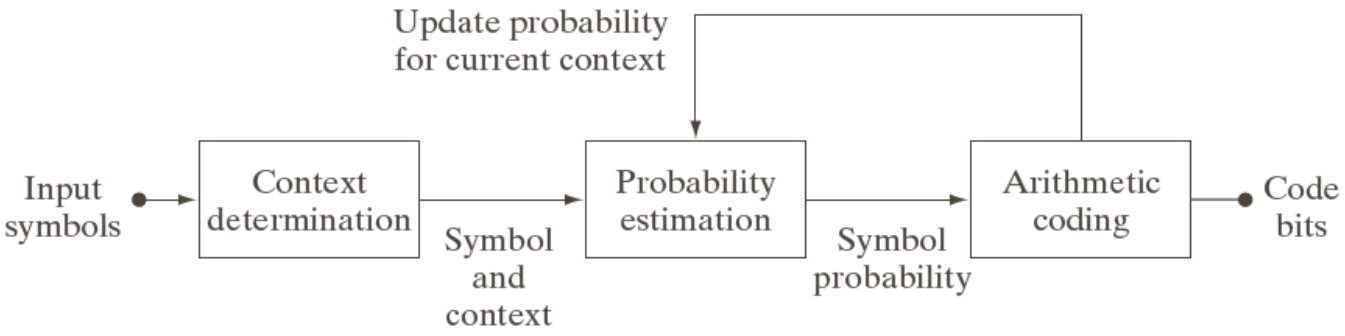
算术编码过程



在该区间的任何数字都可以表示该信息

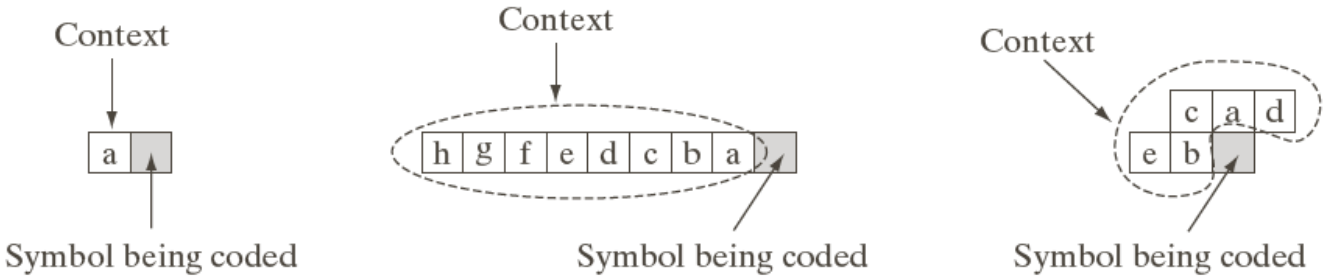
算术编码

- 自适应上下文相关的概率估计



a
b c d

FIGURE 8.13
(a) An adaptive, context-based arithmetic coding approach (often used for binary source symbols). (b)–(d) Three possible context models.



前一符号　　前一组符号　　加上一行扫描符号

(a) 自适应上下文相关的算术编码方法（通常用于二值信源符号）
(b)~(d) 三种可能的上下文模型：

LZW编码

- 无误差压缩图像中空间冗余的方法
- 将定长码字分配各变长信源符号序列
- 编码字典在对数据进行编码的同时创建的

39 39 126 126
39 39 126 126
39 39 126 126
39 39 126 126

Dictionary Location	Entry
0	0
1	1
⋮	⋮
255	255
256	—
⋮	⋮
511	—

Currently Recognized Sequence	Pixel Being Processed	Encoded Output	Dictionary Location (Code Word)	Dictionary Entry
	39			
39	39	39	256	39-39
39	126	39	257	39-126
126	126	126	258	126-126
126	39	126	259	126-39
39	39			
39-39	126	256	260	39-39-126
126	126			
126-126	39	258	261	126-126-39
39	39			
39-39	126			
39-39-126	126	260	262	39-39-126-126
126	39			
126-39	39	259	263	126-39-39
39	126			
39-126	126	257	264	39-126-126
126		126		

LZW编码示例

行程编码

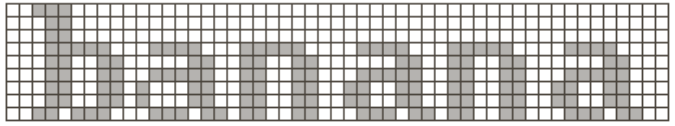
- 处理行（或列）重复灰度的图像
- 行程对指定一个新灰度的开始和具有该灰度的连续像素的数量
- 主要适用于压缩二值图像
- BMP文件格式的RLE是行程编码的一种形式
- 两种古老且应用最广的二值图像压缩标准CCITT Group3和4采用了行程编码

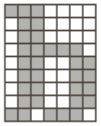
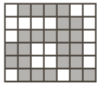
表 8.8　BMP 绝对值编码模式选项。在这种模式中，BMP 对的第一个字节为 0

第二个字节值	条　　件
0	行尾
1	图像尾
2	移到一个新位置
3~255	逐个指定像素

基于符号的编码

- 图像被表示为多幅频繁发生的子图像的集合，称为符号
- 符号字典：符号的集合
- 图像以一个三元组$\{(x_1, y_1, t_1), \{(x_2, y_2, t_2)\}, \ldots\}$的集合来编码，（$x_i$, y_i）规定图像中某个符号的位置，t_i表示该子图在字典中的地址



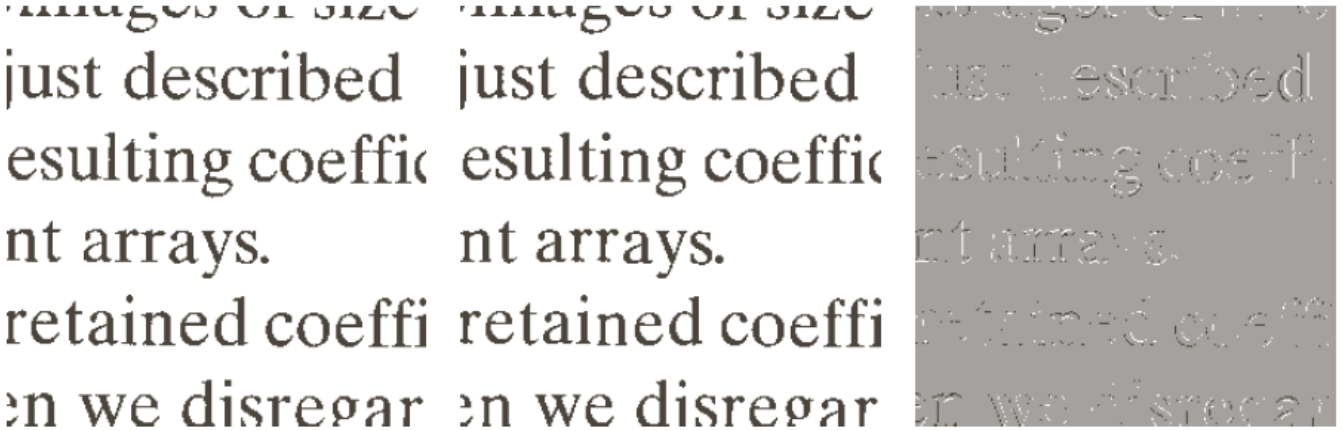
Token	Symbol
0	
1	
2	

Triplet
(0, 2, 0) (3, 10, 1) (3, 18, 2) (3, 26, 1) (3, 34, 2) (3, 42, 1)

(a) 两灰度级文档 (b) 符号字典 (c) 文档中用于定位的三元组

基于符号的编码

- JBIG2是针对二级图像压缩的国际标准
- JBIG2包括无损压缩和感知无损压缩
- **感知无损**指忽略字典位图和图像中相应字符特例之间的差别，但是不影响可读性



