



图像压缩编码

Image Compression Coding





概述 Introduction

一、图像编码技术的研究背景

1. 信息传输方式发生了很大的改变

- 通信方式的改变

文字+语音 → 图像+文字+语音

- 通信对象的改变

人与人 → 人与机器，机器与机器





2. 图像传输与存储需要的信息量空间

以彩色视频信息为例：

对于电视画面的分辨率640*480的彩色图像，每秒30帧，则一秒钟的数据量为：

$$640*480*24*30=221.12\text{M}$$

所以播放时，需要221Mbps的通信回路。

参考数据：宽带网为1M~10M，电话线为56K。

存储时，1张CD可存640M，则仅可以存放2.89秒的数据。





□为什么要进行图像压缩？

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相当大的困难。要占用很多的资源，花很高的费用。

如一幅512x512的灰度图象的比特数为
 $512 \times 512 \times 8 = 256k$

再如一部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



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

如一张CD光盘可存600兆字节数据，这部电影光图像（还有声音）就需要160张CD光盘用来存储。

对图像数据进行压缩显得非常必要。

返回



由于通信方式和通信对象的改变带来的最大问题是：

传输带宽、速度、存储器容量的限制。

给我们带来一个难题，也给了我们一个机会：

如何用软件的手段来解决硬件上的物理极限。

有效的软件解决办法是对图像数据进行压缩！

二、图像数据压缩的可能性

图像数据本身固有的**冗余性**和**相关性**，使得**压缩图像数据成为可能**。

冗余

像素间冗余（相关性）

Interpixel Redundancy

信息熵冗余（编码冗余）

Coding Redundancy

心理视觉冗余

Psychovisual Redundancy

其它冗余

针对数据冗余的类型不同,可以有多种不同的数据压缩方法。





□ 图像数据压缩的可能性

- ✚ 一般原始图像中存在很大的冗余度。
- ✚ 用户通常允许图像失真。
- ✚ 当信道的分辨率不及原始图像的分辨率时，降低输入的原始图像的分辨率对输出图像分辨率影响不大。
- ✚ 用户对原始图像的信号不全都感兴趣，可用特征提取和图像识别的方法，丢掉大量无用的信息。提取有用的信息，使必须传输和存储的图像数据大大减少。



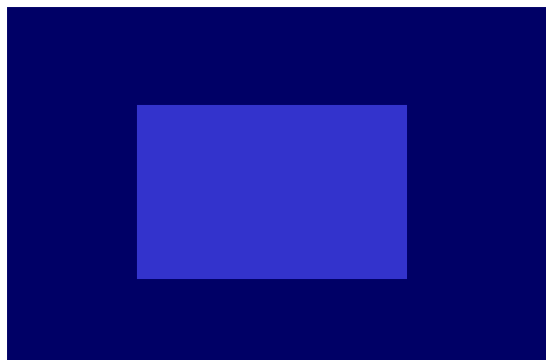
返回



□ 常见的数据冗余

■ (1) 编码冗余:

如果一个图像的灰度级编码，使用了多于实际需要的编码符号，就称该图像包含了编码冗余。



例：如果用8位表示该图像的像素，我们就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，用一位即可表示。



■ (2) 像素冗余:

