

第23 讲: HEVC

- 简介
- 输入视频格式
- HEVC的新功能
- 编码树单元 (CTU) 和编码单元 (CU)
- 预测块 (PB)
- 较大的变换: 变换块 (TB)
- 帧内预测中的新模式
- 熵编码
- 并行处理工具
- 格式范围扩展
- 档次, 级别, 层级
- 屏幕内容编码
- HEVC编码器设计

HEVC – 简介

- 联合标准：由ITU VCEG和ISO MPEG开发的ISO标准ISO / IEC 23008-2 **MPEG-H第2部分**和国际电联**ITU-T H.265**标准。
- 2010年1月，VCEG和MPEG发布了一项关于视频压缩技术的联合建议征集（CfP），并于2010年4月在视频编码联合协作小组（JCT-VC）首次会议上对各种所提建议进行评估。
- **版本1**（2013年4月13日）包含Main，Main 10和Main 静止图像档次。
- **版本2**（2014年10月29日）添加了21个范围扩展档次，两个可伸缩扩展档次和一个多视图扩展档次。
- **版本3**（2015年4月29日）添加了3D主要档次。
- 在相同的视频质量下，与H.264 / MPEG-4 AVC相比，**HEVC使压缩比增加一倍**。
- **支持8K UHD**，分辨率高达 8192×4320 。

输入视频格式

- 从**QVGA (320x240)** 到**4320p (8192x4320)** 的逐行扫描帧率和显示分辨率，包括**HD**和**UHD** (4K和8K) 视频。
- HEVC被设计用于**逐行扫描视频**，因为未来格式是逐行的，因此并没有专门为隔行视频添加编码工具。在以往标准中为隔行视频设定的编码工具，如MBAFF和PAFF，将不再支持。
- HEVC通过发送**元数据** (SEI消息) 来**支持隔行视频**，这些元数据告诉解码器如何编码隔行扫描的视频。隔行视频可以由以下方式编码：
 - 将每帧编码为单独的图像或
 - 将每场编码为单独的图像。
 - HEVC允许使用序列自适应帧场 (SAFF) 在帧编码和场编码之间进行改变，这允许在序列级改变编码模式。

数据格式和比特流语法

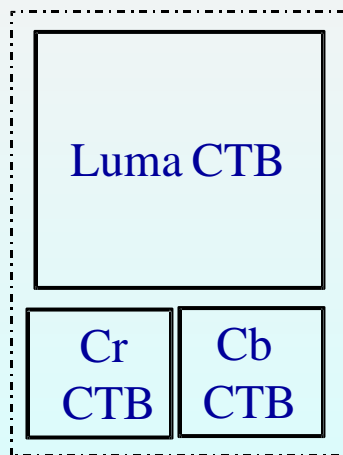
- **闭合GOP**：所有图像都可解码，而不参考其他GOP的图像。在H.264 / AVC中，一致的比特流必须以定义闭合GOP的IDR图像开头。
- **开放GOP**：开放GOP允许来自一个GOP的B帧参考相邻GOP中的I帧或P帧。HEVC引入了一种独特的NAL单元类型，用于发送**清除随机访问（CRA）图像**，从而实现了开放的GOP。也就是说，符合HEVC的比特流可以以IDR或CRA图像开头。在开放的GOP中，一些图像可以依赖于CRA图像前面的图像。**断链访问（BLA）图像**是用于发送比特流拼接点的特殊CRA图像。
- 总之，HEVC高级语法为随机访问和拼接提供了新的功能；并在IDR，CRA和BLA图像处提供随机访问。
- HEVC引入**并行Tile**以进行**并行编码/解码**，并利用相关的片(Slice)以减少开销。

