

## 第二章 数字化

### 一、音频数字化

#### 1. PCM 编码（最常用）

- a. **定义：**PCM 编码就是把我們听到的连续的声音（比如说话、音乐），按照固定的时间间隔进行采样，也就是一小段一小段地“截取”，然后把每一小段声音的音量（振幅）用数字的方式表示出来。这样电脑就能用一串数字来记录原来的声音，之后再把这些数字还原回声音，让我们重新听到几乎一样的效果。

b. **分步：**

步骤	名称	做了什么？	举个例子
一	采样	每隔一点时间“听”一次声音，就像不断拍声音的快照	每秒采样 44,100 次（CD 音质）
二	量化	把每次采到的声音大小（振幅）用数字表示出来	声音大小变成 0~65535 之间的数字（16 位）
三	编码	把所有的数字一个个排好，组成一串数据存进电脑或音频文件	得到一段数字音频，能还原成声音

c. **文件大小计算（未压缩）：**

参数名称	含义说明	示例值
采样率	每秒采集多少次声音数据	44,100 Hz（CD 音质）
位深度	每次采样使用多少位（二进制位）表示	16 位
声道数	声音通道数量（单声道或立体声）	2（立体声）
时长	音频的总时长（单位：秒）	10 秒
公式	文件大小 = 采样率 × 位深度 × 声道数 × 时间 ÷ 8	—
计算过程	$44100 \times 16 \times 2 \times 10 \div 8$	1,764,000 字节
转换为 MB	字节数 ÷ 1024 ÷ 1024	≈ 1.68 MB

d. **压缩：**

项目	PCM 编码（原始）	无损压缩音频	有损压缩音频
是否压缩	不压缩	压缩（不丢失信息）	压缩（丢失部分信息）
音质	非常高（原始音质）	与原始一样	可察觉变化（压缩后音质）
文件大小	最大	中等	最小
是否支持还原	是，直接播放或编辑	是，可 100%还原原始数据	否，信息永久丢失
常见格式	WAV、AIFF	FLAC、ALAC	MP3、AAC、OGG
应用场景	音乐制作、广播、存档	高音质音乐存储、备份	流媒体、网络传输、手机音乐
举个例子	专业录音室录下的声音	无损音乐播放器里的歌曲	音乐软件较低音质的音频

- **PCM 编码**是不压缩的原始声音，音质最好但文件大；

- **无损压缩**压缩后还原不丢音质；

- **有损压缩**更省空间，但会牺牲一点音质。

## 2. DSD 编码

a. **定义：**DSD 编码是一种用极高频率的 **1 位数字信号**来记录声音变化的音频编码方式，声音还原度很高，是一种高保真音频格式。

b. **DSD 与 PCM 的区别：**

**DSD 和 PCM 不一样**，它不是一次记录“声音有多大”，而是用**非常快的速度**（比如每秒 2,822,400 次）记录“声音是变大了还是变小了”。

- 每次采样只用 **1 位（0 或 1）** 来表示声音的趋势。

- 类似“跟踪声音波动的方向”而不是“测量具体数值”。

- 虽然每次只有 1 位，但因为采样非常密集，最后还原的声音也非常细腻。

项目	PCM 编码	DSD 编码
位深度	多位（如 16 位、24 位）	1 位
采样率	一般为 44.1kHz~192kHz	超高，比如 2.8224MHz（DSD64）
表示方式	表示“当前声音有多大”	表示“声音是在变大还是变小”
应用	普通音频、CD、音乐制作	SACD（超级音频 CD）、高端音响
文件大小	相对较小（有压缩时）	较大（虽然是 1 位，但采样极密）
音质风格	清晰、细节多	柔和、自然、类模拟味道

### c. 文件大小计算（未压缩）：

项目	说明	示例值（以 DSD64 为例）
采样率	每秒钟采样次数	2,822,400 Hz（DSD64）
位深度	每次采样使用的位数	1 位
声道数	声音通道数量（单声道=1，立体声=2）	2（立体声）
音频时长	声音的总时长	60 秒
公式	文件大小（字节） = 采样率 × 声道数 × 时间 ÷ 8	= 2,822,400 × 2 × 60 ÷ 8
计算结果	实际字节数	42,336,000 字节 ≈ 40.38 MB

### 常见 DSD 编码格式文件大小对比表（立体声、1 分钟音频）：

编码格式	采样率（Hz）	每分钟文件大小（字节）	约合文件大小（MB）
DSD64	2,822,400	42,336,000	≈ 40.38 MB
DSD128	5,644,800	84,672,000	≈ 80.77 MB
DSD256	11,289,600	169,344,000	≈ 161.53 MB
DSD512	22,579,200	338,688,000	≈ 323.06 MB