由于任何给定的像素值，原理上都可以通过它的邻居预测到，单个像素携带的信息相对是小的。

对于一个图像，很多单个像素对视觉的贡献是冗余的。这是建立在对邻居值预测的基础上。

原始图像越有规则，各像素之间的相关性越强，它可能压缩的数据就越多。





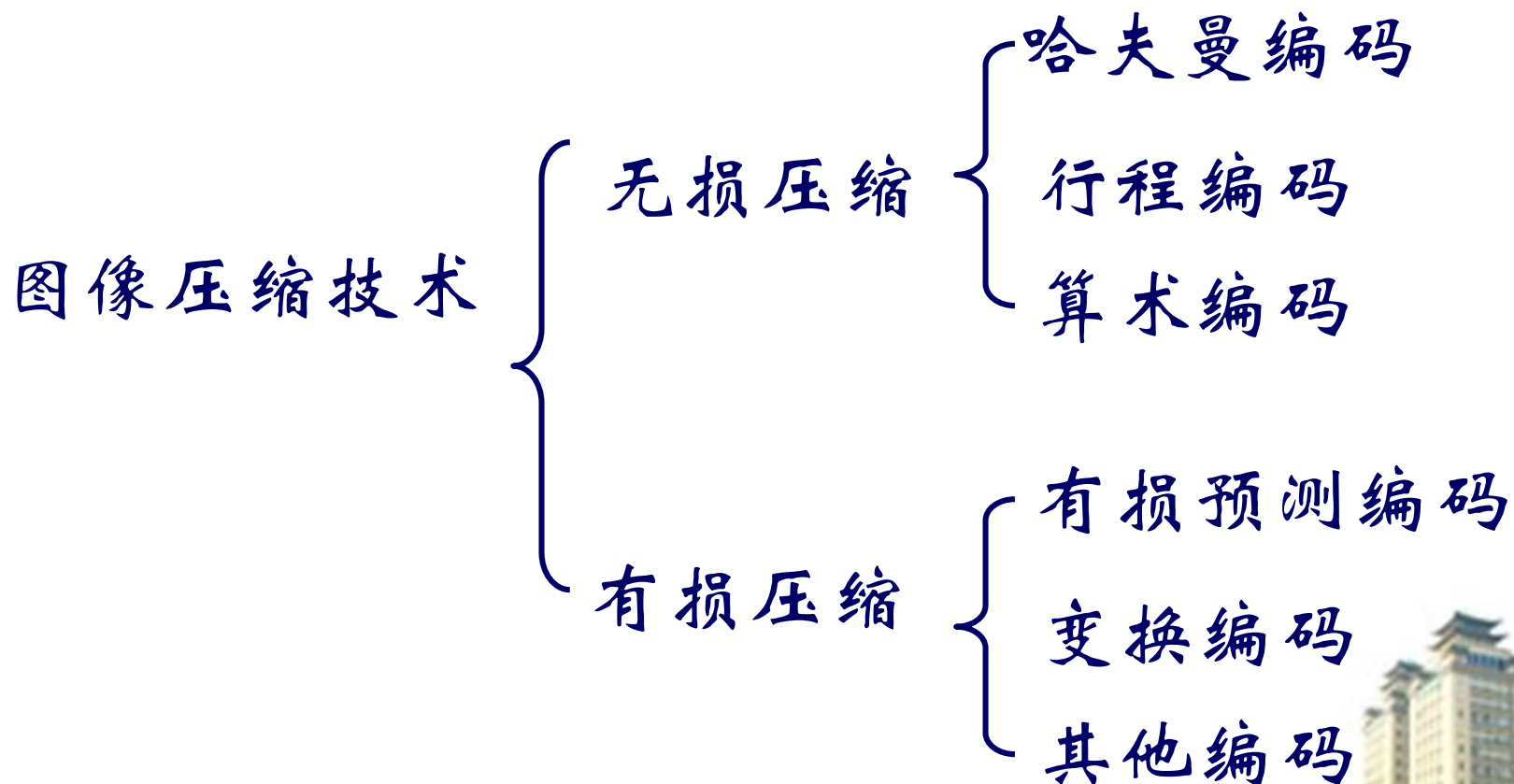
■ (3)视觉心理冗余：

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，这种信息就被称为视觉心理冗余。





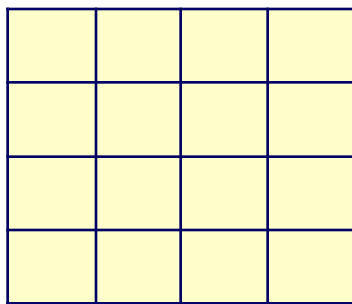
□常用的压缩编码方法





✘ 无损压缩算法中删除的仅仅是图像数据中冗余的信息，因此在解压缩时能精确恢复原图像，无损压缩的压缩比很少能超过3: 1,常用于要求高的场合。

- 图像冗余无损压缩的原理



RGB	RGB	RGB	RGB
RGB	RGB	RGB	RGB
RGB	RGB	RGB	RGB
RGB	RGB	RGB	RGB



16	RGB
----	-----

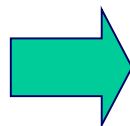
从原来的 $16 \times 3 \times 8 = 284\text{bits}$ 压缩为: $(1+3) \times 8 = 32\text{bits}$



※有损压缩是通过牺牲图像的准确率以实现较大的压缩率，如果容许解压图像有一定的误差，则压缩率可显著提高。有损压缩在压缩比大于30:1时仍然可重构图像，而如果压缩比为10:1到20:1，则重构的图像与原图几乎没有差别

• 图像冗余有损压缩的原理

36	35	34	34	34
34	34	32	34	34
33	37	30	34	34
34	34	34	34	34
34	35	34	34	31



34	34	34	34	34
34	34	34	34	34
34	34	34	34	34
34	34	34	34	34
34	34	34	34	34



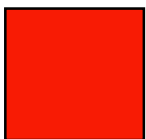
25	34
----	----



- 图象的视觉冗余（彩色）



$$\rightarrow 2^8 * 2^8 * 2^8 = 2^{24}$$
$$2^{24} = 16,777,216$$



(248,27,4)



(251,32,15)



(248,27,4)



(248,27,4)