(a)无损压缩和重建　(b)感知无损压缩　(c)两种压缩程度的差值

比特平面编码

- 将一幅多级图像分解为一系列二值图像，然后使用二值压缩方法进行编码
- 两种分解方法
  - 基2多项式

$$a_{m-1}2^{m-1}+a_{m-2}2^{m-2}+\ldots+a_1 2^1+a_0 2^0$$

  - m比特格雷码表示图像
    - ✓ 将m比特分解为格雷码 $g_{m-1}\cdots g_2 g_1 g_0$

$$g_i = a_i \oplus a_{i+1}, 0 \le i \le m-2$$

$$g_{m-1} = a_{m-1}$$

  - 特点：连续码字只有一个比特位不同
    - ✓ 例如，127和128的格雷码分别是01000000和11000000

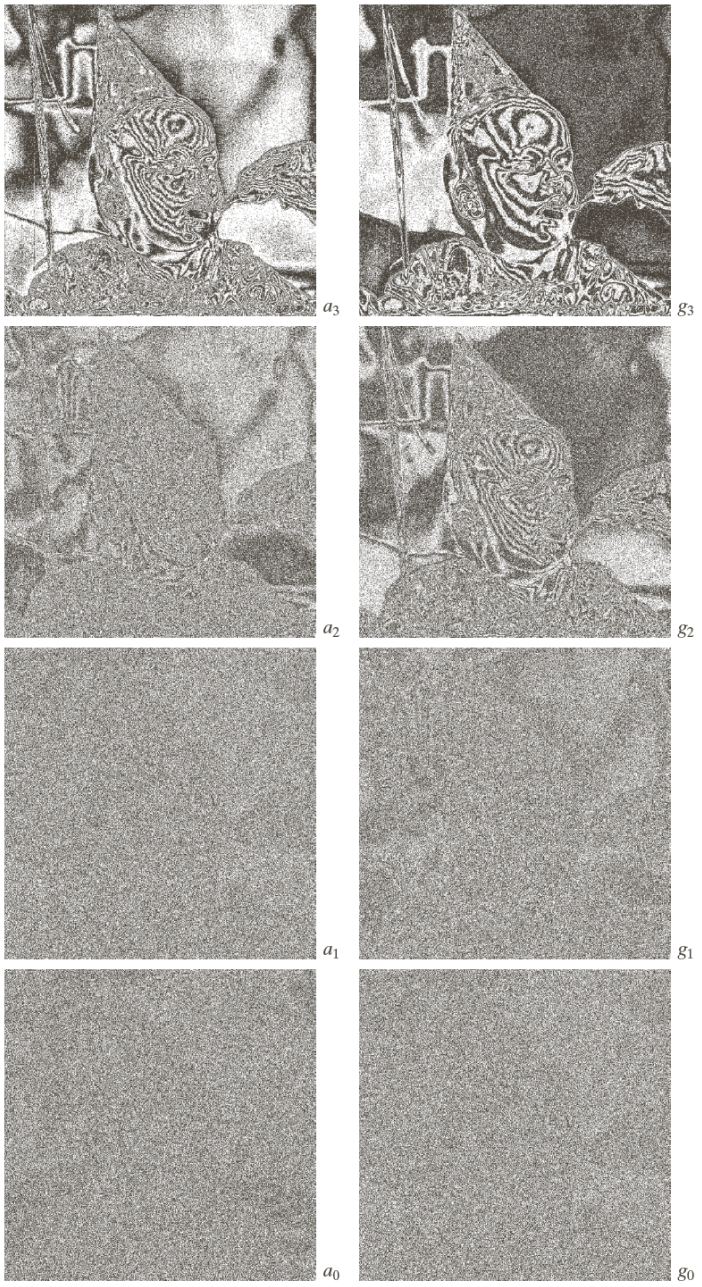
比特平面编码

(a) 一幅256比特单色图像
(b)-(h) 图(a)中图像的4个最高有效二进制编码和格雷编码比特平面



比特平面编码

(a)-(h) 图(a)中图像的4个最低有效二进制编码和格雷编码比特平面

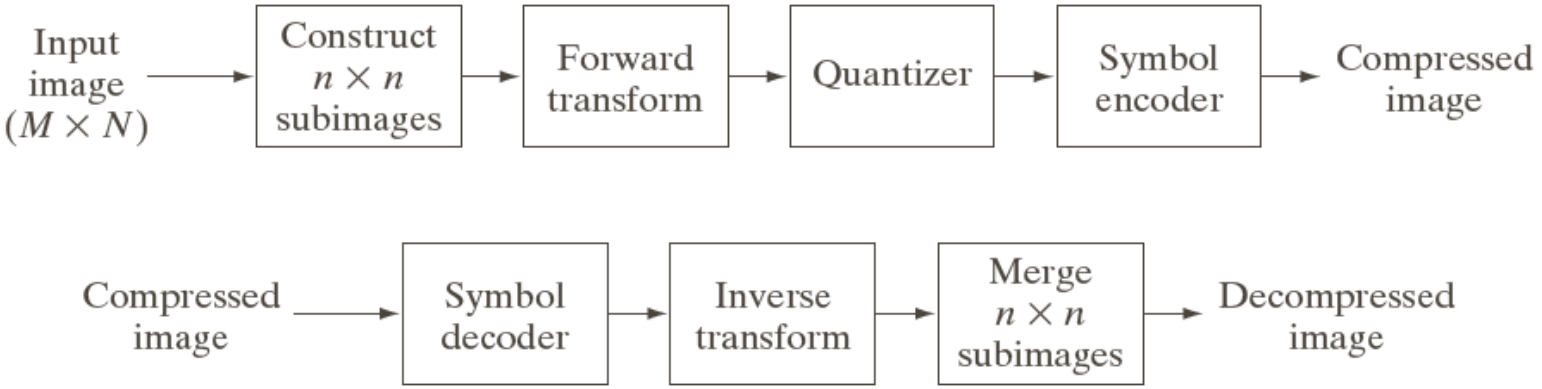


比特平面编码

Coefficient *m*	Binary Code (PDF bits)	Gray Code (PDF bits)	Compression Ratio
7	6,999	6,999	1.00
6	12,791	11,024	1.16
5	40,104	36,914	1.09
4	55,911	47,415	1.18
3	78,915	67,787	1.16
2	101,535	92,630	1.10
1	107,909	105,286	1.03
0	99,753	107,909	0.92

图像的二进制编码和格雷编码比特平面
的JBIG2无损编码结果

块变换编码



块变换编码系统

- 在给定的应用中，特定变换的选择取决于可容忍的重建误差大小和可用的计算资源
- 压缩是在对变换系数的量化期间实现的

正变换 $$T(u,v) = \sum_{x=0}^{n-1}\sum_{y=0}^{n-1} g(x,y)r(x,y,u,v)$$

反变换 $$g(x,y) = \sum_{x=0}^{n-1}\sum_{y=0}^{n-1} T(u,v)s(x,y,u,v)$$

块变换编码

- 傅里叶变换核

$$r(x,y,u,v) = e^{-j2\pi(ux+vy)/n}$$

$$s(x,y,u,v) = \frac{1}{n^2} e^{j2\pi(ux+vy)/n}$$

- Walsh-Hadamard变换（WHT）
  - 实现简单，所有核值均为+1或-1

$$r(x,y,u,v) = s(x,y,u,v) = \frac{1}{n}(-1)^{\sum_{i=0}^{m-1}\lfloor b_i(x)p_i(u)+b_i(y)p_i(v)\rfloor}$$

$$n = 2^m$$

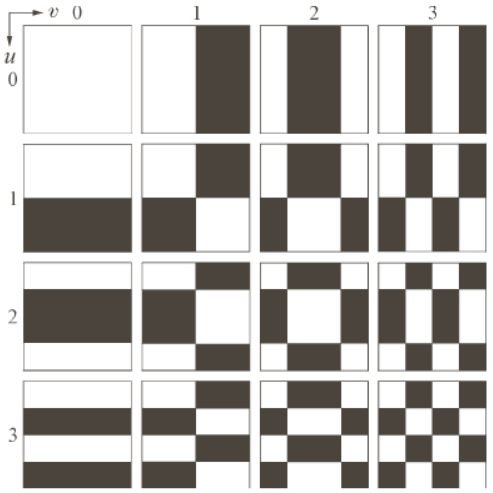
$b_k(z)$ 是z的二进制表示中的第k位

$$p_0(u) = b_{m-1}(u)$$

$$p_1(u) = b_{m-1}(u) + b_{m-2}(u)$$

...

$$p_{m-1}(u) = b_1(u) + b_0(u)$$



n=4时的WHT的基函数

块变换编码

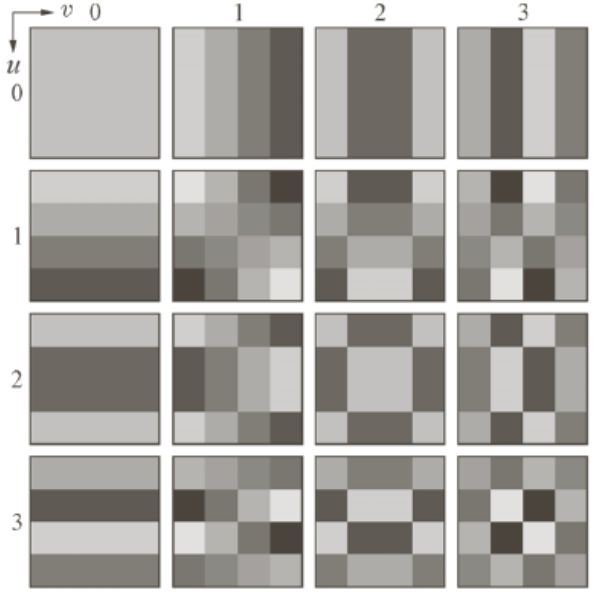
- 离散余弦变换（DCT）

$$r(x,y,u,v)=s(x,y,u,v)=\alpha(u)\alpha(v)\cos\left[\frac{(2x+1)u\pi}{2n}\right]\cos\left[\frac{(2y+1)v\pi}{2n}\right]$$

$$\alpha(u)=\begin{cases}\sqrt{1/n}, u=0\\ \sqrt{2/n}, u=1,2,\ldots,n-1\end{cases}$$

- DCT信息携带能力比DFT和WHT强

- KL变换的信息携带能力更强，但依赖数据



n=4时的DCT的基函数

块变换编码

先分8*8图像块，然后进行变换，对变换后最小的32个系数置零，利用剩下的32个大系数进行重构

重建误差：
傅里叶：2.32；
WHT：1.78；
DCT：1.13

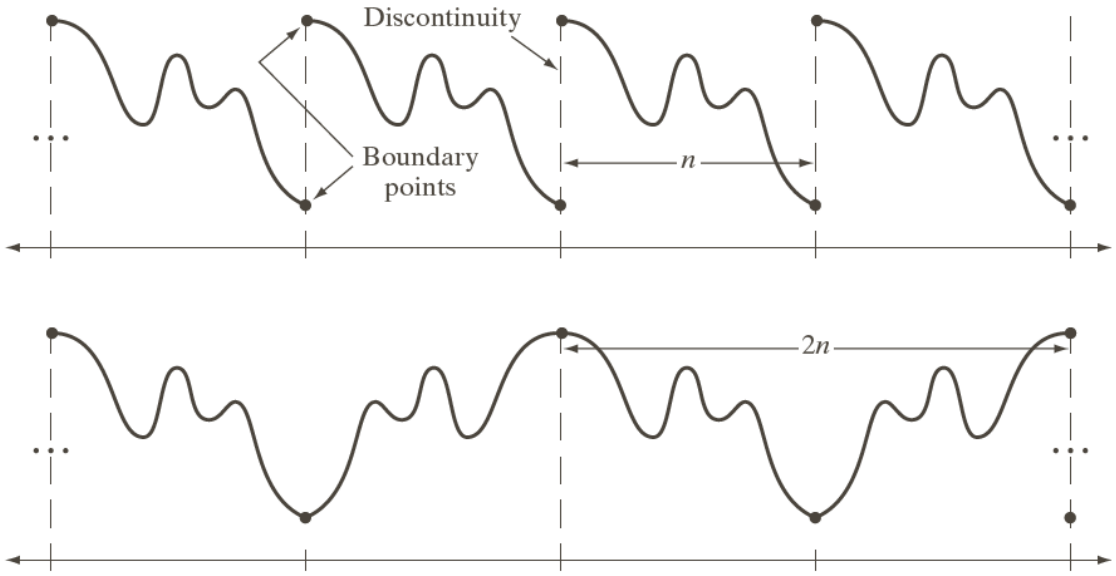


a b c
d e f

FIGURE 8.24 Approximations of Fig. 8.9(a) using the (a) Fourier, (b) Walsh-Hadamard, and (c) cosine transforms, together with the corresponding scaled error images in (d)–(f).

