

Please visit website: <http://cxyroad.com>

关于视频你需要知道的基本概念：码率（Bitrate）、帧率（FPS）、分辨率和清晰度

=====

> 文章首发[个人博客](http://cxyroad.com/
"https://www.taoweng.site/posts/%E5%85%B3%E4%BA%8E%E8%A7
%86%E9%A2%91%E4%BD%A0%E9%9C%80%E8%A6%81%E7%9F%
A5%E9%81%93%E7%9A%84%E5%9F%BA%E6%9C%AC%E6%A6%8
2%E5%BF%B5%E7%A0%81%E7%8E%87bitrate%E5%B8%A7%E7%8
E%87fps%E5%88%86%E8%BE%A8%E7%8E%87%E5%92%8C%E6%
B8%85%E6%99%B0%E5%BA%A6")

前言

— —

想通过一篇文章的科普，让你在使用视频的时候知道如何选择合适的视频参数。

- * 一般涉及到的视频参数就是：封装格式、编码格式、码率、帧率和分辨率。
- * 对用户表现的体感就是：流畅度和清晰度。
- * 开发者需要关心的是：内存、CPU 和 GPU 的消耗、体积大小和带宽。

本篇文章会尝试把这些概率都联系起来，对视频相关的概念有整体的认知。

结论先行

用户视角：

- * 清晰度 = 码率 / 分辨率
* 流畅度 = 帧率

开发者视角：

- * 影响内存的：主要是分辨率

- * 影响 CPU 的：码率和编码格式
- * 影响 GPU 的：分辨率和编码格式。
- * 影响体积大小和带宽：码率

封装格式

****就是把视频数据和音频数据打包成一个文件的规范。*** 一个完整的视频文件，包括音频、视频和基础元信息，我们常见的视频文件如 ****mp4、mov、flv、avi、rmvb**** 等视频文件，就是一个容器的封装，里面包含了音频和视频两部分。

封装格式是不会影响视频的画质，封装成什么格式就看在使用的时候解码器是否支持这个封装格式即可。

> 封装，也叫多路复用(mux)。封装的目的一般为了在一个文件（流）中能同时存储视频（video）、音频（audio）、字幕（subtitle）等内容——这也正是“复用”的含义所在（分时复用）。封装还有另一个作用是在网络环境下确保数据的可靠快速传输。

编码格式

我们知道，其实视频就是一帧一帧的图片。计算一下，一部 25 帧每秒，90 分钟，分辨率为 1024*768，24 位(rgb 每个 8 位)真彩色的视频，没有经过压缩，大小为：

* 1Byte（字节） = 8bit（位）
* 一帧大小 = 1024 * 768 * 24 = 18874368（bit） = 2359296（Byte）
* 总帧数 = 90 * 60 * 25 = 135000
* 总大小 = 一帧大小 * 总帧数 = 2359296 * 135000 = 318504960000（Byte） = 303750（MB） ≈ 296（GB）

从上面的计算可以看出，我们储存一部 90 分钟没压缩的电影需要 296GB 的。所以我们需要对视频进行压缩，这种视频压缩技术就是我们所说的编码。

视频编码方式：H.26X（H.261、H.262、...、H.264（目前最常用）、H.265）

音频编码方式：MP3、AAC 等

通过视频压缩算法，减少了视频文件的大小。压缩比越大，解压缩还原后播放的视频越失真，这是因为压缩的同时不可避免的丢失了视频中原来图像的数据信息。

> 注意编码格式要跟封装格式做区分，两者没有任何关系。

一般我们只需要 `H.264` 和 `H.265` 即可，H.264 是使用最广泛，除了 H.265 之外压缩率最高的，也就是说在相同码率下除了 H.265 之外 体积最小的。

而 H.265 相比于 H.264 相同码率下体积还降低了 **50%** 。

那是不是就可以无脑使用 H.265 了呢？先说说使用 H.265 有什么问题，我认为最大的两点就是：

1. **计算复杂性**。H.265 的编码和解码过程比之前的标准（如 H.264）更复杂，压缩率更高，解码需要更多的计算资源。这意味着在一些低端设备上可能会面临性能问题。
2. **兼容性**。尽管 H.265 已经得到广泛支持，但仍然存在一些老旧设备不支持 H.265 的硬件解码，这就会导致播放器只能使用软解，会增加 CPU 的消耗。