□ 图像压缩的目的

图像数据压缩的目的是在满足一定图像质量条件下，用尽可能少的比特数来表示原始图像，以提高图像传输的效率和减少图像存储的容量。在信息论中称为信源编码。

图像从结构上大体上可分为两大类，一类是具有一定图形特征的结构，另一类是具有一定概率统计特性的结构。

基于不同的图像结构特性，应采用不同的压缩编码方法。



返回



□ 图像数据压缩技术的重要指标

(1) **压缩比**：图像压缩前后所需的信息存储量之比，压缩比越大越好。

(2) **压缩算法**：利用不同的编码方式，实现对图像的数据压缩。

(3) **失真性**：压缩前后图像存在的误差大小。





全面评价一种编码方法的优劣，除了看它的**编码效率**、**实时性**和**失真度**以外，还要看它的**设备复杂程度**，是否**经济与实用**。

常采用混合编码的方案，以求在性能和经济上取得折衷。

随着计算方法的发展，使许多高效而又比较复杂的编码方法在工程上有实现的可能。



返回



三、图像数据压缩方法分类

1. 从图像编码方法对编码分类

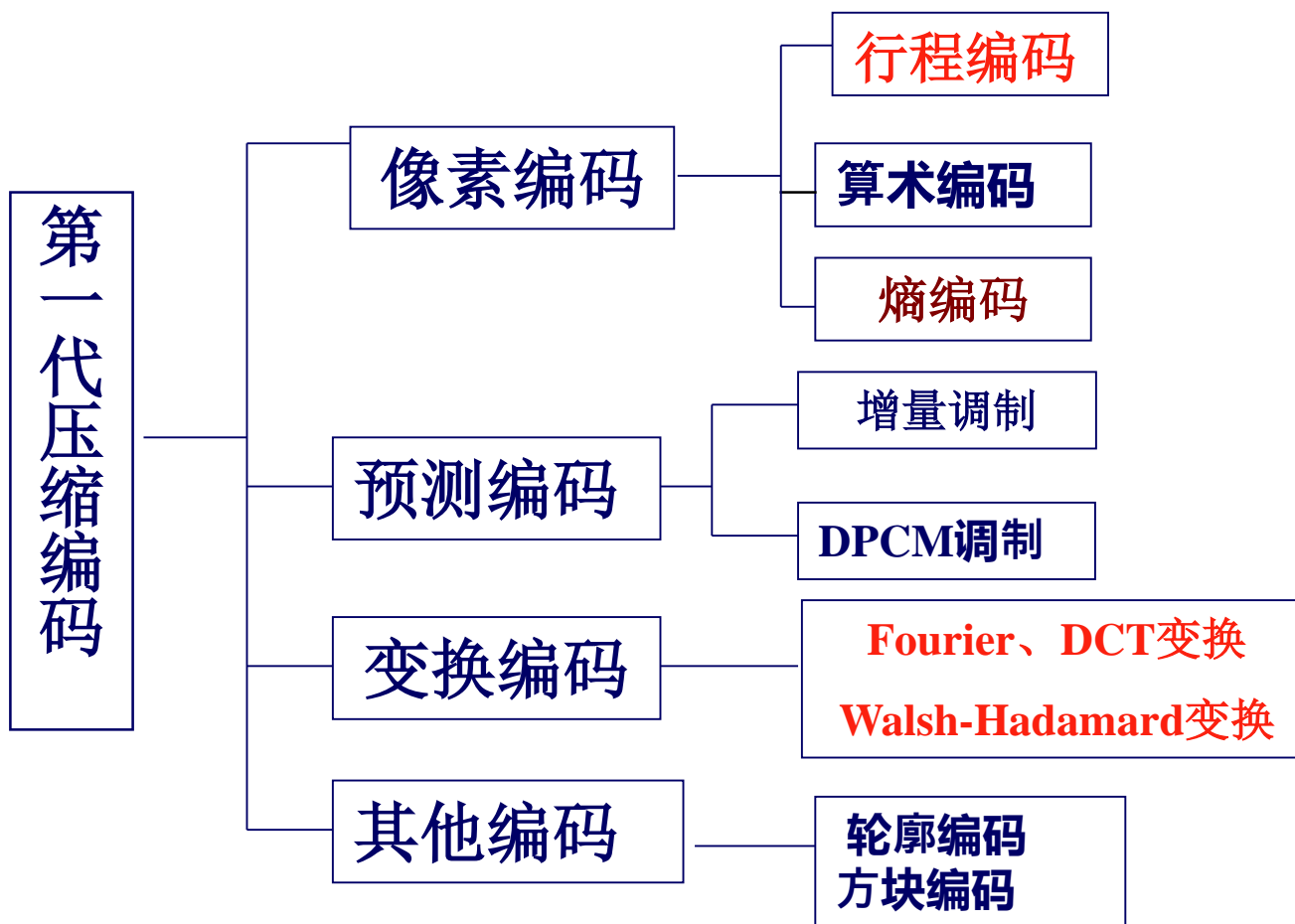
- **第一代压缩编码**

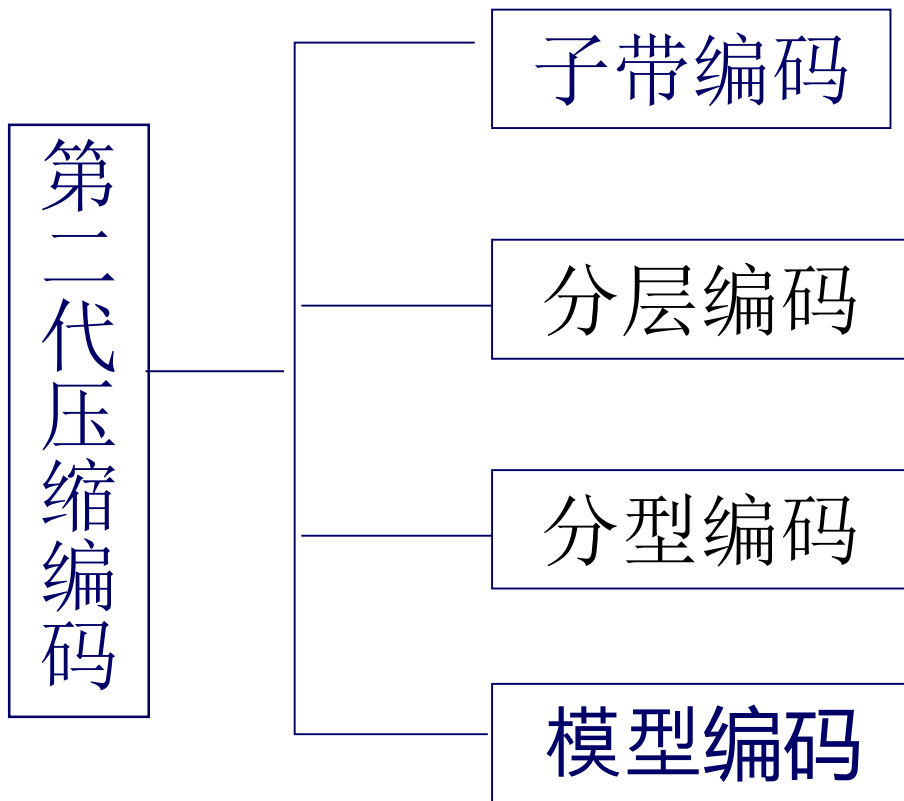
八十年代以前，主要是根据传统的信源编码方法。

- **第二代压缩编码**

八十年代以后，突破信源编码理论，结合分形、模型基、神经网络、小波变换等数学工具，充分利用视觉系统生理心理特性和图像信源的各种特性。

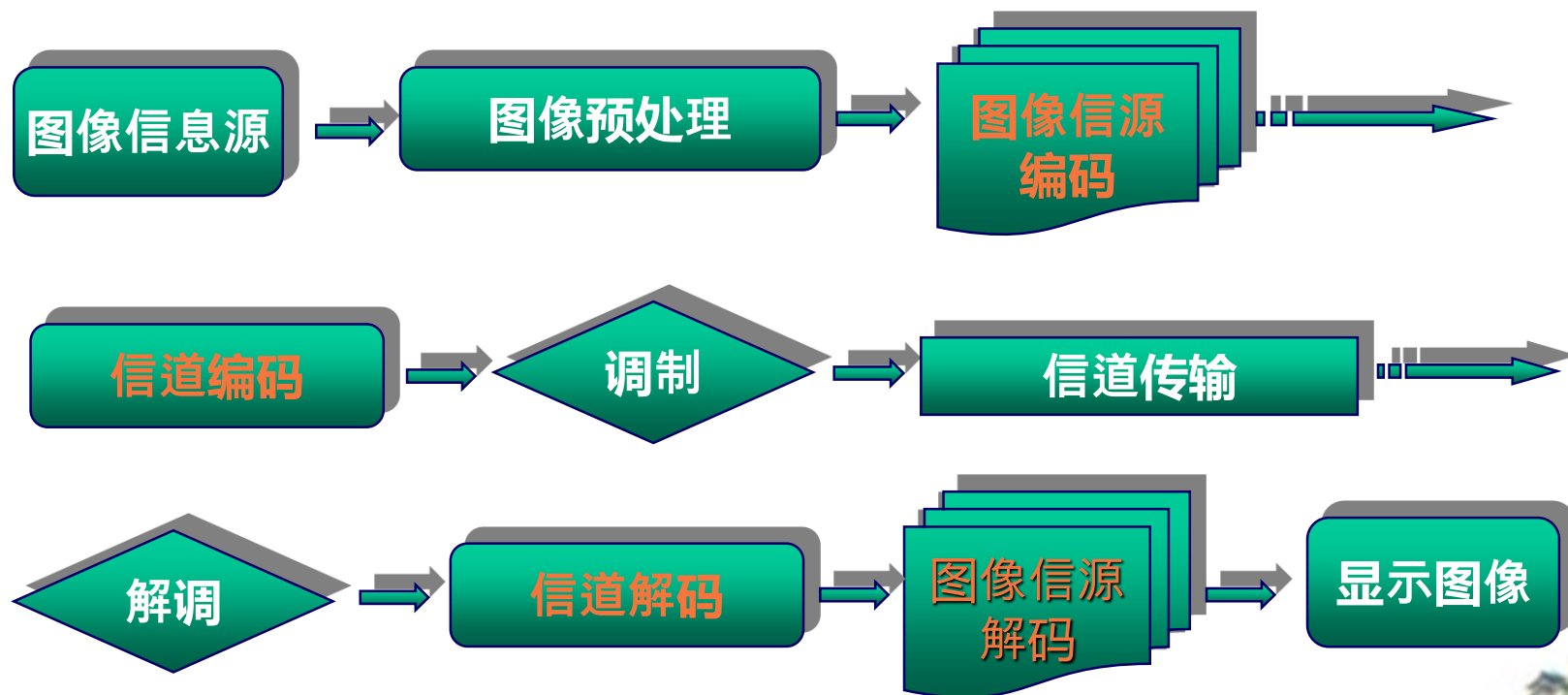








四、图像数据压缩与通信系统





□ 图像编码中的保真度准则

图像信号在编码和传输过程中会产生误差，尤其是在有损压缩编码中，产生的误差应在允许的范围之内。在这种情况下，保真度准则可以用来衡量编码方法或系统质量的优劣。通常，这种衡量的尺度可分为客观保真度准则和主观保真度准则。





保真度准则 Fidelity Criteria

保真度准则： 图像的编码质量评价

定量分析丢失信息的性质和范围, 包括(1) 客观保真度准则 (2) 主观保真度准则

1.客观保真度准则Objective Fidelity Criteria

当信息损失的程度可以表示成初始图像或输入图像以及先被压缩而后被解压缩的输出图像的函数时,就说这个函数是基于**客观保真度准则**的。

令 $f(x, y)$ 表示输入图像,

$\hat{f}(x, y)$ 表示由对输入先压缩后解压缩得到的 $f(x, y)$ 的估计量

则 $f(x, y)$ 和 $\hat{f}(x, y)$ 之间的误差 $e(x, y)$ 可定义为:

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





保真度准则：

总体误差为：

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

均方根误差为：

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

均方信噪比：

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





除了均方信噪比外，还有基本信噪比，它用分贝表示压缩图像的定量性评价。

设：

基本信噪比定义为：

$$\bar{f} = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [f(x, y)]$$
$$SNR = 10 \lg \left[\frac{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [f(x, y) - \bar{f}]^2}{\sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [f(x, y) - \hat{f}(x, y)]^2} \right]$$

峰值信噪比（PSNR）定义为：