对比 H.264/AVC HEVC的新特征

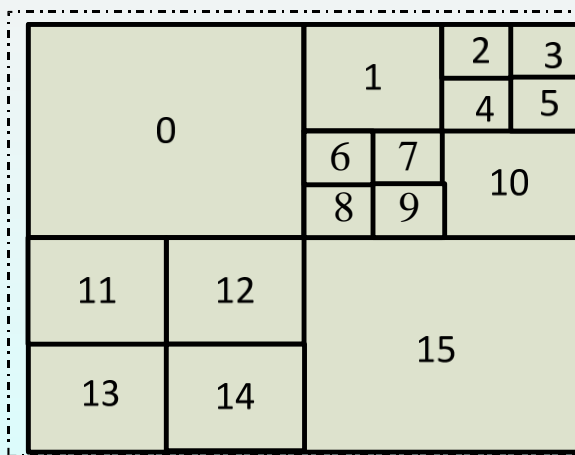
- 支持更**高分辨率**的视频
- **编码树单元**：更大更灵活的块尺寸
- 预测块：用于运动补偿的**非对称分区**
- 变换块：**较大的变换**
- **更多帧内预测**模式
- 较大的插值滤波器
- **高级运动矢量预测**和合并模式
- 高吞吐量的CABAC
- 改进的去块滤波器和**采样自适应偏移 (SAO)**
- 高级**并行编码/解码**工具

编码树单元 (CTU)

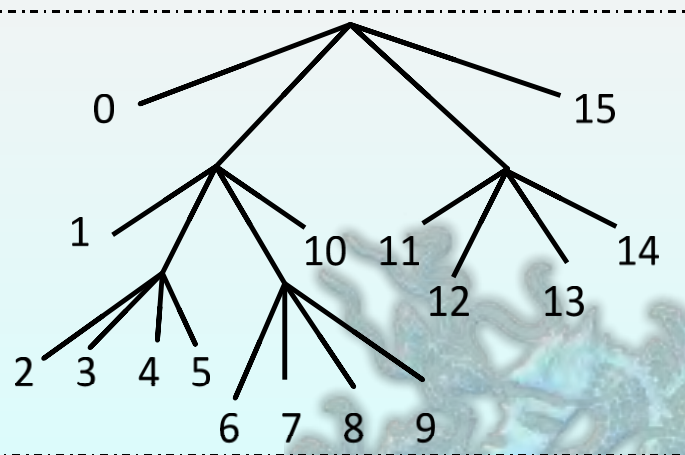
- HEVC首先将图像的每个片划分成**编码树单元 (CTU)**，它们分别是属于Y，Cr和Cb的 $N \times N$ 的矩形块，其中 $N = 16, 32$ 或 64 。它是序列级参数，且其每个组件被称为编码树块 (CTB)。
- 经典的 16×16 像素MB被**替换为CTU**，可以更好地将图像分割成可变大小的块，以提高编码效率。
- CTU可以是具有**帧内和帧间编码块的组合**。



编码树单元



亮度 CTB 64×64

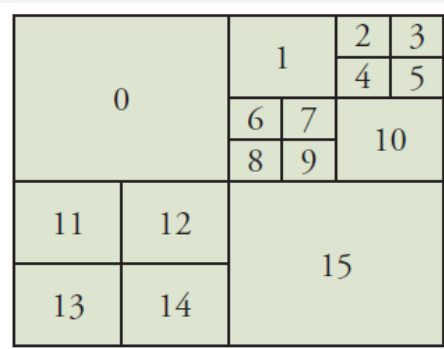


分区的二叉树表示

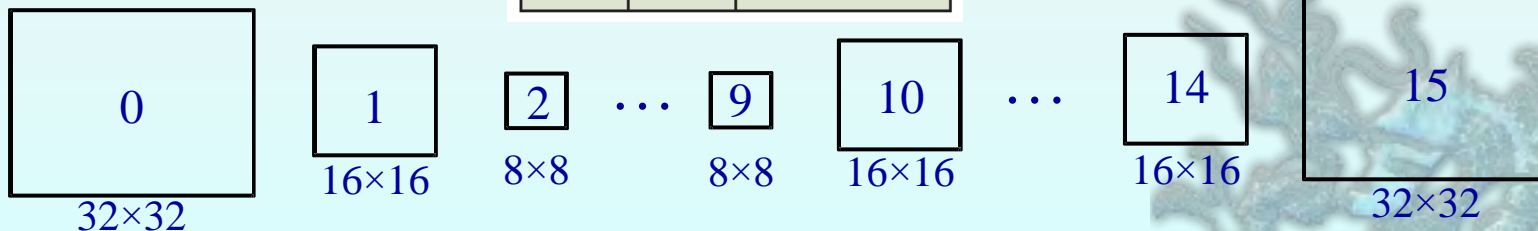
编码单元 (CU)