### d. 压缩：

压缩类型	说明	示例格式	特点	文件大小
不压缩	完全没有任何压缩，保存原始的 DSD 数据	DSD Raw 或 DSF（无压缩）	音质保持原样，但文件非常大，不做任何数据丢失	文件非常大，接近原始大小
无损压缩	压缩后可以 100% 恢复原始音质	DST（Direct Stream Transfer）	压缩后音质不丢失，适用于高保真音频	文件大小减少约 30%-50%
有损压缩	压缩后丢失部分音质，牺牲一部分细节	DSF（降低了采样率）	压缩比高，适合流媒体和存储，但音质会有所牺牲	文件大小大幅减少（可高达 90%）

## 二、图像数字化

### 1. 位图

a. 定义: 位图(Bitmap)是一种用于存储图像数据的文件格式或数据结构。在位图中, 图像被表示为一个像素网格, 每个像素有特定的颜色值。通常, 位图是通过以二维矩阵的方式排列像素来表示图像的, 每个像素对应一个数据单元。

- 位图的分辨率由图像的像素数量决定, 分辨率越高, 图像越清晰。
- 由于每个像素都需要存储信息, 位图文件通常会比较大, 特别是在高分辨率或颜色丰富的情况下。
- 当你缩放位图图像时, 图像会失去清晰度, 可能会出现锯齿状的边缘或模糊现象。
- 常见的位图格式包括 BMP、PNG、JPEG、GIF 等。

b. 文件大小计算 (未压缩):

项目	含义	示例值
图像宽度	图像水平像素的数量	1000 像素
图像高度	图像垂直像素的数量	800 像素
像素总数	图像的总像素数量 (宽度 × 高度)	$1000 \times 800 = 800,000$ 个像素
颜色深度	每个像素所占用的位数 (bit), 表示颜色信息的丰富程度	24 位
每像素字节数	每个像素占用的字节数, 计算方式为颜色深度除以 8	$24 / 8 = 3$ 字节
文件大小 (字节)	图像的文件大小, 计算方式为像素总数乘以每像素的字节数	$800,000 \times 3 = 2,400,000$ 字节

色彩表达方式:

表达方式	定义描述	示例数值	说明
个 (种) 色	表示图像可显示的颜色数量, 即总的颜色个数,	256 个 (种) 色、65,536 个 (种) 色、16,777,216 个 (种)	数值通常为 2 的 n 次方, 例如 8 位色对应 $2^8=256$

	用“个”作为单位。	色	个色。
位色	指图像的色深,即每个像素用来表示颜色信息的位数,用“位色”描述。	8 位色、16 位色、24 位色、32 位色	例如 8 位色表示每个像素用 8 位存储颜色,通常对应 256 个色。
色位	同“个(种)色”,也用于描述像素颜色的种数。	8 色位、16 色位、256 色位	注意:这种表示方法没有被广泛使用,说法不一。

### c. 压缩:

属性	不压缩 位图	无损压缩	有损压缩	说明
文件大小	最大	较小	最小	不压缩存储所有像素数据;无损压缩在不丢失信息的前提下减小体积;有损压缩则通过舍弃部分细节达到更高压缩率。
图像质量	保持原始质量,无任何损失	完全保留原始图像质量	存在不同程度的质量损失	无损压缩不影响图像质量,有损压缩在高压比下可能出现伪影和细节损失。
处理速度	数据量大,读写速度较慢	需要解压缩,但通常较快	解压和压缩处理较复杂,部分场景下较慢	无损和有损压缩都需要额外的压缩/解压过程,但有损压缩算法通常针对传输优化。
编辑灵活性	直接编辑,无重复编码问题	编辑时需解压为原始数据,后续可恢复	多次编辑和重压缩会累计损失图像质量	不压缩和无损压缩适合频繁编辑,有损压缩不适合反复处理。
存储与传输	占用大量空间,传输带宽要求高	存储和传输需求适中	占用最少存储空间,传输更高效	在网络应用和移动设备中,有损压缩常用以降低传输延时,无损压缩则用于对质量要求较高的场合。
应用场景	专业图像处理、医学影像、扫描件等	数据归档、无损要求的数字摄影和设计	数字摄影、网页图片、社交媒体、视频流媒体等	根据应用场景选择合适的存储方式:高质量需求时选用不压缩或无损压缩;对传输和存储要求较高时可采用有损压缩。

2. 矢量图

a. 定义：矢量图（Vector）是一种利用数学公式描述图形和图像的表现方式，其主要特点是用点、线、曲线和多边形等基本几何元素来构造图形，而不是像位图那样记录每一个像素的信息。这种表示方法的主要优点包括：