$$PSNR = 10 \lg \left[\frac{f_{\max}^2}{\frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} [f(x, y) - \hat{f}(x, y)]^2} \right]$$

f_{\max} 为图像 $f(x, y)$ 的最大值。





2. 主观保真度准则

图像处理的结果,大多是给人观看,由研究人员来解释的,因此,图像质量的好坏,既与图像本身的客观质量有关,也与视觉系统的特性有关。

有时候,客观保真度完全一样的两幅图像可能会有完全不同的视觉质量,所以又规定了主观保真度准则,这种方法是把图像显示给观察者,然后把评价结果加以平均,以此来评价一幅图像的主观质量。

另外一种方法是规定一种绝对尺度,如:





- 1) 优秀高质量图像;
- 2) 好的是可供观赏的高质量的图像,干扰并不令人讨厌;
- 3) 可通过的图像质量可以接受,干扰不讨厌;
- 4) 边缘的图像质量较低,希望能加以改善,干扰有些讨厌;
- 5) 劣等的图像质量很差,尚能观看,干扰显著地令人讨厌;
- 6) 不能用图像质量非常之差,无法观看。





2.主观保真度准则 Subjective Fidelity Criteria

得分	第一种评价标准	第二种评价标准
5	优秀	没有失真的感觉
4	良好	感觉到失真，但没有不舒服的感觉
3	可用	感觉有点不舒服
2	较差	感觉较差
1	差	感觉非常不舒服

3.压缩比Compression Ratio

一般定义为原始图像每个像素的平均比特数 n_1 和编码后每个像素的平均比特数 n_2 的比值。

$$C_R = \frac{n_1}{n_2}$$





熵编码 Entropy Coding

一、信息论基础 Information Theory Fundamentals

1. 图像的熵 (Entropy)

设数字图像像素的灰度集合为 $\{w_1, w_2, \dots, w_M\}$ ，其对应的概率分别为 p_1, p_2, \dots, p_M ，按信息论中信源熵的定义，可以定义图像的熵 H 为：

$$H = - \sum_{k=1}^M p_k \log_2 p_k \quad (\text{bit})$$

由上述定义可以看到：

图像的熵 H 是表示其各个灰度级比特数的统计平均值。





例如：

①设 随机序列 M 由8个变量组成，等概率出现，
即 $p_1 = p_2 = \dots = p_8$ ，则：

$$H = -\sum_{k=1}^8 p_k \log_2 p_k = -\left(\frac{1}{8} \log_2 \frac{1}{8}\right) \times 8 = 3 \quad (\text{bit})$$

②设 随机序列 M 由8个变量组成， $p_1=1$ ， $p_2 = \dots = p_8 = 0$ ，
则：

$$H = -\log_2 1 = 0 \quad (\text{bit})$$

因此，当 M 等于8时， H 的范围从 0 到 3，
即 $H=0 \sim \log_2 M$ ，其中 $H=3$ 说明信号的随机程度最大。





2. 平均码字长度 Average Codeword Length

设 N_k 为数字图像第 k 个码字 c_k 的长度(二进制数的位数)，其对应出现的概率为 p_k ，则该数字图像的码字平均长度定义为：

$$\bar{N} = \sum_{k=1}^M N_k p_k \quad (\text{bit})$$

3. 编码效率 Coding Efficiency

定义数字图像编码的效率为：

$$\eta = \frac{H}{\bar{N}} * 100\%$$

4. 冗余度 Redundancy

定义图像编码的冗余度为：

$$R_d = 1 - \eta$$





根据香农信源编码信息论：

在 $\bar{N} \geq H$ 情况下总可以设计出某种无失真编码方法，若 \bar{N} 接近于 H ，则说明码编的较好，称为最佳编码。若要求编码结果 $\bar{N} < H$ ，则必然要丢失信息而引起图像失真。