> 硬件解码：顾名思义就是通过硬件进行解码的计算，通常是 GPU。软件解码：顾名思义就是通过软件计算解码，软件计算主要依靠的是 CPU。CPU 的计算能力相比 GPU 来说弱很多，就会导致 CPU 占用率高，导致设备发热发烫。

所以在设备相对较好的情况下，无脑使用 H.265 是没有问题的，如果对于低端设备使用 H.265 可能会带来性能问题，导致视频播放卡顿，需要用空间来换时间了。

视频分辨率

视频分辨率跟我们平时说的图像分辨率差不多，主要是指 **单位英寸中所包含的像素点数。** 可以用长 x * 宽的形式表示比如 1280 * 720 的分辨率，也可以以像素点总数的形式表示，比如 200 万像素(1920 * 1080)。

常见的分辨率：

- * 720 P(1280 * 720)
- * 1080 P(1920 * 1080)
- * 2K(2560 * 1440)
- * 4K(4096 * 2160)

- * 视频的分辨率与像素不可分，比如一个视频的分辨率为 1280 * 720,就代表了这个视频的水平方向有 1280 个像素，垂直方向有 720 个像素。
- * 另外需要知道的：

- + 720P 是指视频有 720 行像素，P 是 Progressive 逐行扫描。
- + 2K 是指视频像素的总列数。
- + MP 是像素总数，指像素的行数(P)与列数(K)乘积的结果（百万像素） -
- + 1080P 分辨率为 1920 * 1080，总像素 2073600，通常将 1080P 叫做 200 万像素分辨率
- * 通常视频在同样视窗的情况下，分辨率越高，所包含的像素就越多，视频画面就越细腻、越清晰。

> 另外需要注意分辨率越大在解析到内存之后占用的内存也会更大，这个原理跟图片的分辨率对内存的影响是一致的。对于内存优化，通常降低分辨率是一个有效的方式。

视频帧率

- * 帧率的单位是：FPS 是指每秒帧数（Frames Per Second）简写：P，“如 30 fps 和 30 p 指的都是每秒钟播放 30 张图片的意思”。
- * 高低帧率：低帧率会伴随模糊、卡顿、镜头移动时不顺畅等问题，但也不是越高越好，会造成眩晕的感觉。
- * 不同帧率的视频在不同设备上播放：
 - ① 若一个播放设备最高支持 60fps，在此设备上播放 120fps 的影片，则播放设备会每隔一张删除 120fps 的影片，被删除的图片成为无效帧。

这就导致高帧率的影片在低帧率的设备上播放时，会降低播放流畅度。

② 若一个播放设备最高支持 120fps，在此设备上播放 60fps 的影片，则播放设备会每将每张图片复制一张，以填补空缺的帧。但是效果和 60fps 的设备上播放一样，对播放流畅度没有提升。

帧率是直接影响流畅度的指标，视频一般 24FPS/25FPS 已经足够，低于 24 将会感到卡顿。

视频码率（视频比特率）

> 非常非常重要的一个参数。

码率就是数据传输时单位时间(1s)传送的数据位数，一般我们用的单位是 kbps 即千位每秒。

码率与体积的关系：

* 在相同的时间下，码率越大，体积就越大。
* **存储体积 = 码率 × 时间**。

码率与画质的关系：

* **同样分辨率下，视频文件的码率越大，压缩比就越小，画面质量就越高。码率越高，精度就越高，处理出来的文件就越接近原始文件，图像质量越好，画质越清晰，要求播放设备的解码能力也越高。**

* **编码模式**

码率取样率，单位时间内取样率越大，精度就越高，处理出来的文件就越接近原始文件，但是文件体积与取样率是成正比的，所以几乎所有的编码格式重视的都是如何用最低的码率达到最少的失真，围绕这个核心衍生出来 cbr（固定码率）、vbr（可变码率）、abr（平均码率）；

CBR: constant bitrate(固定码率)

* 以恒定比特率方式进行编码，有运动发生时，由于码率恒定，只能通过增大 QP 来减少码字大小，图像质量变差，当场景静止时，图像质量又变好，因此图像质量不稳定。

* 这种算法优先考虑码率(带宽)。适合在流式播放中应用。

* CBR 编码的缺点在于编码内容的质量不稳定,容易产生马赛克。因为对于某些较复杂的图像比较难压缩，所以 CBR 流的某些部分质量就比其他部分差。

VBR: variable bitrate(动态码率)

