1.视频的基础知识

1.1 I、P、B帧和GOP

视频压缩中,每帧代表一幅静止的图像。而在实际压缩时,会采取各种算法减少数据的容量,比如H264 压缩编码等,其中IPB就是最常见的。

简单地说,I帧是关键帧,属于帧内压缩。就是和AVI的压缩是一样的。 P是向前搜索的意思。B是双向搜索。他们都是基于I帧来压缩数据。

I帧表示关键帧,可以理解为这一帧画面的信息完整保留;解码时只需要本帧数据就可以完成图像的还原(因为包含完整画面)

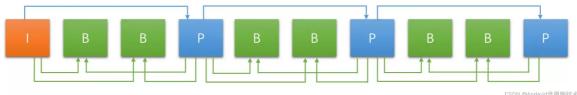
P帧表示的是这一帧跟之前的一个关键帧(或P帧)的差别,解码时需要用之前缓存的画面叠加上本帧定义的差别,生成最终画面。(也就是差别帧,P帧没有完整画面数据,只有与前一帧的画面差别的数据)

B帧是双向差别帧,也就是B帧记录的是本帧与前后帧的差别(具体比较复杂,有4种情况),换言之,要解码B帧,不仅要取得之前的缓存画面,还要解码之后的画面,通过前后画面的与本帧数据的叠加取得最终的画面。

由此可见,如果一个视频中采用B帧压缩比较多,则压缩率会高很多,但是解码时CPU会比较累,因为 总是需要双向搜索才能解析出来

	帧类型	预测方式	参考帧	优缺点
l帧	帧内编码帧	只进行帧内预测	无	自身能独立完成编码解码。 压缩率小。
P帧	前向编码帧	可以进行帧间预测和 帧内预测	参考前面已编码的I帧 和P帧	压缩率比I帧高。 必须要参考帧能够正确编解码。
B帧	双向编码帧	可以进行帧间预测和 帧内预测	参考前面或者后面已 编码的I帧和P帧	压缩率最好。 需要缓存帧,延时高。 RTC场最不适合站台音视频技术

三种帧的示例图如下所示。例如,从左向右,第一个 B 帧参考第一个 I 帧和第一个 P 帧,第一个 P 帧只参考第一个 I 帧(箭头是从参考帧指向编码帧)。



CSDN @Android音视频技术

1.2 IDR帧

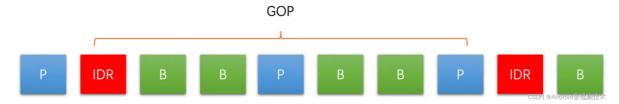
在关键帧I帧中,我们还会定义一个特殊的I帧,称之为IDR帧,也叫立即刷新帧。这个帧是干嘛用的呢?

在上述所讲的B帧和P帧,它们是会去参考其他的帧来进行解析的。假如前面有一帧I帧出错了,那么后面参考它的P帧也出错了,接着参考这个I帧或P帧的其他P帧和B帧,也都相继出错。错误可能就会不断传递下去。为了防止这种情况,H264 编码标准中规定,IDR 帧之后的帧不能再参考 IDR 帧之前的帧。也就是说我们在编码的时候可以编码一个IDR帧,这样这个帧之前的若有错误的帧,对这个IDR帧之后的数据的解析就不会有影响了。

实际上,在真实场景中,我们大概率会用IDR帧,而比较少用普通的I帧。不过如果你只用普通的I帧,标准也是支持的,但是不推荐。

1.3 GOP

GOP是代表从一个 IDR 帧开始到下一个 IDR 帧的前一帧为止的中间包含的所有IDR 帧、普通 I 帧、P 帧和 B 帧,也就是一个图像组。**GOP的大小就是两个IDR帧之间的间隔,也就关键帧间隔。**



GOP 越大,编码的 I 帧就会越少。相比而言,P 帧、B 帧的压缩率更高,因此整个视频的编码效率就会越高。但是 GOP 太大,也会导致 IDR 帧距离太大,点播场景时进行视频的 seek 操作就会不方便。并且,在 RTC 和直播场景中,可能会因为网络原因导致丢包而引起接收端的丢帧,大的 GOP 最终可能导致参考帧丢失而出现解码错误。

因此GOP不是越大越好,也不是越小越好,要根据实际场景来决定。

1.4 DTS & PTS

视频和音频的同步过程是一个你等我赶的过程,快了则等待,慢了就加快速度。这就需要一个量来判断(和选择基准比较),到底是播放的快了还是慢了,或者正以同步的速度播放。在视音频流中的包中都含有DTS和PTS,就是这样的量。

DTS(Decoding Time Stamp):解码时间戳;告诉解码器packet的解码顺序,也就是先解哪个包,才能解哪个包,需要按照顺序来,才能正确地一个一个解析出来

PTS(Presentation Time Stamp):显示时间戳;指得是packet解析出来后,要显示的时候是想显示哪个再显示哪个,指明的数据的显示播放的顺序

在这里可能有人会有个疑问,这两个东西为什么会不同?一个一个解析,然后一个一个显示,这样的话 不就两个应该是一样的吗?

在这里可能有人会有个疑问,这两个东西为什么会不同?一个一个解析,然后一个一个显示,这样的话不就两个应该是一样的吗?

其实,之所以两者会不同,主要也是跟上面提到的I、B、P帧有关,尤其是B帧,因为B帧要解析需要参考前面的或者后面的帧率,那么就代表着B帧没法按照显示顺序来解析,可能需要先解析后面的某个I帧,P帧,才能解析当前的这个B帧。这也是导致两者时间顺序不同的主要原因。

例如,假如一个视频序列,要这样显示IBBP,但是需要在B帧之前得到P帧的信息,因此帧可能以这样的顺序来存储IPBB,这样其解码顺序和显示的顺序就不同了,这也是DTS和PTS同时存在的原因。DTS指示解码顺序,PTS指示显示顺序。所以流中可以是这样的:

CSDN @Android音视频技术