二、变长编码定理 Variable-Length Coding Theorem

定理：

在变长编码中，若对出现概率大的信息赋予短码字，而对于出现概率小的信息赋予长码字，如果码字长度严格按照所对应符号出现的概率大小而逆序排列，则此种编码结果的平均码字长度一定小于其它任何排列形式得到的编码。



例如：

输入数据	W_1	W_2	W_3	W_4
概率	$1/2$	$1/4$	$1/8$	$1/8$

图像的熵：
$$H = - \sum_{k=1}^4 p_k \log_2 p_k = 1 \frac{3}{4} \text{ (bit)}$$

采用等长编码：

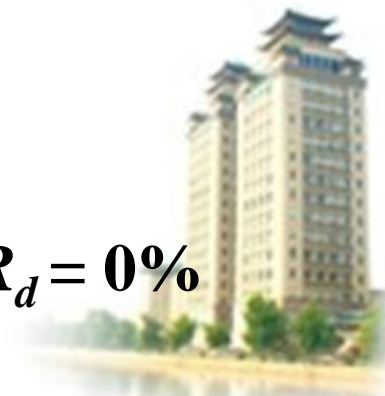
W_1	W_2	W_3	W_4
00	01	10	11

平均码长 $R=2(\text{bit})$ ，编码效率 $\eta=87.5\%$ ， $R_d=12.5\%$

采用不等长编码：

W_1	W_2	W_3	W_4
0	10	110	111

平均码长 $R=7/4(\text{bit})$ ，编码效率 $\eta=100\%$ ， $R_d=0\%$





6.3.1 霍夫曼编码 Huffman Coding

Huffman 编码是根据可变长度编码定理，应用*Huffman* 编码 算法而产生的一种编码方法，它的平均码字长度在系统的输入概率集合下，比其它唯一可译码都小。因此也称为**紧凑码**。

Huffman 编码的原则是概率大的信息用短码，而概率小的信息用长码，即：

若： $p_1(w_1) > p_2(w_2) > \dots \dots > p_M(w_M)$

则取： $N_1(c_1) < N_2(c_2) < \dots \dots < N_M(c_M)$



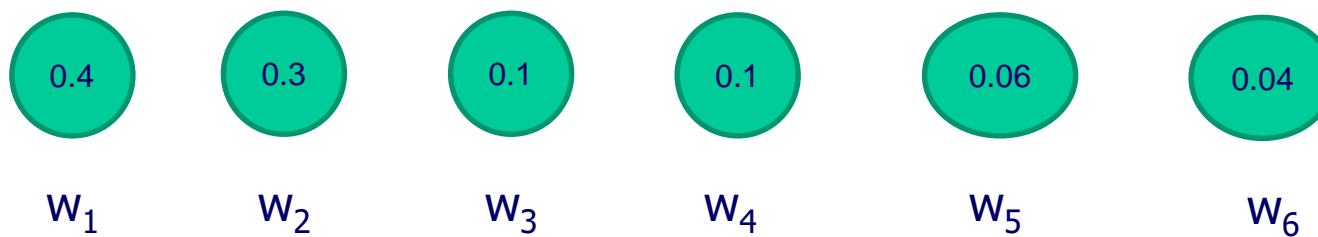


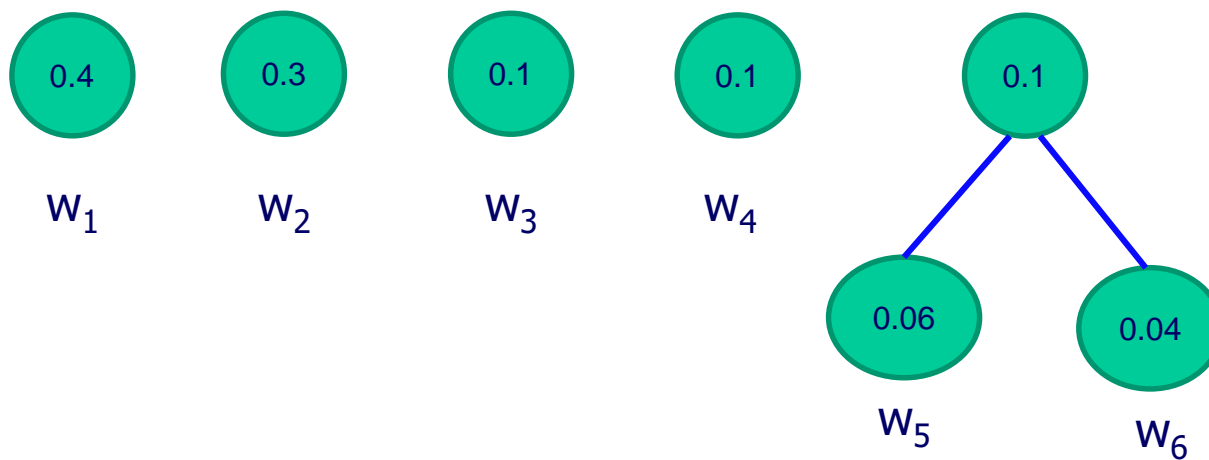
Huffman 编码的编码步骤:

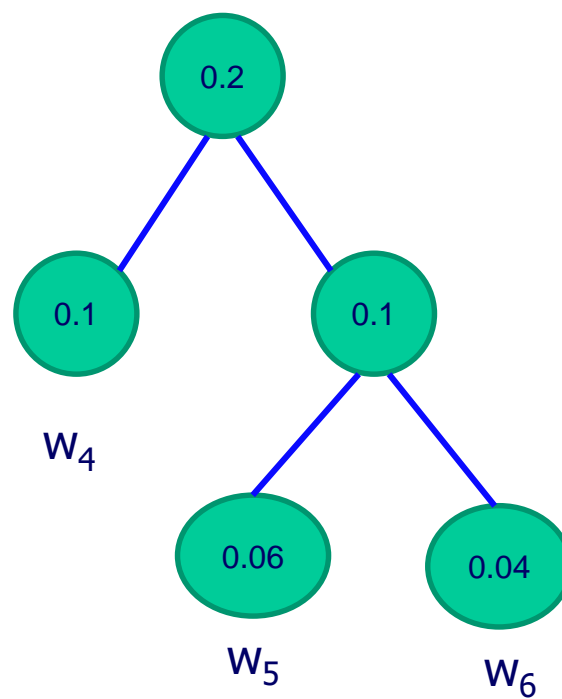
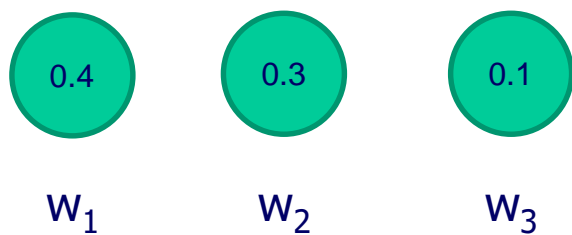
- ①将信源符号（图像的灰度等级）按概率由大到小排列，
概率相同的可以任意放
- ②将两个最小概率相加，形成新的概率集合，
并按①的原则重新排队
- ③重复②的过程，直到只剩下两个概率为止
- ④分配码字进行编码，原则是从后到前，
左0右1(或左1右0)

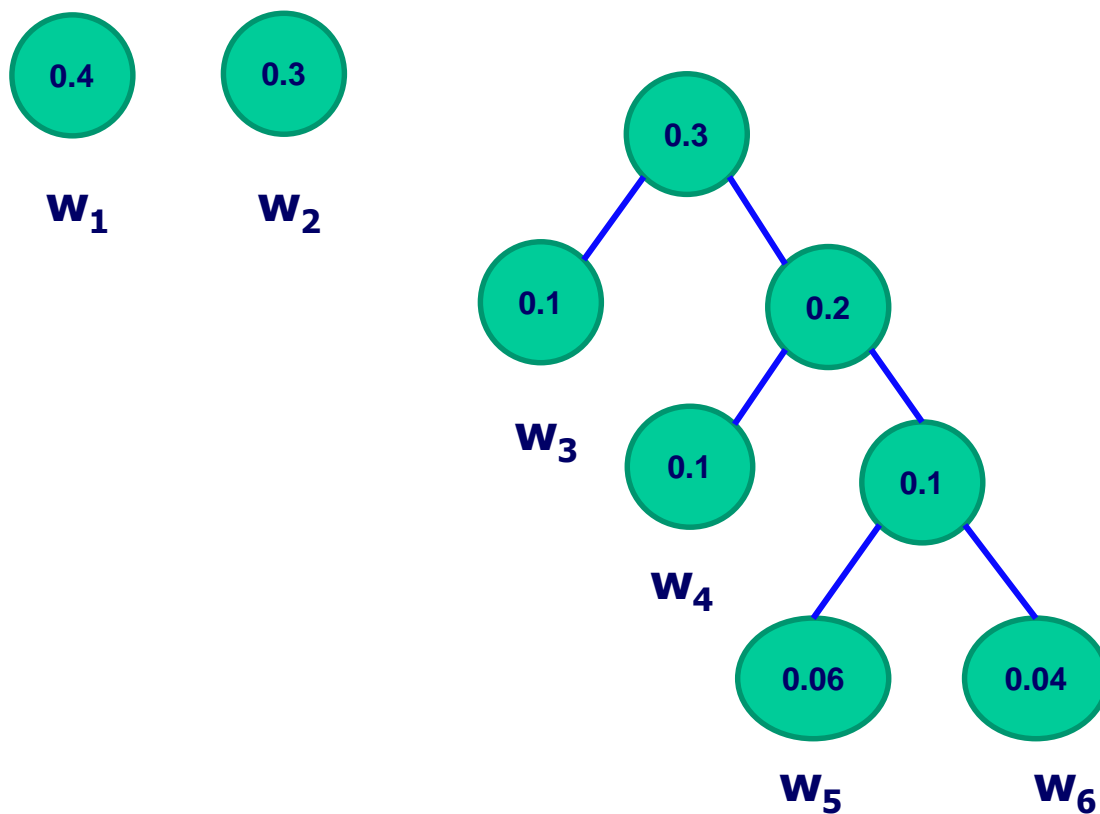
例：设一幅图像的六个灰度等级 $w_1, w_2, w_3, w_4, w_5, w_6$ 出现的概率分别为**0.4, 0.3, 0.1, 0.1, 0.06, 0.04**，对此图像进行**Huffman**编码