- * 比特分配根据图像内容的复杂度进行。如果图像细节较丰富或者含有大量的运动，则给其分配大一点的码流，若图像比较平坦，就给其分配较少的码流，这样既保证了质量，又兼顾带宽限制。
- * 这种算法适合图像内容变化幅度较大的情况；
- * 适合的应用场景是媒体存储，不适合网络传输；

QVBR (Quality Variable Bit rate) 质量码流

- * QVBR 在 VBR 的基础上，增加了自适应动态质量控制功能。编码器会根据图像质量目标和帧率等因素自适应调节码率，以确保图像质量。这种编码方式减少了码率的小部分波动对图像质量的影响，进一步提高了图像的清晰度和稳定性。
- * 优点：能够在保证画面质量的前提下减少码流的波动
- * 缺点：相比 VBR 需要更高的码率带宽比。

ABR: average bitrate(平均码率，是 VBR 的一种插值参数)

- * 是在 CBR 和 VBR 两者之间的一种权衡，即设定一段时间的平均码率，在此时间内，对简单，静态的图像分配低于平均码率的码率，对于复杂的，大量运动的图像分配高于平均码率的码流；
- * 适合网络传输；

CVBR: Constrained VariableBit Rate(有限码流)

- * 是 VBR 的一种改进方法，对应的 Maximum bitRate 恒定或者 Average BitRate 恒定。兼顾了 CBR 和 VBR 的优点：在图像内容静止时，节省带宽，有运动发生时，利用前期节省的带宽来尽可能的提高图像质量，达到同时兼顾带宽和图像质量的目的。这种方法通常会让用户输入最大码率和最小码率，静止时，码率稳定在最小码率，运动时，码率大于最小码率，但是又不超过最大码率。

联系

--

帧率相关

- * 对视频画质的影响：画面流畅度，不卡顿。
- * 与码率的联系：帧率越大，画面越流畅；每秒展示的画面（图片越多），需要的解码能力越强；若解码能力跟不上，则编码器就必须加大对单帧画面的压缩比，也就是通过降低画质来承载足够多的帧数，画面容易出现跳跃，不连贯，即丢帧；反之亦然。

码率

- * 影响体积，与体积成正比：码率越大，体积越大；码率越小，体积越小。

分辨率：

- * 影响图像大小，与图像大小成正比：分辨率越高，图像越大；分辨率越低，图像越小
- * 如果限定一个码率，比如 800kbps，那么分辨率越高就会让编码器越“为难”，可以想象，它必须拆东墙补西墙，通过减少色彩信息或者引入马赛克这种“鱼目混珠”的手段来承载足够多的像素点。所以，同样的是 2G 的一个电影文件，1080p 画质的版本可能不如 720p 画质的版本看起来更清晰。

清晰度：

- * 码率一定的情况下，分辨率与清晰度成反比关系：分辨率越高，图像越不清晰，分辨率越低，图像越清晰。
- * 在分辨率一定的情况下，码率与清晰度成正比关系，码率越高，图像越清晰；码率越低，图像越不清晰。

总结

--

没有完美的算法，无非是时间和空间的平衡，二者不可兼得。

作为一个开发者，我们需要在用户的终端性能、网络、成本等各方面权衡，在不同的场景使用不同的参数的视频，可以提升用户的使用体验，降低公司的成本。

在了解各个参数之后，在对视频做优化的时候就会游刃有余：

* 当需要优化视频传输成本，降低视频的体积，最有效的就是降低码率，这样可能会降低清晰度，也可从视频编码格式入手，能否将 H.264 升级成 H.265.

* 当终端设备内存受限，页面需要播放多个视频导致内存不够，首要考虑降低视频分辨率从而降低内存消耗。

参考文章：

* [github.com/CharonChui/...](http://cxyroad.com/"https://github.com/CharonChui/AndroidNote/blob/master/VideoDevelopment/%E8%A7%86%E9%A2%91%E5%B0%81%E8%A3%85%E6%A0%BC%E5%BC%8F/%E8%A7%86%E9%A2%91%E5%B0%81%E8%A3%85%E6%A0%BC%E5%BC%8F.md")

* [blog.csdn.net/qq_42024067...](http://cxyroad.com/"https://blog.csdn.net/qq_42024067/article/details/107954595")

原文链接: <https://juejin.cn/post/7362196417278066703>