块变换编码

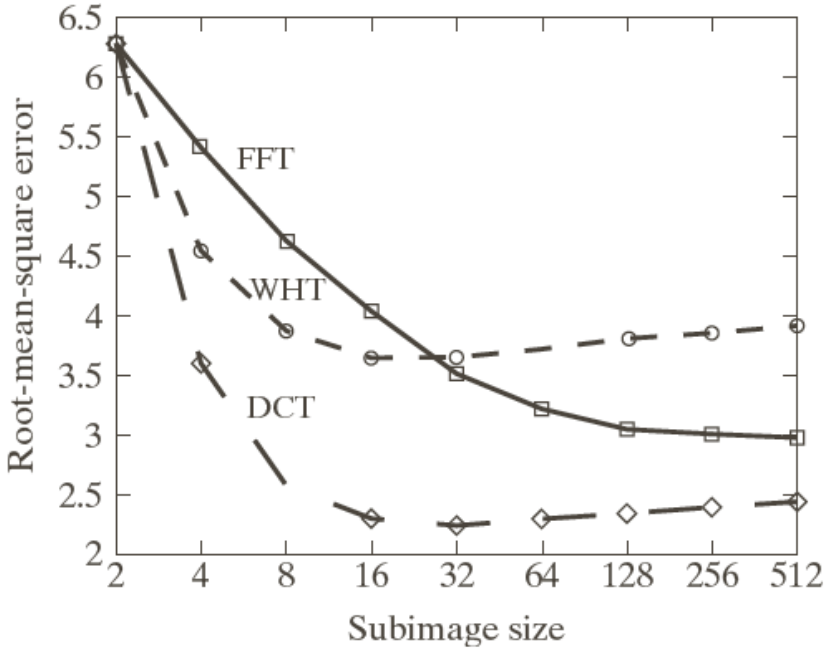
- DCT的优点
  - 用单片集成电路即可实现
  - **可将最多信息装入最少系数中**
  - 可使块效应最小化



(a)一维DFT中隐含的周期性，存在边界不连续，导致块缺陷的快效应
(b)一维DCT中隐含的周期性，无不边界不连续

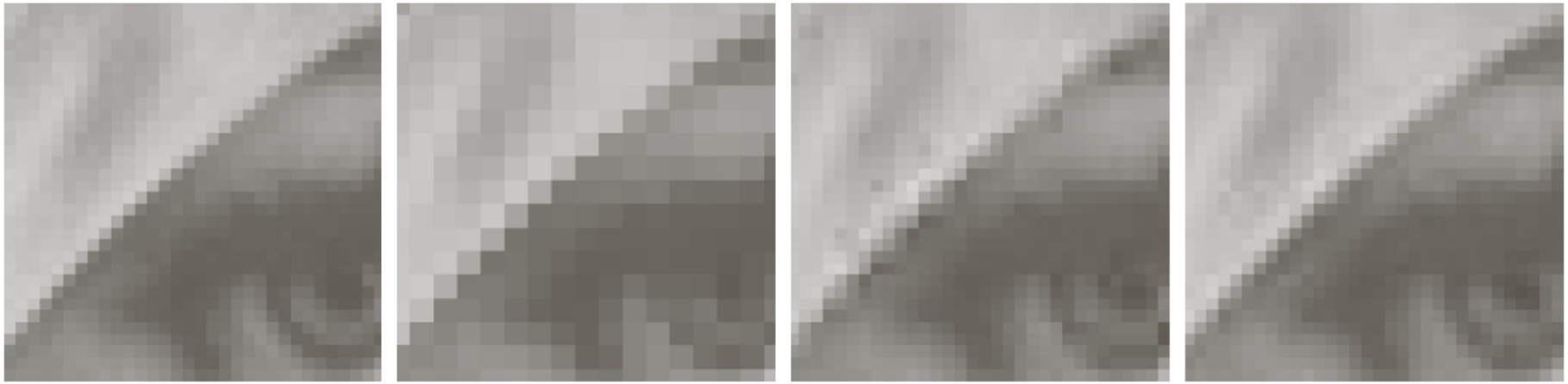
块变换编码

- 子图像尺寸的选择
  - 子图像的尺寸是影响编码误差和计算复杂性的重要因素
  - 一般压缩效率和计算复杂度随子图像尺寸的增加而增加
  - 最常用的子图像尺寸为8 × 8和16 × 16



重建误差与子图像尺寸的关系
（除去75%的变换系数后反变换重建图像）

块变换编码



使用25%DCT系数对原图的近似
(a)原图
(b)使用2 ×2大小的子图的近似
(c)使用4×4大小的子图的近似
(d) 使用8 ×8大小的子图的近似

块变换编码

- 编码方式
  - 区域编码
    - ✓ 保留的系数是根据最大方差进行选择的，方差通过统计子图像计算得到
  - 阈值编码
    - ✓ 保留的系数是根据最大幅度进行选择的
- 比特分配
  - 对变换后的子图像的系数进行截取、量化和编码的过程



a b
c d

FIGURE 8.28 Approximations of Fig. 8.9(a) using 12.5% of the 8×8 DCT coefficients: (a)—(b) threshold coding results; (c)—(d) zonal coding results. The difference images are scaled by 4.

块变换编码

- 区域编码的实现
  - 具有最大方差的变换系数被保留下来
  - 区域取样可看做变换矩阵T(u,v)乘以区域模板的对应元素

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

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

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

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

a b
c d

FIGURE 8.29 A typical (a) zonal mask, (b) zonal bit allocation, (c) threshold mask, and (d) thresholded coefficient ordering sequence. Shading highlights the coefficients that are retained.

块变换编码

- 阈值编码的实现
  - 区域编码对所有子图像采用单个固定模板来实现
  - 阈值编码在不同子图像保留系数的位置上具有自适应性
  - 阈值模板函数的生成方法有三种
    - ✓ 对所有子图像采用单一的全局阈值
      - ➢ 对不同图像的压缩水平不同，取决于超过全局阈值的系数数量
    - ✓ 对每幅子图像采用不同的阈值
      - ➢ 对每幅子图像丢掉相同数量的系数，代码率恒定
    - ✓ 阈值随子图像内每个系数位置的变化而变化

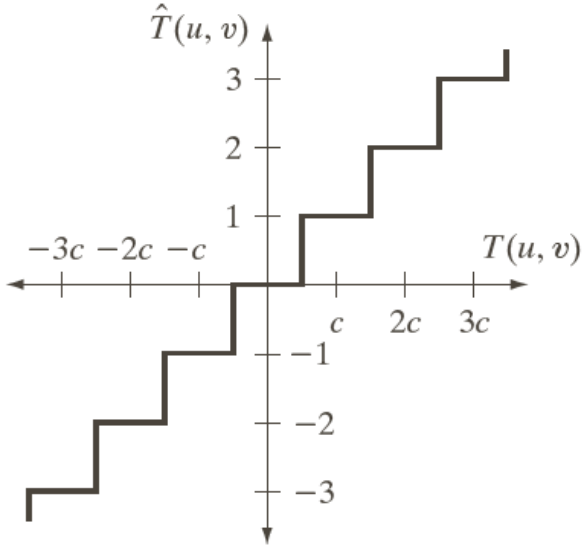
$$\hat{T}(u,v) = \text{round}\left(\frac{T(u,v)}{Z(u,v)}\right) \qquad \dot{T} = \hat{T}(u,v)\,Z(u,v)$$

$$Z = \begin{bmatrix} Z(0,0) & Z(0,1) & \cdots & Z(0,n-1) \\ Z(1,0) & & \cdots & \\ & & & \\ Z(n-1,0) & Z(n-1,1) & \cdots & Z(n-1,n-1) \end{bmatrix}$$

变换标准化阵列

块变换编码

- 阈值编码的实现



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

a b

FIGURE 8.30 (a) A threshold coding quantization curve [see Eq. (8.2-29)]. (b) A typical normalization matrix.

块变换编码

- 阈值编码的实现



FIGURE 8.31 Approximations of Fig. 8.9(a) using the DCT and normalization array of Fig. 8.30(b): (a) **Z**, (b) 2**Z**, (c) 4**Z**, (d) 8**Z**, (e) 16**Z**, and (f) 32**Z**.

压缩比C：12:1; 19:1; 30:1; 49:1; 85:1; 182:1
相应的均方误差：3.83,4.93,6.62,9.35,13.94和22.46

块变换编码

- JPEG标准
  - 广泛使用的连续色调静止帧压缩标准
  - JPEG定义了三种编码系统
    - ✓ 有损的基本编码系统，以DCT为基础
    - ✓ 扩展的编码系统，用于更大压缩、更高精度或渐进式重建
    - ✓ 面向可逆压缩的无损独立编码系统
  - JPEG顺序基线系统
    - ✓ 输入和输出数据精度为8比特，量化后的DCT值为11比特
    - ✓ 压缩步骤：DCT计算、量化、变长编码

块变换编码

- JPEG基本编码与解码
  - 8 ×8子图像

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	63	58	75
85	71	64	59	55	61	65	83
87	79	69	68	65	76	78	94

  - 移位128个灰度级

−76	−73	−67	−62	−58	−67	−64	−55
−65	−69	−62	−38	−19	−43	−59	−56
−66	−69	−60	−15	16	−24	−62	−55
−65	−70	−57	−6	26	−22	−58	−59
−61	−67	−60	−24	−2	−40	−60	−58
−49	−63	−68	−58	−51	−65	−70	−53
−43	−57	−64	−69	−73	−67	−63	−45
−41	−49	−59	−60	−63	−52	−50	−34

块变换编码

- JPEG基本编码与解码
  - 前向DCT

−415	−29	−62	25	55	−20	−1	3
7	−21	−62	9	11	−7	−6	6
−46	8	77	−25	−30	10	7	−5
−50	13	35	−15	−9	6	0	3
11	−8	−13	−2	−1	1	−4	1
−10	1	3	−3	−1	0	2	−1
−4	−1	2	−1	2	−3	1	−2
−1	−1	−1	−2	−1	−1	0	−1

  - 量化、缩放和截断

−26	−3	−6	2	2	0	0	0
1	−2	−4	0	0	0	0	0
−3	1	5	−1	−1	0	0	0
−4	1	2	−1	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