- 无损缩放：由于图形由数学表达式定义，放大或缩小时不会失去清晰度或出现锯齿现象。
- 较小文件体积：在图形内容较为简单的情况下，矢量图的文件通常比同尺寸的位图要小，因为它只存储形状和颜色的描述信息。
- 便于编辑：矢量图中的各个对象可以独立修改、调整形状、颜色、位置等，适合需要频繁修改和调整的设计工作。
- 矢量图常用于标志设计、图标、插图、技术绘图和排版等场合。常见的矢量图格式有 SVG、EPS、AI 和 PDF 等。

b. 文件大小计算（未压缩）：

矢量图没有文件大小的计算公式。

图形类型	描述	典型文件大小
简单 SVG（单矩形）	一个矩形，无复杂属性	约 100 字节
中等复杂 SVG（多形状）	包含多个矩形和圆，少量路径	约 1-10 KB
复杂 SVG（多路径）	包含详细路径和文本，如技术图	数十 KB 至数百 KB
高度详细矢量图	复杂插图或动画，包含大量元素	可能达 1 MB 以上

c. 压缩:

属性	不压缩	无损压缩矢量图	有损压缩矢量图	说明
文件大小	最大, 保存完整的矢量图形数据	较小, 去除冗余数据	最小, 通过丢弃部分信息进一步减小文件大小	不压缩保存完整图形, 无损压缩通过去冗余数据减小体积, 有损压缩通过丢弃细节减少体积。
图像质量	完全保留原始图像质量, 无任何损失	保持原始图像质量	存在质量损失, 特别是在高压缩比时	不压缩保持最佳质量, 无损压缩没有质量损失, 有损压缩会导致图形的细节丢失。
处理速度	读写速度较慢, 文件较大	读写速度快, 但需要解压	读写速度最快, 尤其是对于压缩较小的文件	压缩文件通常处理速度较快, 特别是有损压缩, 适合高效传输和加载。
存储与传输	存储空间大, 传输带宽需求高	节省存储空间, 传输效率更高	文件非常小, 传输和存储成本最低	无损压缩适合高质量但空间有限的需求, 有损压缩适合大规模传输, 节省带宽和存储。
编辑灵活性	便于直接编辑, 无需解压	需要解压或逐步解压编辑内容	多次编辑可能影响质量, 特别是有损压缩	不压缩和无损压缩都适合频繁编辑, 有损压缩不适合多次编辑, 会导致质量损失。
适用场景	专业设计、高质量存档、印刷等	网络应用、图形设计、文档归档、出版等	数字广告、网站图标、移动设备应用等	不压缩适用于高精度设计, 无损压缩适用于需要高质量同时节省空间的场景, 有损压缩适用于大规模传输。

3. 位图、矢量图的对比

属性	位图 (Bitmap)	矢量图 (Vector)	说明
基本定义	由像素组成, 每个像素存储颜色值	通过数学公式描述图形, 使用点、线、曲线、形状等元素	位图依赖于像素, 矢量图依赖于几何元素。
文件大小	文件较大, 特别是在高分辨率时	文件较小, 特别是在图形内容简单时	位图需要存储所有像素信息, 矢量图只存储几何描述。
图像质量	放大时会出现像素化、模糊或锯齿现象	放大或缩小时不会失真, 保持清晰度	位图的质量随分辨率变化, 矢量图无论如何缩放都保持高

			质量。
编辑灵活性	编辑时修改像素，细节丢失可能较大，特别是放大时	易于编辑，可以独立修改每个对象，如路径、形状、颜色等	位图编辑较为复杂，矢量图编辑灵活，且不会损失质量。
存储与传输	存储空间需求高，尤其是高分辨率图像，传输带宽较高	存储空间小，传输效率较高	位图文件较大，传输时需要更多带宽；矢量图文件较小，适合传输。
图像表现	适合表现复杂的细节，适合照片、艺术图像等	适合表现图标、标志、插图、简化图形等	位图表现更细腻，矢量图适合简洁、图形化的表现。
常见格式	JPEG、PNG、BMP、GIF、TIFF 等	SVG、EPS、PDF、AI 等	位图常用于照片和复杂图像，矢量图用于图标、插图和排版设计等。
适用场景	数码摄影、网页设计、打印输出等高精度图像要求的场景	标志设计、图标、插图、技术图纸、UI 设计等	位图适合照片和复杂的图像，而矢量图适合需要清晰和缩放的设计。

