

/\*

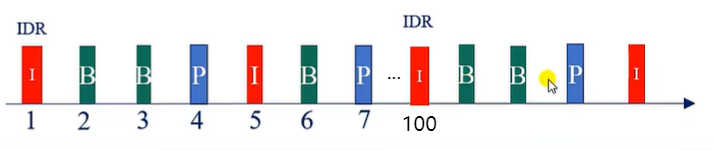
I帧：只进行空间上的压缩，不进行时间上的压缩。(帧内压缩编码)只和本帧内的图像像素有关。与前后帧没有关系。就会导致有许多重复冗余信息。压缩比不大。但压缩后得到的图像质量仍然较好。解码时直接解码即可，不需要参考前后其它帧图像。只对本帧图像压缩。



在编码的时候，已经划分好了宏块。那么我们可以通过123宏块推测第4个宏块。那么我们对第4个宏块就不需要进行编码了，只需要对123宏块进行编码即可。

I帧分为普通I帧和IDR帧：

IDR帧就是关键帧，很重要。是特殊的I帧。IDR帧相当于分水岭。例如：



假设一个200帧的码流，第一个帧为IDR帧，第100个帧也是IDR帧，则从第101帧开始，之后的帧 不能再参考100帧前面的哪些帧了。所以说IDR帧就相当于一个分水岭。可以把码流成为一段一段的。

两个IDR帧内部，是可以互相参考的。

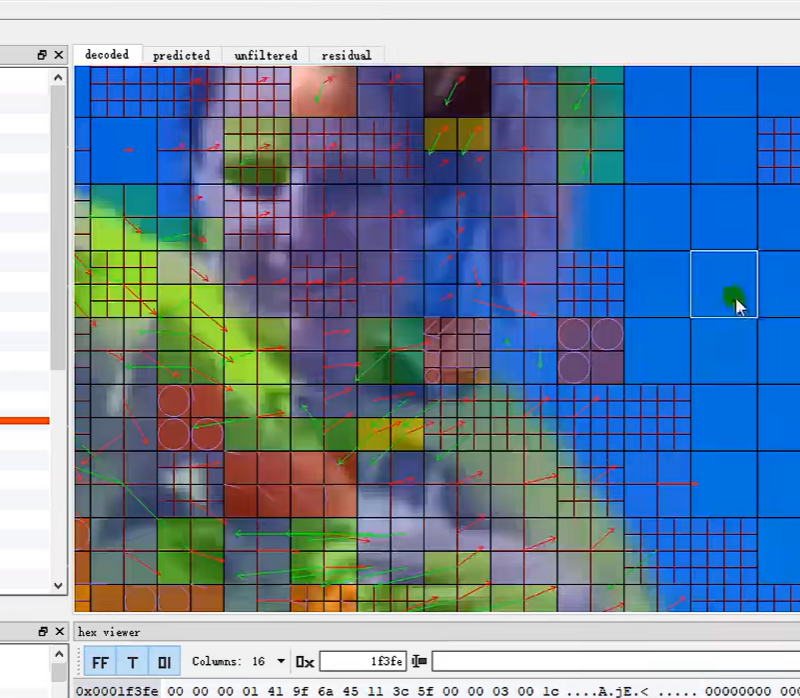
\*/

/\*

P帧属于参考帧，需要参考前一帧的I图像或者P图像来预测。是前向参考帧，P帧压缩率比价高，占用的空间较小。常用于运动估计。

我们在使用软件分析帧的时候会看到P和B帧的图上会有一些小箭头，P帧没有：

这些小箭头就表示物体的运动矢量。也就是平移的大小。



编码是以宏块为单位进行编码的。例如我们在编码P帧的时候，并不需要将整个宏块进行编码，而是以前面帧对应的宏块为参考，编码他们之间的差值即可（也就是编码残差即可）。

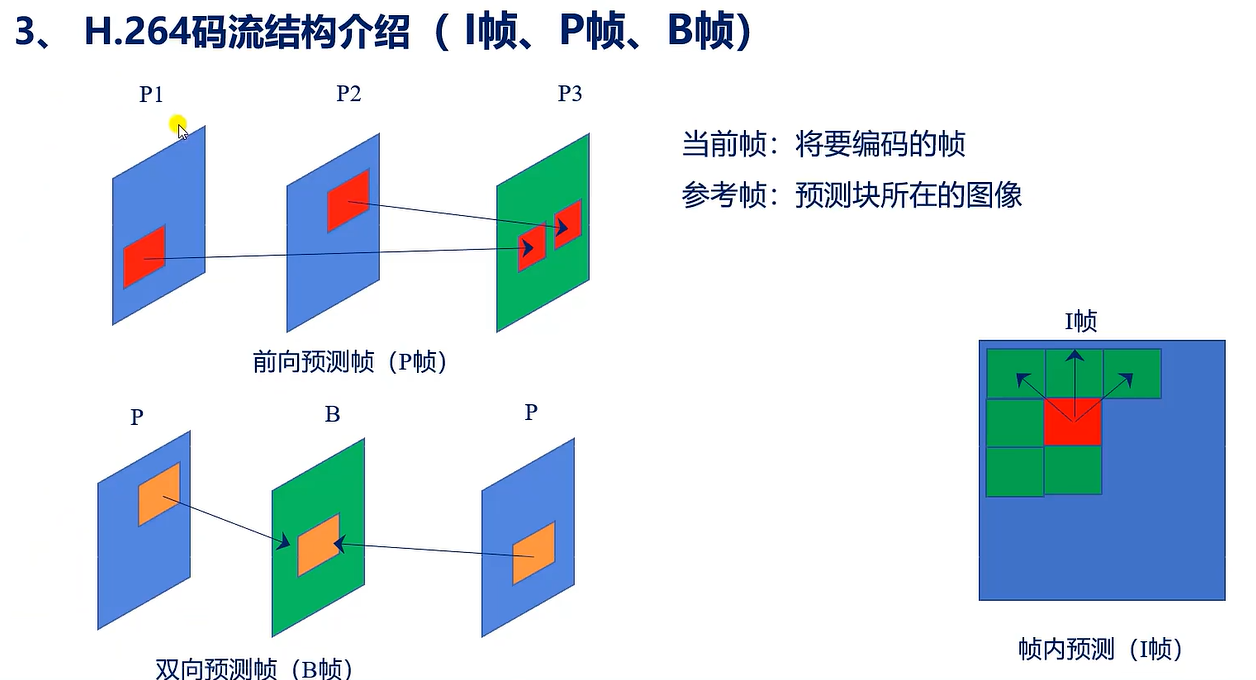
也就是说，编码对应块和参考块的残差就行。

查找相似块的过程就称为运动搜索/运动估计。我们用残差和之前的参考块就可以推算出当前的块。这个过程就叫做运动补偿。这种编码方式就叫做帧间编码。帧间压缩剔除了时间和空间上的冗余。节省了大量的空间。

\*/

/\*

B帧也称为双向预测图像，在压缩编码的时候需要参考前面和后面两个方向的图像进行编码和预测。通常采用帧间和帧内混合编码的方式。解码的时候也就需要前后的帧才能解码。



由于B帧需要根据前后帧来解码，所以会有帧延时，不适用于实时视频传输。也不适用于直播等。但适用于电影文件这种类型中。

\*/