块变换编码

- JPEG基本编码与解码
  - Z型扫描系数重排

[−26 −3 1 −3 −2 −6 2 −4 1 −4 1 1 5 0 2 0 0 −1 2 0 0 0 0 0 −1 −1 EOB]

  - 完整编码阵列

1010110 0100 001 0100 0101 100001 0110 100011 001 100011 001
001 100101 11100110 110110 0110 11110100 000 1010

  - 重新生成的量化系数阵列

−26	−3	−6	2	2	0	0	0
1	−2	−4	0	0	0	0	0
−3	1	5	−1	−1	0	0	0
−4	1	2	−1	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

块变换编码

- JPEG基本编码与解码
  - 反量化化

−416	−33	−60	32	48	0	0	0
12	−24	−56	0	0	0	0	0
−42	13	80	−24	−40	0	0	0
−56	17	44	−29	0	0	0	0
18	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

  - 反DCT变换

−70	−64	−61	−64	−69	−66	−58	−50
−72	−73	−61	−39	−30	−40	−54	−59
−68	−78	−58	−9	13	−12	−48	−64
−59	−77	−57	0	22	−13	−51	−60
−54	−75	−64	−23	−13	−44	−63	−56
−52	−71	−72	−54	−54	−71	−71	−54
−45	−59	−70	−68	−67	−67	−61	−50
−35	−47	−61	−66	−60	−48	−44	−44

块变换编码

- JPEG基本编码与解码
  - 移位128个灰度级

58	64	67	64	59	62	70	78
56	55	67	89	98	88	74	69
60	50	70	119	141	116	80	64
69	51	71	128	149	115	77	68
74	53	64	105	115	84	65	72
76	57	56	74	75	57	57	74
83	69	59	60	61	61	67	78
93	81	67	62	69	80	84	84

  - 原图与重建的差别

−6	−9	−6	2	11	−1	−6	−5
7	4	−1	1	11	−3	−5	3
2	9	−2	−6	−3	−12	−14	9
−6	7	0	−4	−5	−9	−7	1
−7	8	4	−1	6	4	3	−2
3	8	4	−4	2	6	1	1
2	2	5	−1	−6	0	−2	5
−6	−2	2	6	−4	−4	−6	10

块变换编码

- JPEG基本编码与解码



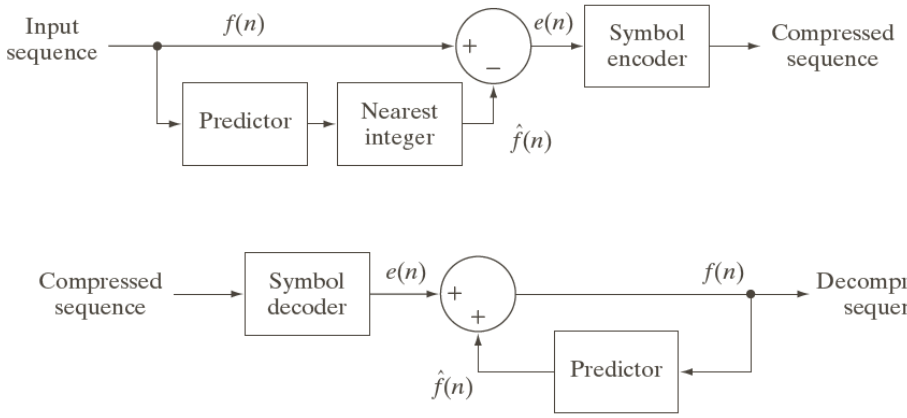
a b c
d e f

FIGURE 8.32 Two JPEG approximations of Fig. 8.9(a). Each row contains a result after compression and reconstruction, the scaled difference between the result and the original image, and a zoomed portion of the reconstructed image.

第一个结果压缩率25:1，第二个结果压缩率52:1

预测编码

- 概述
  - 预测编码通过消除紧邻像素在空间和时间上的冗余来实现
  - 一个像素的新信息定义为实际值和预测值的差值
- 无损预测编码



a
b

FIGURE 8.33 A lossless predictive coding model: (a) encoder; (b) decoder.

- 预测误差 $e(n) = f(n) - \hat{f}(n)$
- 重建 $f(n) = e(n) + \hat{f}(n)$

预测编码

- 无损预测编码
  - 多数情况下，预测可由m个样值进行线性组合

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

  - 对一幅图像，预测值可以是来自当前扫描行（一维线性预测编码）、当前行和前几个扫描行
  - 一维线性预测图像编码

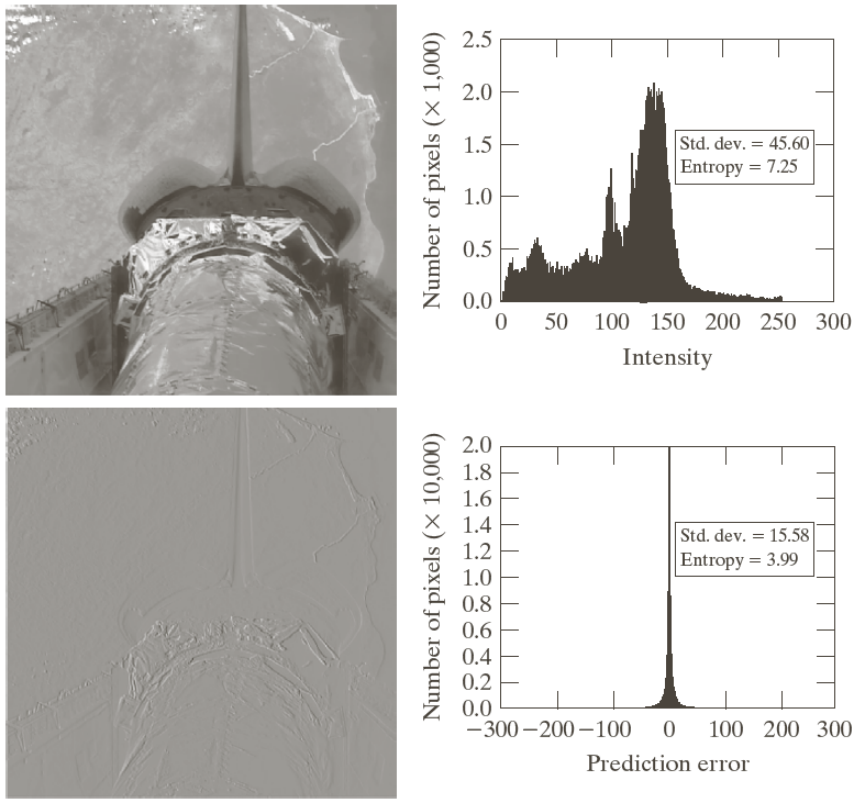
$$\hat{f}(x,y) = \text{round}\left[\sum_{i=1}^{m} \alpha_i f(x,y-i)\right]$$

预测编码

无损预测编码

一维线性预测图像编码示例1

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



a b
c d

FIGURE 8.34
(a) A view of the Earth from an orbiting space shuttle. (b) The intensity histogram of (a). (c) The prediction error image resulting from Eq. (8.2-34). (d) A histogram of the prediction error. (Original image courtesy of NASA.)

零均值拉普拉斯分布

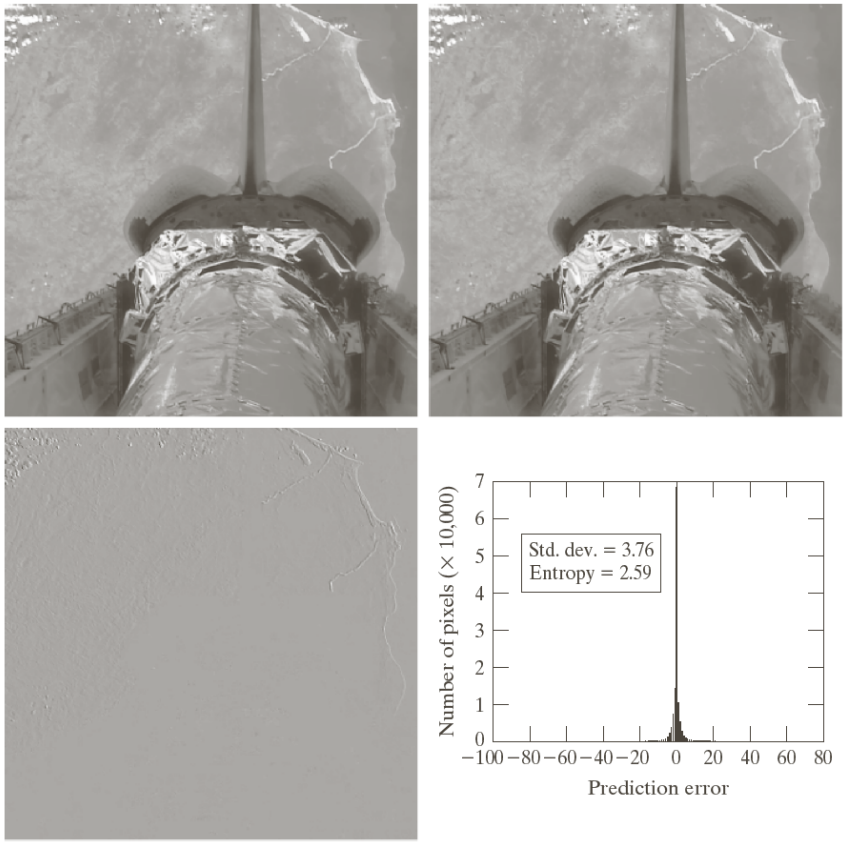
$$p_e(\text{e}) = \frac{1}{\sqrt{2}\sigma_e} e^{\frac{-\sqrt{2}|\text{e}|}{\sigma_e}}$$

预测编码

无损预测编码

一维线性预测图像编码示例2

$$\hat{f}(x,y,t) = \text{round}\left[\alpha f(x,y,t-1)\right]$$



a b
c d

FIGURE 8.35
(a) and (b) Two views of Earth from an orbiting space shuttle video. (c) The prediction error image resulting from Eq. (8.2-36). (d) A histogram of the prediction error. (Original images courtesy of NASA.)