### 三、视频数字化

#### 1. 文件大小计算（未压缩）

项目名称	含义说明	示例值（以 1920×1080, 24fps, 24 位色, 10 秒为例）
分辨率(Resolution)	每帧图像的像素宽度 × 高度	$1920 \times 1080 = 2,073,600$ 像素/帧
色深 (Color Depth)	每个像素占用的位数，常见为 24 位（8 位 RGB 各占 8 位）	24 位 = 3 字节（每像素）
每 帧 大 小 (Frame Size)	每帧所占的总字节数 = 分辨率 × 每像素字节数	$2,073,600 \times 3 = 6,220,800$ 字节 $\approx 5.93$ MB
帧率 (Frame Rate)	每秒钟的帧数 (FPS)	24 帧/秒
每秒大小	每秒视频数据大小 = 每帧大小 × 帧率	$5.93 \text{ MB} \times 24 = 142.32 \text{ MB/秒}$



视频时长 (Duration)	视频总时长, 以秒计	10 秒
总文件大小	视频总大小 = 每秒大小 × 视频时长	$142.32 \text{ MB} \times 10 = 1,423.2 \text{ MB}$ $\approx 1.39 \text{ GB}$

## 2. 压缩

### a. 未压缩 vs 压缩视频文件大小计算方法对比表

项目	未压缩视频计算方法	压缩视频估算方法	说明
原理	逐帧逐像素记录所有颜色信息	利用视频编码器压缩冗余数据(时间冗余、空间冗余、颜色冗余)	压缩算法通过预测和参考帧等技术极大减少数据量
基本公式	分辨率 × 色深 ÷ 8 × 帧率 × 时长 (秒)	码率 (bitrate) × 时长 ÷ 8 (换算成字节)	未压缩精确计算, 压缩视频以码率为估算依据
分辨率	必须考虑, 影响帧大小	影响编码器压缩效率, 影响最终码率	高分辨率 → 高数据量 → 高码率
色深	直接影响每像素占用的字节数	通常已包含在码率中间接体现	高色深时压缩效率略低, 文件略大
帧率 (FPS)	直接乘以每帧大小	影响压缩效率, 更多帧 = 更多数据	高帧率视频需要更高的码率保持同样清晰度
时长	线性增长	线性增长	视频越长, 文件越大
关键参数: 码率 (Bitrate)	N/A (未压缩)	是计算核心: 如 4 Mbps = 4,000,000 bps = 500,000 B/s	常见码率: 720p (1-5 Mbps)、1080p (3-8 Mbps)、4K (15-50 Mbps) 等
示例计算	$1920 \times 1080 \times 24 \text{ bit} \div 8 \times 24 \text{ fps} \times 10 \text{ s}$ $\approx 1.39 \text{ GB}$	$4 \text{ Mbps} \times 10 \text{ s} \div 8$ $= 5 \text{ MB}$	同样视频未压缩需 1.39GB, 压缩后只需 5MB (假设 4 Mbps)

压缩率	无压缩	可达 100:1 或更高	依编码器和视频内容而异, 运动越复杂压缩率越低
适用场景	专业制作、无损编辑、母版存档等	网络发布、移动播放、流媒体、存储优化	未压缩质量极高但文件大, 压缩适合大多数实际应用

b. 压缩视频大小估算公式:

$$\text{文件大小 (字节)} = \text{码率 (bps)} \times \text{视频时长 (秒)} \div 8$$

示例对比 (1080p, 10 秒):

类型	示例参数	估算文件大小
未压缩	1920×1080, 24 位色, 24fps, 10 秒	≈ 1.39 GB
压缩 (中等)	码率 4 Mbps	$4,000,000 \times 10 \div 8 = 5 \text{ MB}$
压缩 (高质)	码率 8 Mbps	$8,000,000 \times 10 \div 8 = 10 \text{ MB}$

c. 视频压缩编码格式对比表:

编码格式	全称	压缩效率	文件大小	解码兼容性	画质表现	开源	适用场景
H.264 (AVC)	Advanced Video Coding	中等	中等	极高 (几乎所有设备)	优秀	否	YouTube、直播、摄像头、智能手机等
H.265 (HEVC)	High Efficiency Video Coding	高	更小	较低 (新设备支持)	更好	否	4K/8K 视频、高质量流媒体、蓝光
VP8	-	中	中等偏小	中等	尚可	是	网络视频、WebRTC、旧版 WebM
VP9	-	高	更小	好	优秀	是	YouTube