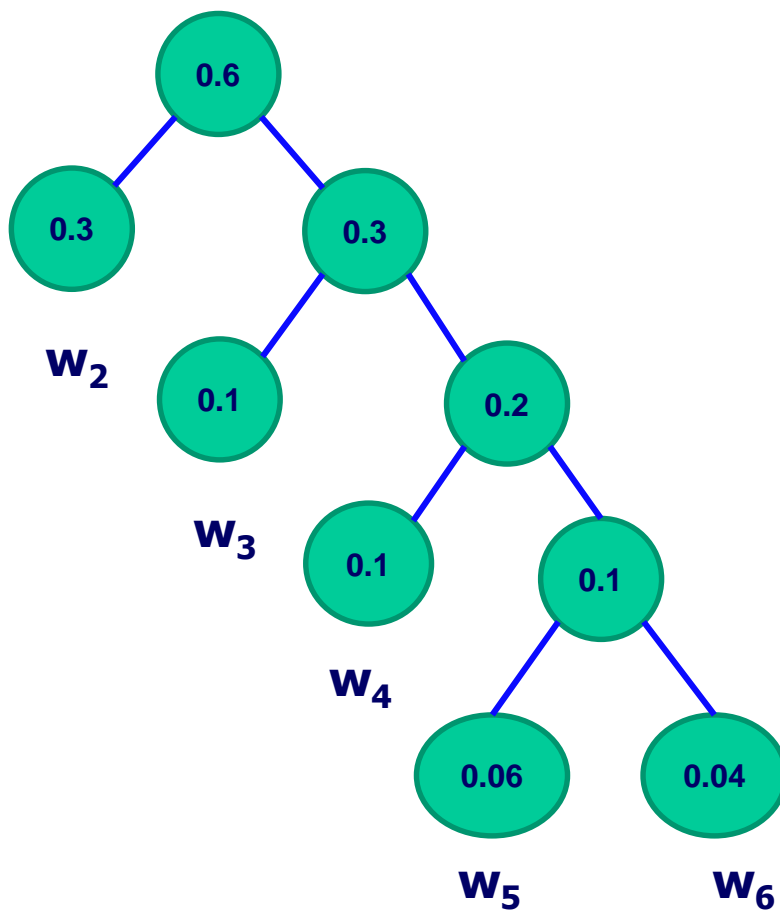


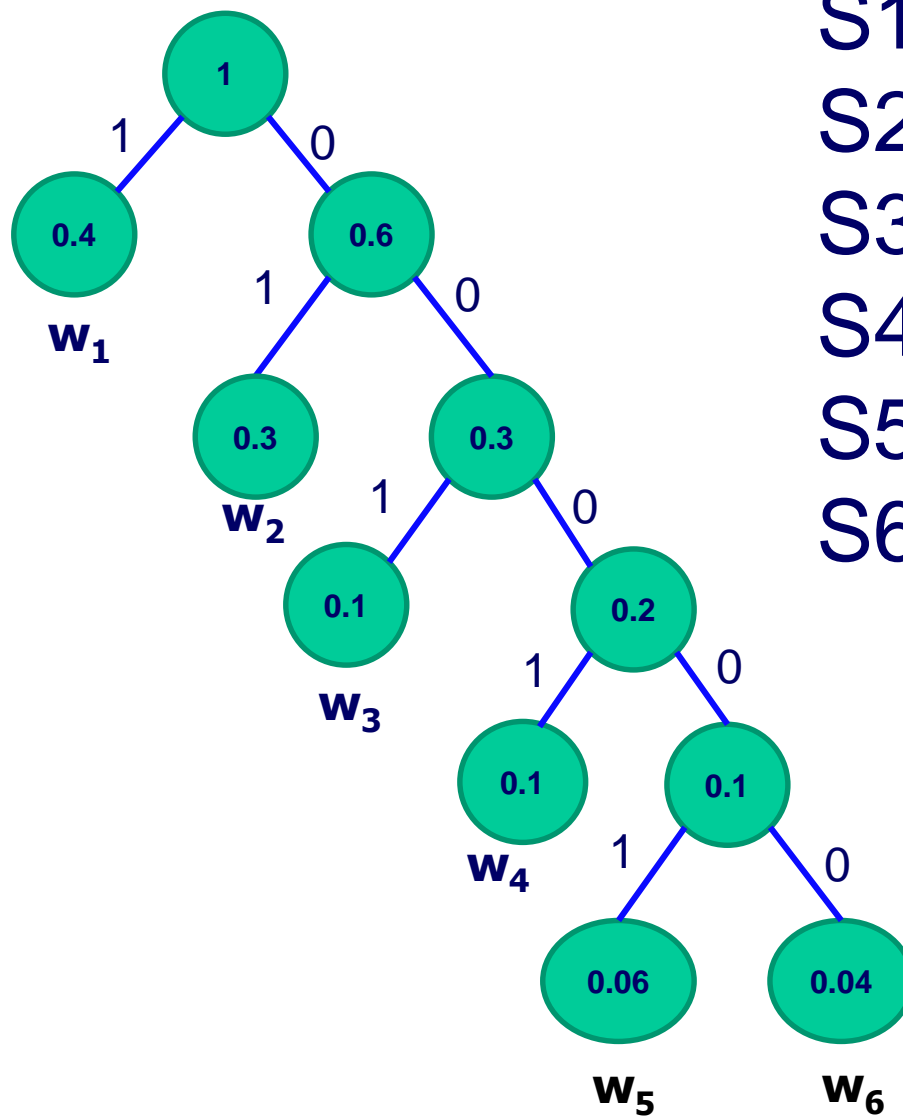






w_1





$S1=1$

$S2=01$

$S3=001$

$S4=0001$

$S5=00001$

$S6=000000$





计算编码效率：

熵：
$$H = -\sum_{k=1}^6 p_k \log_2 p_k = 2.14 \quad bit$$

平均码长：

$$\bar{N} = 0.4 + 0.3 \times 2 + 0.1 \times 3 + 0.1 \times 4 + 0.06 \times 5 + 0.04 \times 5 = 2.2 \quad bit$$

编码效率：

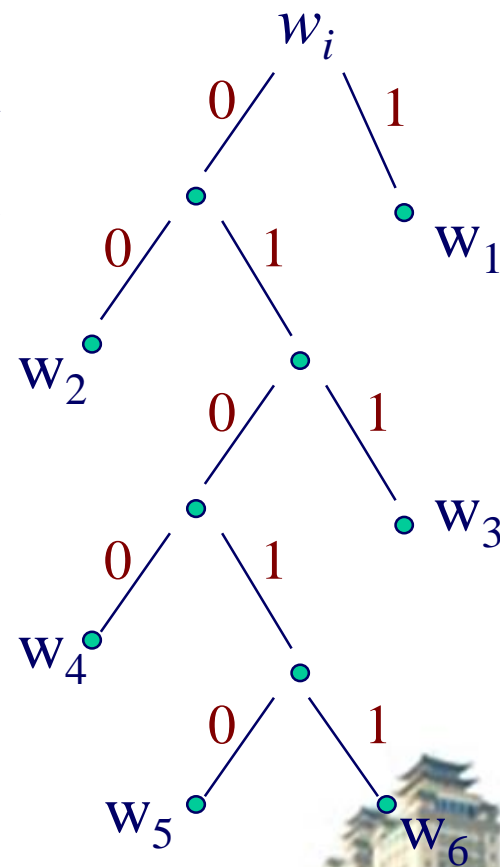
$$\eta = \frac{H}{\bar{N}} = \frac{2.14}{2.2} \times 100\% = 97.3 \%$$