帧内预测 vs 帧间预测
预测误差：15.58 vs 3.76

零均值拉普拉斯分布

$$p_e(\mathrm{e}) = \frac{1}{\sqrt{2}\sigma_e} e^{\frac{-\sqrt{2}|\mathrm{e}|}{\sigma_e}}$$

预测编码

- 运动补偿预测残差
  - 帧序列包含快速运动的物体
  - 相减的方式不再有效
  - 运动补偿广泛用于视频编码中
- 运动补偿基本原理
  - 将视频帧分为互不重叠的子块，称为宏块
  - 每个宏块在前一帧（称为参考帧）中找到最合适的块
  - 使用运动矢量表示当前宏块在参考帧的位置
  - 为提高运动搜索的精度，往往采用1/4运动补偿

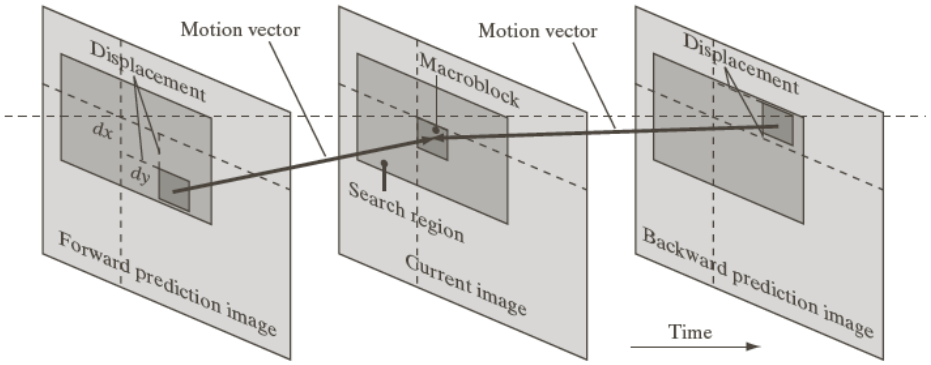
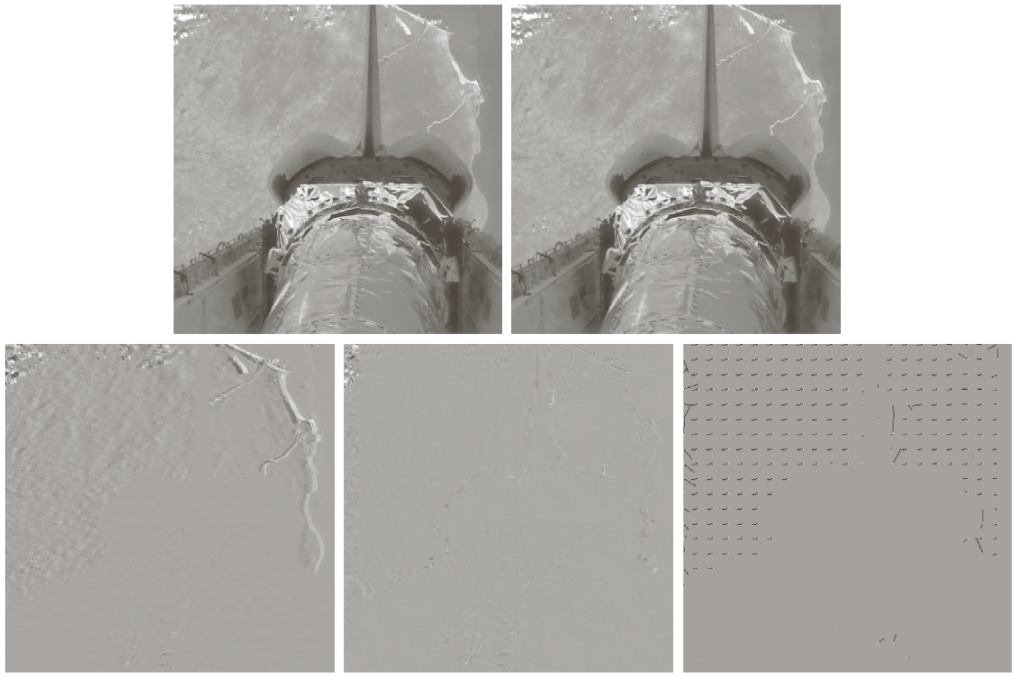


FIGURE 8.36 Macroblock motion specification.

预测编码

- 运动补偿预测示例 1



c)预测残差：12.73灰度级；熵：4.17bit/pix；C=1.92

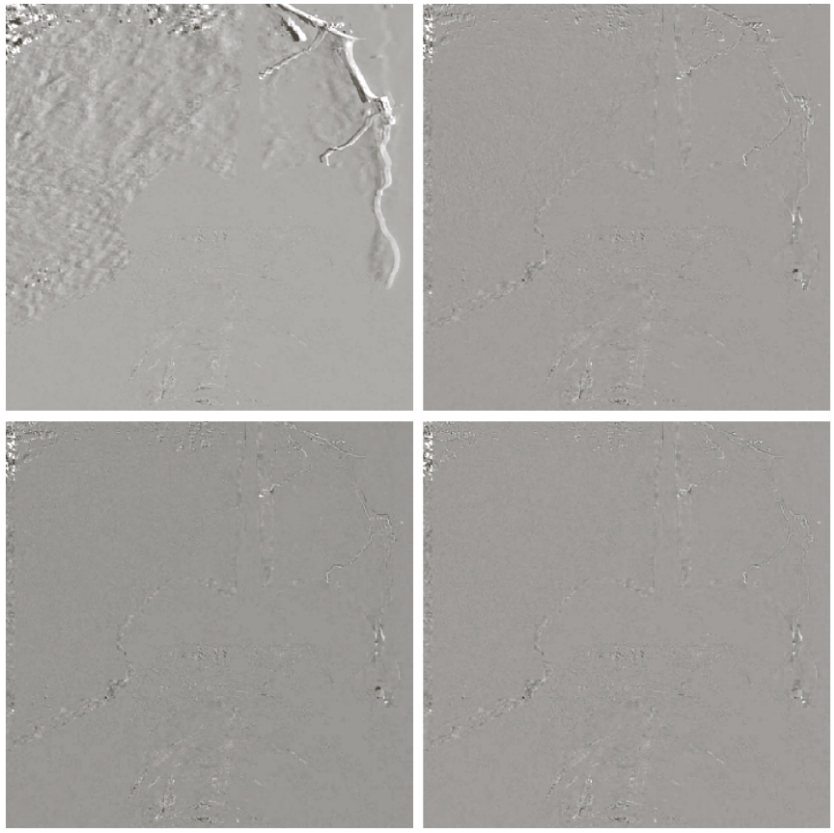
d)预测残差：5.62灰度级；熵：3.04bit/pix；C=2.63

a b
c d e

FIGURE 8.37 (a) and (b) Two views of Earth that are thirteen frames apart in an orbiting space shuttle video. (c) A prediction error image without motion compensation. (d) The prediction residual with motion compensation. (e) The motion vectors associated with (d). The white dots in (d) represent the arrow heads of the motion vectors that are depicted. (Original images courtesy of NASA.)

预测编码

- 运动补偿预测示例 2



a b
c d

FIGURE 8.38 Sub-pixel motion compensated prediction residuals: (a) without motion compensation; (b) single pixel precision; (c) $\frac{1}{2}$ pixel precision; and (d) $\frac{1}{4}$ pixel precision. (All prediction errors have been scaled to the full intensity range and then multiplied by 2 to increase their visibility.)

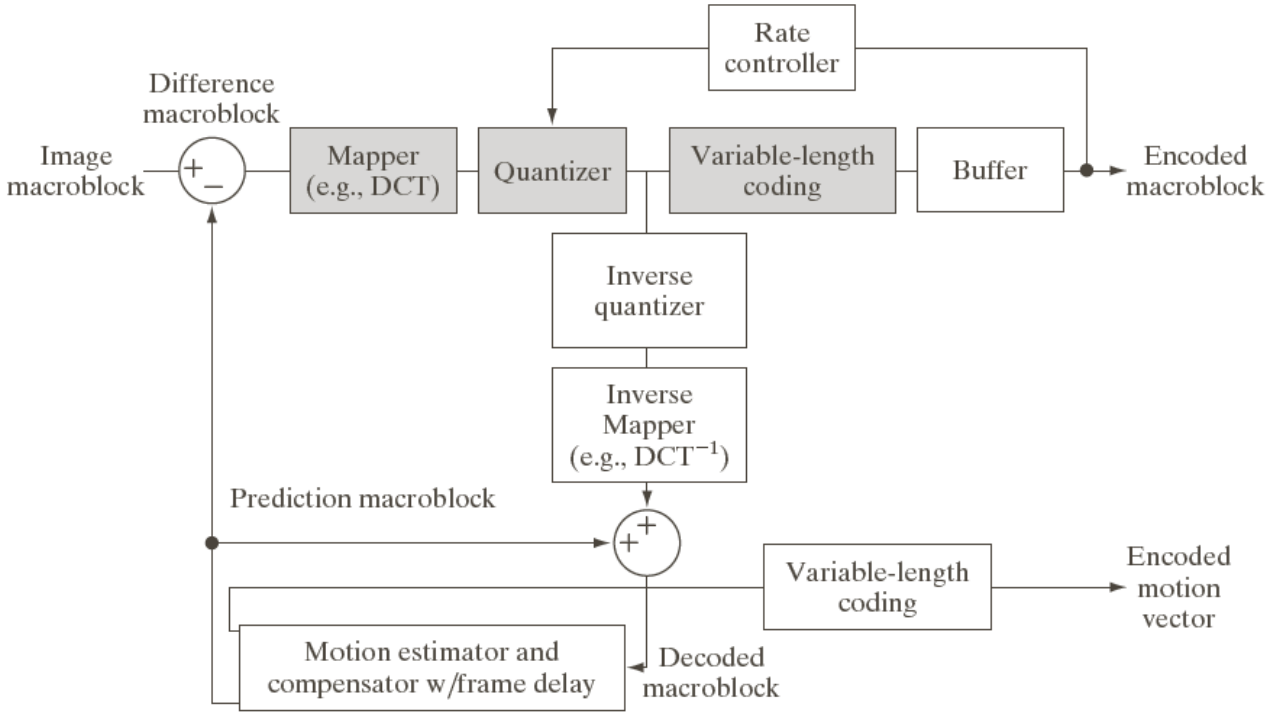
预测编码

- 运动估计的过程要求的计算量较大
- 只有编码端需要运动估计
- 视频压缩标准未包含运动估计，只在宏块维数、运动矢量精度等设置约束

Parameter	H.261	MPEG-1	H.262 MPEG-2	H.263	MPEG-4	VC-1 WMV-9	H.264 MPEG-4 AVC
Motion vector precision	1	1/2	1/2	1/2	1/4	1/4	1/4
Macroblock sizes	16 × 16	16 × 16	16 × 16 16 × 8	16 × 16 8 × 8	16 × 16 8 × 8	16 × 16 8 × 8	16 × 16 16 × 8 8 × 16 8 × 8 8 × 4 4 × 8 4 × 4
Transform	8 × 8 DCT	8 × 8 DCT	8 × 8 DCT	8 × 8 DCT	8 × 8 DCT	8 × 8 8 × 4 4 × 8 4 × 4 Integer DCT	4 × 4 8 × 8 Integer
Interframe predictions	P	P, B	P, B	P, B	P, B	P, B	P, B
I-frame intra-predictions	No	No	No	No	No	No	Yes