- 每个**CTB**可以被分成不同的**多个编码块 (CB)**。HEVC支持与CTB尺寸相同的CB尺寸，可小至 8×8 。
- 预测类型在CU语法中指定。也就是说，每个CU可以以不同的编码模式（帧内或帧间）进行编码。
- **编码单元 (CU)** 由**亮度CB**和相关联的**色度 (Cr和Cb) CB**组成。

编码树块
 64×64 亮度



编码块
(亮度)



预测块 (PB)

- 每个编码块可以进一步分为**预测块 (PB)**。每个PB可以具有不同的运动矢量或预测类型。
- 帧内编码CU可以是一个PB，只有 **8×8 CU**可以进一步**划分为四个 4×4 正方形预测块**。
- 只要满足**最短边至少为4像素**的条件，帧间编码的CU可分为**正方形或非对称分区**。为了避免太多MV，不允许 4×4 PB，而 4×8 和 8×4 PB只能用于单向运动补偿。



帧间编码CU的可能预测块

变换块 (TB)

- HEVC支持 **4×4 , 8×8 , 16×16 或 32×32** 变换块。
 - 两个 4×4 变换（**帧内**是基于**IDST**，**帧间**是基于**IDCT**），而 8×8 , 16×16 或 32×32 变换块只基于IDCT。
- **整数变换**避免了编码器 - 解码器的不匹配和漂移。
- 在帧间编码的CU中，**TB边界不需要与PB边界对齐**。也就是说，可以对来自多个PB的残差执行单个变换用于帧间预测CU。
- 每个CU的变换块大小通过**残差二叉树**的方法进行指示。
- 较大块的压缩效率提高了5-10%。
- 与H.264 / AVC相比，**计算复杂度更高**。
 - 每系数8倍多的计算
 - 16倍大的转置存储