				(Chrome/ Android)			高清/4K、 WebM 格式
AV1	AOMedia Video 1	极高	最小	中等（新硬 件支持）	优秀至极高	是	新一代网络 流媒体、 YouTube 8K、 Netflix、 未来网页视 频标准
MPEG-2	Moving Picture Experts Group 2	低	最大	高	普通	否	DVD、老式 电视广播
ProRes	Apple ProRes	几乎无压 缩	非常大 （母版 级）	低（苹果生 态为主）	极高	否	专业影视后 期、素材存 储、非线性 编辑
CineForm	-	轻压缩	较大	中等	极高	是	专业剪辑、 GoPro 制 作、后期处 理
Theora	-	中	中	较低	一般	是	自由软件项 目、老式开 源网页视频

#### 四、拓展

##### 拓展 1. 令人类感到舒适的媒体分辨率

类型	常见分辨率/参数	单位	感知体验	适用场景
音频	44.1 kHz / 16 bit	采样率 / 量 化位数	CD 音质，适合大多数人	音乐、普通语音、电影
	48 kHz / 24 bit		高清音频	电影音轨、专业录音

	≥96 kHz / 24 bit		极高保真，人耳难明显区分	专业音频制作、高保真爱好者
图像	72-100 PPI（显示）	像素每英寸	屏幕常用分辨率，正常浏览无压力	网页、显示器查看图像
	300 DPI（印刷）	点每英寸	印刷品质清晰，适合近距离阅读	照片、杂志、书籍
	1080×1080（社交媒体头像）	像素	头像清晰	社交媒体的头像
视频	720p（1280×720）	像素	基础高清，移动设备观看清晰	手机视频、老电视
	1080p（1920×1080）		全高清，大多数场合最舒适	家用电视、流媒体标准
	4K（3840×2160）		超高清，近距离或大屏幕体验更佳	影院、大尺寸电视、游戏
	8K（7680×4320）		超高分辨率，人眼难完全分辨	专业制作、未来应用

## 拓展 2．注意题目要求的最终单位

各种存储单位的英文全称及其与 **bit** 的转换倍率列表，分为十进制（基于 1000）和二进制（基于 1024），此表格数据均以十进制表示：

缩写	英文全称	进制类型	换算为 bit 的倍率
bit	bit	-	1
Byte	Byte	-	8 bits
KB	Kilobyte	十进制	1 KB = 1,000 Bytes = 8,000 bits
KiB	Kibibyte	二进制	1 KiB = 1,024 Bytes = 8,192 bits
MB	Megabyte	十进制	1 MB = 1,000,000 Bytes = 8,000,000 bits
MiB	Mebibyte	二进制	1 MiB = 1,048,576 Bytes = 8,388,608 bits
GB	Gigabyte	十进制	1 GB = 1,000,000,000 Bytes = 8,000,000,000 bits
GiB	Gibibyte	二进制	1 GiB = 1,073,741,824 Bytes = 8,589,934,592 bits
TB	Terabyte	十进制	1 TB = 1,000,000,000,000 Bytes = 8,000,000,000,000 bits

TiB	Tebibyte	二进制	1 TiB = 1,099,511,627,776 Bytes = 8,796,093,022,208 bits
-----	----------	-----	--

注意：

- Byte = 8 bits 是基本换算。
- 带 i (如 KiB, MiB) 的单位是 二进制 (IEC 标准)。
- 不带 i 的 (如 KB, MB) 是 十进制 (SI 标准)。
- 考试时若无特殊说明，会混用\*B和\*iB，此时按 1024 (二进制标准) 计算。

### 拓展 3. 声道个数

声道类型	格式名称	组成说明	低 频 通 道 (LFE)	实际总声道数 (含 LFE)
Mono	1.0	单声道	0	1
Stereo	2.0	左、右声道	0	2
Stereo + LFE	2.1	左、右 + 低频	1	3
Surround	3.0	左、右 + 中央	0	3
	3.1	左、右 + 中央 + 低频	1	4
Quadraphonic	4.0	左、右 + 左后、右后	0	4
	4.1	同上 + 低频	1	5
5.0	5.0	左、右 + 中央 + 左环绕、右环绕	0	5
5.1	Dolby Digital, DTS 5.1	左、右 + 中央 + 左/右环绕 + LFE	1	6
6.1	DTS-ES, Dolby Digital EX	5.1 + 后中央声道	1	7

7.0	7.0	左、右 + 中央 + 左/ 右环绕 + 左/右后	0	7
7.1	Dolby TrueHD, DTS-HD MA	5.1 + 左/右后	1	8
9.1	扩展环绕音效	7.1 + 前高左/右或前 宽声道	1	10
11.1	Dolby Atmos, Auro 11.1	7.1 + 前宽 + 前高 + 顶部或额外中置	1	12
22.2	NHK Super Hi- Vision	超高清多声道系统(3 层 布局)	2	24

---



---