预测编码

- 典型的基于运动补偿的视频编码器



预测编码

- 视频压缩

5G

↓

44.56M

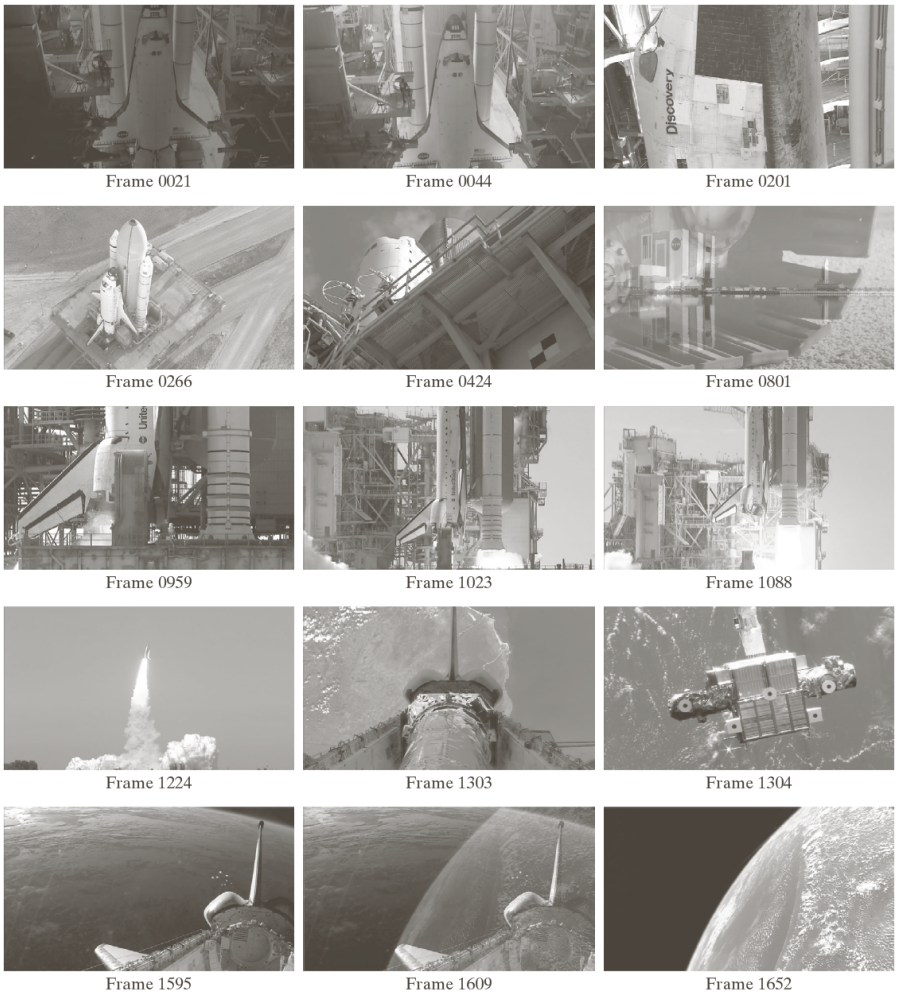
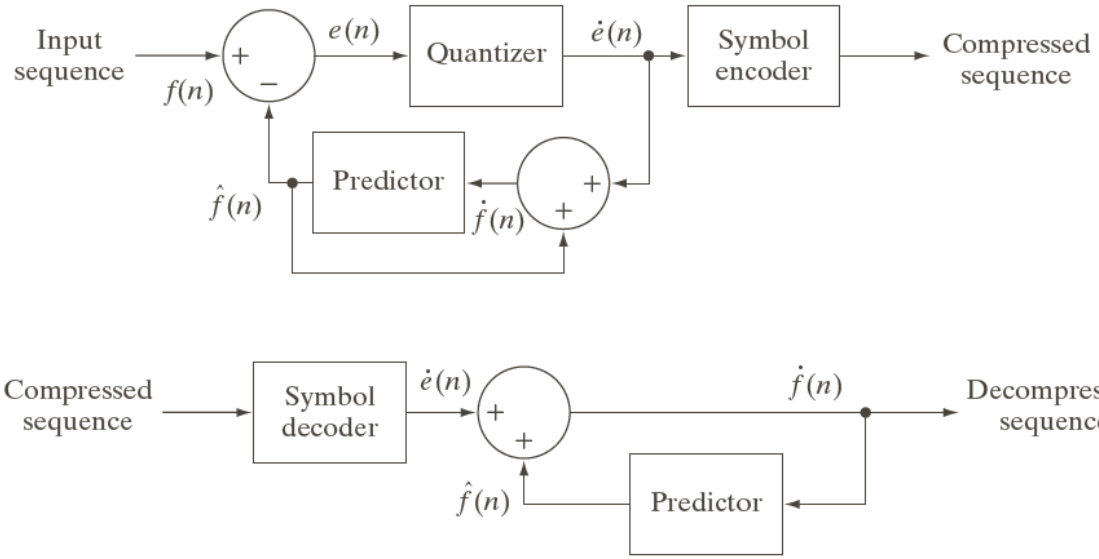


FIGURE 8.40 Fifteen frames from an 1829-frame, 1-minute NASA video. The original video is in HD full color. (Courtesy of NASA.)

预测编码

- 有损预测编码
  - 在反馈环中放置一个有损编码器的预测器
  - 闭环结构可以防止在解码器的输出处形成误差

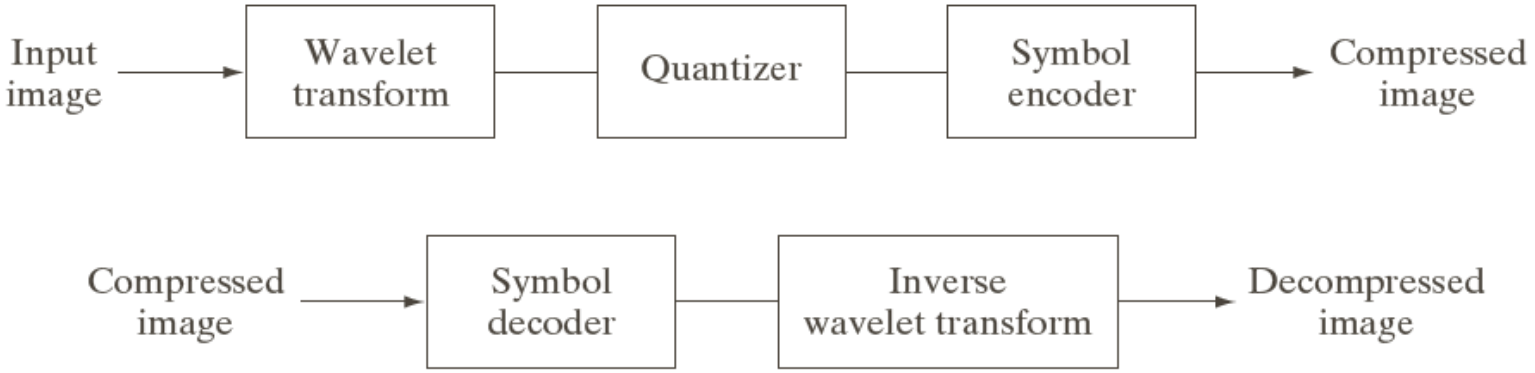


$$\dot{f}(n) = \dot{e}(n) + \hat{\dot{f}}(n)$$

小波编码

概念

- 对图像像素解除相关的变换系数进行编码，比对原图像像素本身进行编码效率更高
- 选择合适的变换基函数（小波函数），将大多数重要的视觉信息包装到少量系数中，剩下的系数可以被粗略量化甚至截取为0，而图像几乎没有失真



小波编码

小波基

- 图像编码中广泛使用的小波有：
  - ✓ Harr小波
  - ✓ Daubechies小波
  - ✓ 对称小波
  - ✓ 双正交小波

a b
c d

FIGURE 8.46 Three-scale wavelet transforms of Fig. 8.9(a) with respect to (a) Haar wavelets, (b) Daubechies wavelets, (c) symlets, and (d) Cohen-Daubechies Feauveau biorthogonal wavelets.



TABLE 8.13 Wavelet transform filter taps and zeroed coefficients when truncating the transforms in Fig. 8.46 below 1.5.

Wavelet	Filter Taps (Scaling + Wavelet)	Zeroed Coefficients
Haar (see Ex. 7.10)	2 + 2	33.8%
Daubechies (see Fig. 7.8)	8 + 8	40.9%
Symlet (see Fig. 7.26)	8 + 8	41.2%
Biorthogonal (see Fig. 7.39)	17 + 11	42.1%

小于1.5系数截短为0

小波编码

分解级别选择

- 变换分解级别的数量是影响计算复杂性和重建误差的重要因素
- 正反变换计算操作次数随分解级数的增加而增加

Decomposition Level (Scales or Filter Bank Iterations)	Approximation Coefficient Image	Truncated Coefficients (%)	Reconstruction Error (rms)
1	256 × 256	74.7%	3.27
2	128 × 128	91.7%	4.23
3	64 × 64	95.1%	4.54
4	32 × 32	95.6%	4.61
5	16 × 16	95.5%	4.63

TABLE 8.14 Decomposition level impact on wavelet coding the 512 × 512 image of Fig. 8.9(a).