帧内预测的新模式

- HEVC支持**35个帧内预测模式**，而AVC中则为10个。

- **模式 0: 平面**

- 水平和垂直平均

- **模式 1: DC**

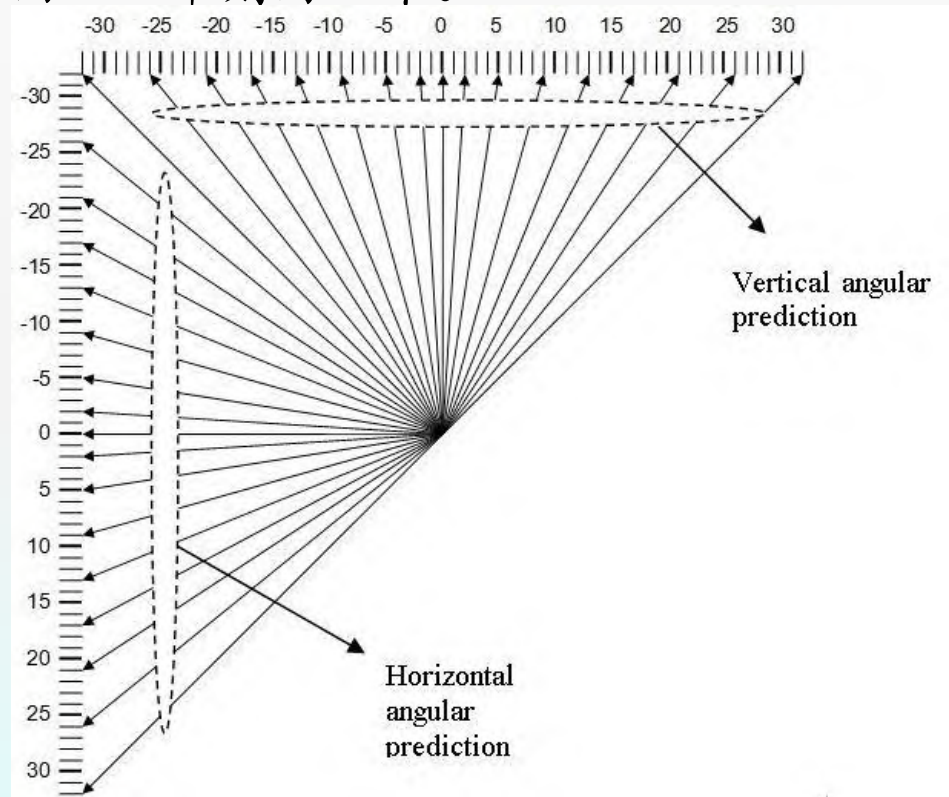
- 常数，是所有参考像素的平均值。

- **模式 2-34是角度模式**

- 基于角度的参考像素插值：双线性插值，1/32像素精度

- 帧内预测分区**遵循TB四叉树**。

- 应用预处理和后处理来减轻帧内预测损伤。



加权预测与MV精度

- **加权预测工具**对于淡化或照明变化场景编码时特别有用。
- **乘性加权因子**和**附加偏移**被应用于运动补偿预测。
- HEVC支持**单向和双向PB的加权预测**，其中权重在片头中显式传输；而不是像AVC那样的隐式加权预测。
- 与AVC类似，HEVC有**两个参考列表L0和L1**，每个参考列表可以保存16个参考，但唯一图像的最大数量为8。
 - 编码器可以选择将相同的图像多次添加到列表中，以便能够以不同的权重预测相同的图像。

运动矢量精度与范围

- 亮度运动矢量 (MV) 可以达到**1/4像素精度**。
- HEVC分别使用可分离的**8抽头**和**7抽头内插滤波器**，用于1/2和1/4样本亮度位置
 - 与使用6抽头和2抽头双线性滤波器的H.264 / AVC不同。

FILTER COEFFICIENTS FOR LUMA FRACTIONAL SAMPLE INTERPOLATION

index	-3	-2	-1	0	1	2	3	4
hfilter[i]	-1	4	-11	40	40	-11	4	1
qfilter[i]	-1	4	-10	58	17	-5	1	

- **色度 MV**可以达到**1/8像素精度**，并使用4抽头滤波器。
- 所有PB需要在所有边界进行扩展，为滤波器提供所需数量的边界样本。
- HEVC为水平和垂直MV定义了一个带符号的16比特范围，允许MV具有-32768至32767的范围。给定1/4像素精度，HEVC允许MV范围为-8192至8191.75亮度样本。
 - 相比之下，H.264 / AVC允许-2048至2047.75亮度像素的水平MV范围和-512至511.75亮度像素的垂直MV范围。

运动矢量编码

- HEVC允许进行**高级运动矢量预测 (AMVP)** 和合并模式的两种MV预测/编码模式。合并和AMVP都会构造一个候选MV列表，然后使用在比特流中编码的索引来选择其中的一个。编码器为帧间预测块 (PB) 决定使用哪个。
- 每个MV执行一次AMVP模式。它使用来自参考图像的数据和来自相邻预测块的数据。它对增量MV进行编码。
- 当当前PB的MV与其相邻块之一相同并且允许从相邻PB继承MV时，使用**合并模式**。与H.264 / AVC中的“跳过”和“直接”运动模式相比，HEVC“合并”模式提供了两个改进：
 - 第一个改进是HEVC使用索引信息来选择几个可用候选者之一。
 - 第二个改进是HEVC使用参考图像列表和参考图像索引中的信息

自适应环路滤波器



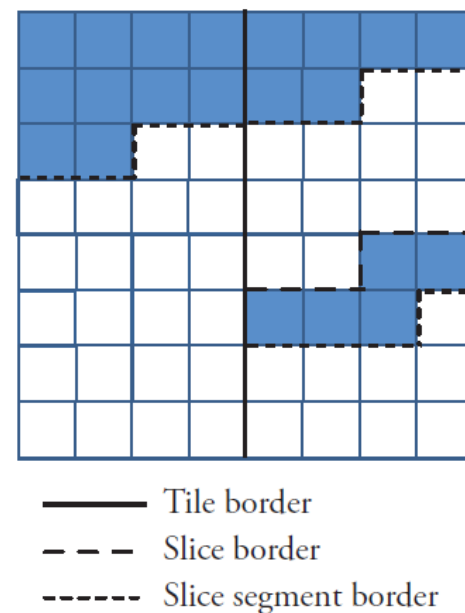
- **去块滤波器**：基于滤波器强度处理边界两边的两个或三个像素，与 H.264 / AVC 基本类似，不同之处：
 - 较大块：仅 8×8 块
 - 所有垂直边界首先被去块滤波，然后是所有的水平边界。
 - 因此，可以对图像的每个 8 像素列执行垂直边缘滤波
 - 仅当边界一侧的 CB 被帧内编码时，色度才被去块滤波。
- **样本自适应偏移 (SAO)**：解码器端基于像素的非线性后处理，以抑制因振铃现象导致的伪边缘
 - 根据像素位置处的局部梯度使用查表法将偏移量加到选定像素。