为在接收端对上述编码进行解码，可以采用树形解码方法唯一地解码，每输入一位即可确定分支情况，并自动确定码字的起止位。为此需要建立右图的解码树。

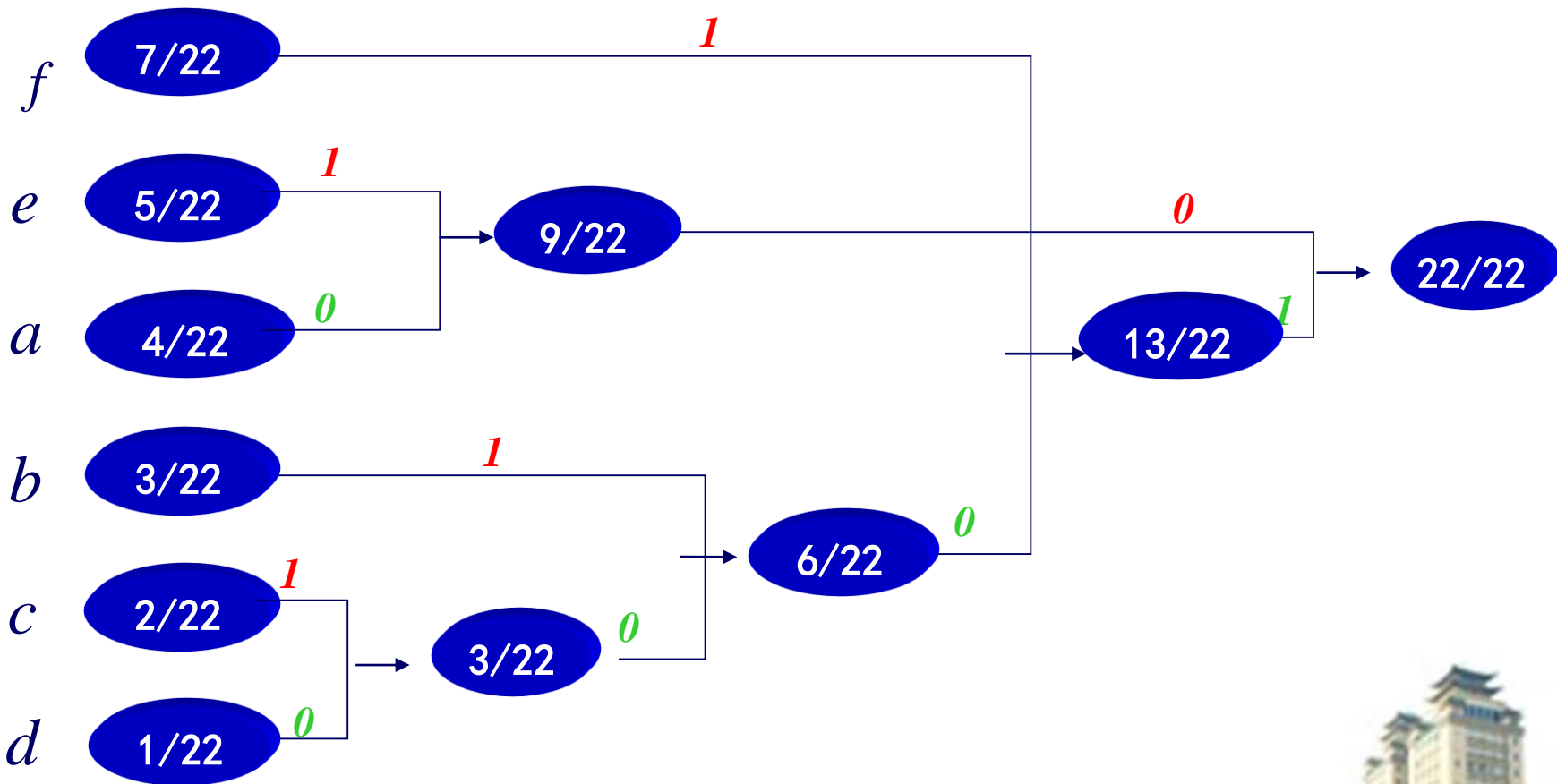
在解码时将输入的数码按树去分配，得到码字的切分和代码符号，例如，输入序列

$$1 \mid 0 \mid 1 \mid 1 \mid 1 \mid 0 \mid 0 \mid 0 \mid 1 \mid 0 \mid 1 \mid 0 \mid 1$$


从上述解码过程可以看到，虽然 *Huffman* 码不是等长码，但解码中能自动确定起止位。解码结果是唯一的。



编码举例



f=11 e=01 a=00 b=101 c=1001 d=1000

返回



概率分布为 2 的负幂次方				概率分布为均匀分布		
信源符号	出现概率	哈夫曼码字	码字长度	出现概率	哈夫曼码字	码字长度
S_0	2^{-1}	1	1	0.125	111	3
S_1	2^{-2}	01	2	0.125	110	3
S_2	2^{-3}	001	3	0.125	101	3
S_3	2^{-4}	0001	4	0.125	100	3
S_4	2^{-5}	00001	5	0.125	011	3
S_5	2^{-6}	000001	6	0.125	010	3
S_6	2^{-7}	0000001	7	0.125	001	3
S_7	2^{-7}	0000000	7	0.125	000	3
编码效率	$H=1.984\ 375$	$R=1.984\ 375$	$\eta=100\%$	$H=3$	$R=3$	$\eta=100\%$





Huffman编码的特点小结

(1) Huffman编码构造程序是明确的，但编出的码不是唯一的，其原因之一是两个概率分配码字“0”和“1”是任意选择的（大概率为“0”，小概率为“1”，或者反之）。第二原因是在排序过程中两个概率相等，谁前谁后也是随机的。这样编出的码字就不是唯一的。

(2) Huffman编码结果，码字不等长，平均码字最短，效率最高，但码字长短不一，实时硬件实现很复杂（特别是译码），而且在抗误码能力方面也比较差。

(3) Huffman编码的信源概率是2的负幂时，效率达100%，但是对等概率分布的信源，产生定长码，效率最低，因此编码效率与信源符号概率分布相关，故Huffman编码依赖于信源统计特性，编码前必须有信源这方面的先验知识，这往往限制了哈夫曼编码的应用。

(4) Huffman编码只能用近似的整数位来表示单个符号，而不是理想的小数，这也是Huffman编码无法达到最理想的压缩效果的原因。



作业：

1.有如下信源 X ,

$X =$

u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8
P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8

其中： $P_1 = 0.21$, $P_2 = 0.09$, $P_3 = 0.11$, $P_4 = 0.13$,
 $P_5 = 0.07$, $P_6 = 0.12$, $P_7 = 0.08$, $P_8 = 0.19$ 。

将该信源进行哈夫曼编码。





2. 设一幅灰度级为8（分别用 S_0 、 S_1 、 S_2 、 S_3 、 S_4 、 S_5 、 S_6 、 S_7 表示）的图像中，各灰度所对应的概率分别为0.40、0.18、0.10、0.10、0.07、0.06、0.05、0.04。现对其进行哈夫曼编码