*采用双正交小波和固定全局阈值为25

量化器设计

- 系数量化是影响小波编码压缩和重建误差的最重要因素
- 进一步改进
  - 引入以零为中心的较大量化间隔，称为**死区**
  - 从一个尺度到另一个尺度自适应调制量化间隔的大小

小波编码

量化器设计

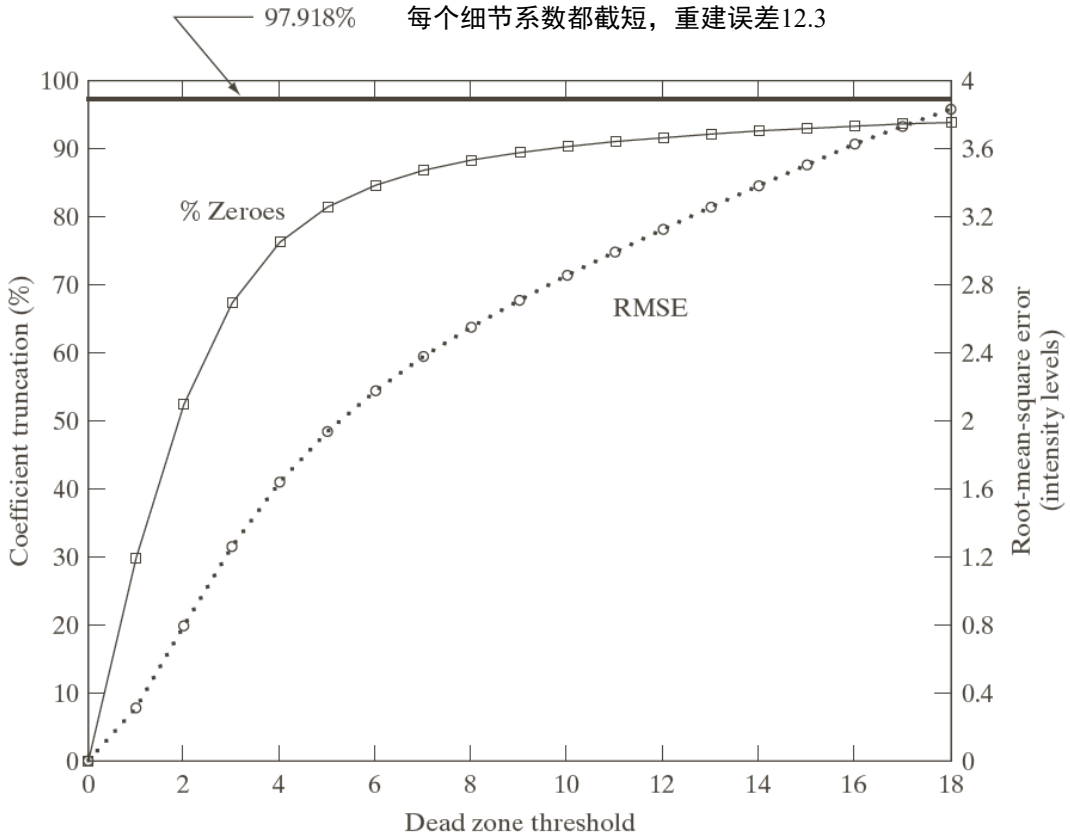


FIGURE 8.47 The impact of dead zone interval selection on wavelet coding.

死区尺寸增大时，被截断系数的数量同样增大

小波编码

- JPEG2000: 基于小波变换的图像编码标准
  - JPEG2000扩充了JPEG标准，在连续色调静止图像压缩和压缩数据的访问提供了更多灵活性
  - JPEG2000编码步骤
    - 对编码的图像样本进行直流电平平移和去相关性
    - 将各个分量分成多个像块，即被单独处理的像素的矩阵阵列
    - 计算每个像块分量的行和列的一维离散小波变换
      - 无损压缩以双正交、5-3系数尺度小波向量为基础
      - 有损压缩中，采用9-7系数尺度-小波向量

$$Y(2n+1) = X(2n+1) + \alpha\left[X(2n) + X(2n+2)\right],\ i_0 - 3 \le 2n+1 < i_1 + 3$$

$$Y(2n) = X(2n) + \beta\left[Y(2n-1) + Y(2n+1)\right],\ i_0 - 2 \le 2n < i_1 + 2$$

$$Y(2n+1) = Y(2n+1) + \gamma\left[Y(2n) + Y(2n+2)\right],\ i_0 - 1 \le 2n+1 < i_1 + 1$$

$$Y(2n) = Y(2n) + \delta\left[Y(2n-1) + Y(2n+1)\right],\ i_0 \le 2n < i_1$$

$$Y(2n+1) = -K\,Y(2n+1),\ i_0 \le 2n+1 < i_1$$

$$Y(2n) = Y(2n) / \mathrm{K},\ i_0 \le 2n < i_1$$

提升和尺度操作

小波编码

- JPEG2000
  - JPEG2000编码步骤
    - 计算每个像块分量的行和列的一维离散小波变换

Filter Tap	Highpass Wavelet Coefficient	Lowpass Scaling Coefficient
0	−1.115087052456994	0.6029490182363579
±1	0.5912717631142470	0.2668641184428723
±2	0.05754352622849957	−0.07822326652898785
±3	−0.09127176311424948	−0.01686411844287495
±4	0	0.02674875741080976

TABLE 8.15 Impulse responses of the low- and highpass analysis filters for an irreversible 9-7 wavelet transform.

小波编码

JPEG2000

JPEG2000编码步骤

- 变换生成四个子带，分别是像块分量的低分辨率近似及分量的水平、垂直和对角频率特征。将后续迭代限制为前面的分解的近似系数，把该变换重复N次，可得到N尺度小波变换

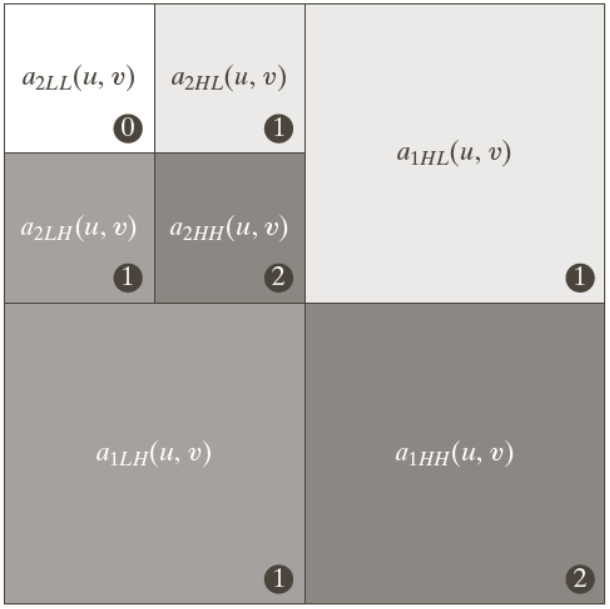


FIGURE 8.48 JPEG 2000 two-scale wavelet transform tile-component coefficient notation and analysis gain.

小波编码

- JPEG2000
  - JPEG2000编码步骤
    - 量化，子带b的系数 $a_b(u,v)$ 量化为 $q_b(u,v)$

$$q_b(u,v) = sign\left[a_b(u,v)\right] floor\left[\frac{|a_b(u,v)|}{\Delta_b}\right]$$

$$\Delta_b = 2^{R_b - g_b}\left(1 + \frac{\mu_b}{2^{11}}\right)$$

    - 每个像块分量的子带系数被排列为称为码块的矩形块，码块被单独编码，一次一个比特平面

小波编码

JPEG2000

从a)到d)压缩比C为25，52，75，105；

a) 误差：JPEG 5.4
J2000 3.86;

b) 误差： JPEG 10.7
J2000 5.77



FIGURE 8.49 Four JPEG-2000 approximations of Fig. 8.9(a). Each row contains a result after compression and reconstruction, the scaled difference between the result and the original image, and a zoomed portion of the reconstructed image. (Compare the results in rows 1 and 2 with the JPEG results in Fig. 8.32.)

数字图像水印

- 数字图像水印处理是把数据插入到一幅图像中的过程，从而形成对图像的保护。
- 简单的可见水印：

$$f_w = (1-\alpha)f + \alpha w$$

其中 f 为原图，w 为水印



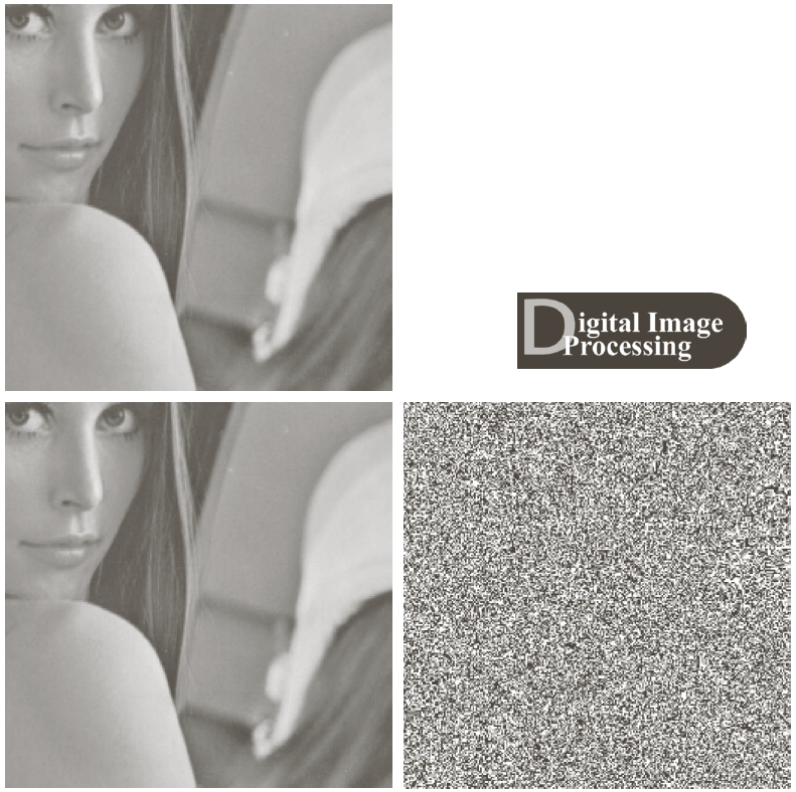
a
b c

FIGURE 8.50 A simple visible watermark: (a) watermark; (b) the watermarked image; and (c) the difference between the watermarked image and the original (non-watermarked) image.

数字图像水印

- 不可见水印
  - 不可见水印裸眼是看不见的
  - 使用合适的解码算法可以恢复原图
  - 不可见水印是通过作为可见的冗余信息插入来保证的

$$f_w = 4\left(\frac{f}{4}\right) + \frac{w}{64}$$

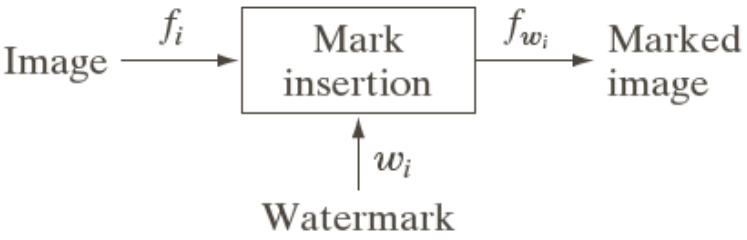


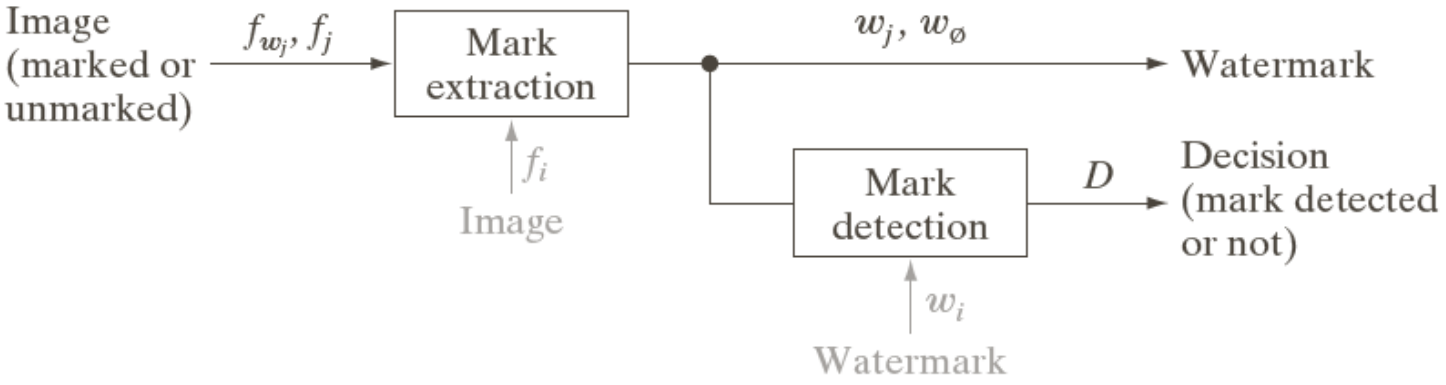
a b
c d

FIGURE 8.51 A simple invisible watermark: (a) watermarked image; (b) the extracted watermark; (c) the watermarked image after high quality JPEG compression and decompression; and (d) the extracted watermark from (c).

数字图像水印

- 典型的图像水印处理系统





数字图像水印

- 基于DCT的不可见的鲁棒水印

① 计算被加入水印图像的二维DCT

② 按幅值定位它的K个最大系数$c_1,c_2,\ldots,c_k$.

③ 通过生成一个K元素伪随机数序列$w_1,w_2,\ldots,w_k$创建一个水印，这些伪随机数取自 $\mu=0,\sigma^2=1$ 的高斯分布

④ 使用下式将步骤3得到的水印嵌入步骤2的K个最大DCT系数中

$$c_i' = c_i * (1+\alpha\omega_i),\ 1 \le i \le K$$

⑤ 计算步骤4结果的反DCT

数字图像水印

- 基于DCT的不可见的鲁棒水印



a b
c d

FIGURE 8.53 (a) and (c) Two watermarked versions of Fig. 8.9(a); (b) and (d) the differences (scaled in intensity) between the watermarked versions and the unmarked image. These two images show the intensity contribution (although scaled dramatically) of the pseudo-random watermarks on the original image.

基于DCT的不可见的鲁棒水印的判断

- 确定一幅特定图像是否是一幅事先加水印图像的副本，该图像具有水印$w_1,w_2,\ldots,w_k$和DCT系数$c_1,c_2,\ldots,c_k$.
  - 计算问题中图像的二维DCT
  - 提取K个DCT系数，记为 $\hat{c}_1,\hat{c}_2,...,\hat{c}_K$ 。若问题图像是事先加过水印的图像（未被修改），则对$1\le i\le K$有 $\hat{c}_i=c_i'$ 。否则若它是加过水印图像的修改副本，则 $\hat{c}_i\approx c_i'$
  - 使用下式计算 $\hat{\omega}_1,\hat{\omega}_2,...,\hat{\omega}_K$

$$\hat{\omega}_i=\frac{\hat{c}_i-c_i}{\alpha c_i},\ 1\le i\le K$$

  - 使用相关系数度量 $\hat{\omega}_1,\hat{\omega}_2,...,\hat{\omega}_K$ 和$\omega_1,\omega_2,...,\omega_K$ 的相似度

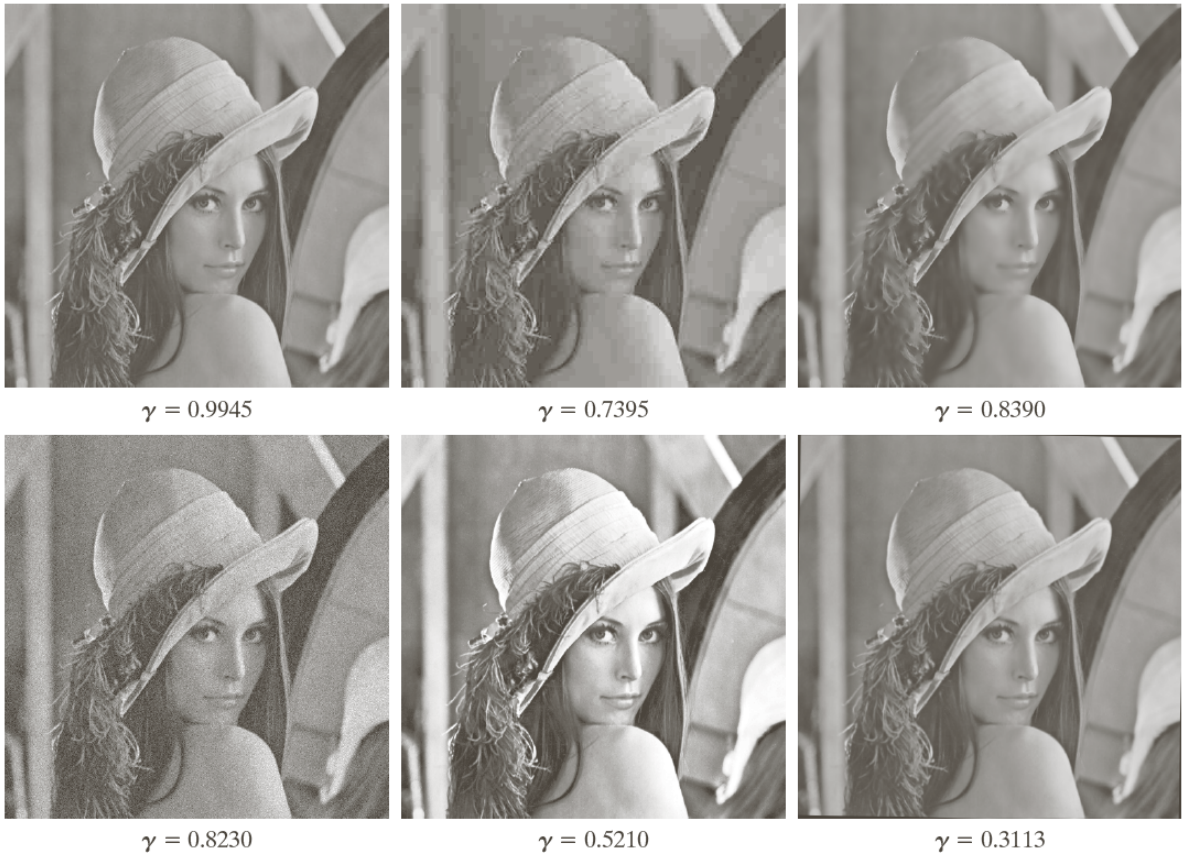
$$\gamma=\frac{\sum_{i=1}^{K}(\bar{\hat{\omega}}_i-\bar{\hat{\omega}})(\omega_i-\bar{\omega})}{\sqrt{\sum_{i=1}^{K}(\bar{\hat{\omega}}_i-\bar{\hat{\omega}})^2*\sum_{i=1}^{K}(\omega_i-\bar{\omega})^2}},\ 1\le i\le K$$

基于DCT的不可见的鲁棒水印的判断

- 确定一幅特定图像是否是一幅事先加水印图像的副本，该图像具有水印$w_1,w_2,\ldots,w_k$和DCT系数$c_1,c_2,\ldots,c_k$.
  - 将度量相似度 γ 和一个预定义的阈值T进行比较，进行二值检测判决。
    - ✓ D=1代表水印$\omega_1,\omega_2,...,\omega_K$ 存在，D=0代表水印$\omega_1,\omega_2,...,\omega_K$ 不存在

$$D=\begin{cases}1, & \gamma \geq T\\ 0, & other\end{cases}$$

数字图像水印



a b c
d e f

FIGURE 8.54 Attacks on the watermarked image in Fig. 8.53(a): (a) lossy JPEG compression and decompression with an rms error of 7 intensity levels; (b) lossy JPEG compression and decompression with an rms error of 10 intensity levels (note the blocking artifact); (c) smoothing by spatial filtering; (d) the addition of Gaussian noise; (e) histogram equalization; and (f) rotation. Each image is a modified version of the watermarked image in Fig. 8.53(a). After modification, they retain their watermarks to varying degrees, as indicated by the correlation coefficients below each image.