熵编码

- HEVC使用与H.264 / AVC基本相似的上下文自适应二进制算术编码 (CABAC) 算法。 **CABAC**是HEVC中允许的**唯一熵编码方法**。
- HEVC中的CABAC和变换系数的熵编码设计可实现比H.264 / AVC更高的吞吐量，同时相对于简单的扩展，可对较大的变换块尺寸实现更高的压缩效率。例如，
 - 上下文编码项的数量已经减少了8倍
 - CABAC旁路模式在其设计方面得到了改进，以提高吞吐量。
 - 编码数据之间的依赖关系进行了调整，以进一步提高吞吐量。
- HEVC中的上下文建模也得到了改进，因此与H.264 / AVC相比，CABAC可以**更好地选择那些提高编码效率的上下文**。

并行编码/解码工具

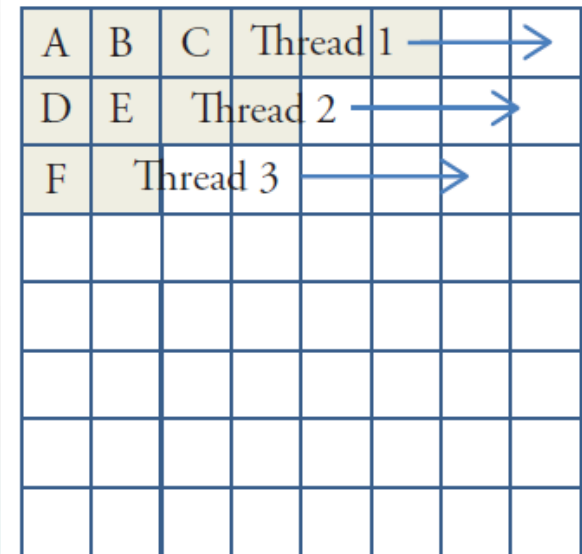
- **Tiles: 独立可解码**的矩形区域，至少应为 256×64 个亮度的像素。
- 在**子图级别并行化**，不需要线程之间的同步。
同时便于感兴趣区域（ROI）解码。
- 可以在共享公共头的**单个片内包含多个tile**，或者相反地，**tile可以包含多个片**。



由 8×8 CTU组成的图像被分为两个由垂直实线表示的tile。 请注意，**tile始终与CTU边界对齐**。 **第一个tile包含单个片**，其被分成两个分割片，其边界用虚线表示。 由阴影CTU标记的第一个分割片是独立的分割片，另一个被配置为依赖分割片。 **第二个tile被分成两个片**，每个片具有一个独立的和一个相关的分割片。

并行编码/解码工具 (cont'd)

- **波前并行处理 (WPP)**
- WPP通过将片分割成CTU行来实现子片级并行。每行可以由独立的线程处理;然而,线程之间将存在两个CTU处理延迟。也就是说,第二行的处理只有在第一行的前两个CTU (A和B) 已经完成之后才能开始,即C和D可以并行处理,并且第三行的处理可以在第二行的两个CTU (D和E) 已经完成之后开始,等等。
- WPP不允许与tile一起使用,并且与tile相比通常提供更好的压缩效率,因为帧内预测或运动矢量的预测不能跨tile边界执行。



附加模式

- 无损视频编码

- 旁路变换，量化和环路滤波器
- 在HEVC中， $QP = 4$ 表示旁路量化。

- 变换跳过

- 仅旁路变换，执行量化和环路滤波。该功能旨在提高屏幕内容编码的编码效率，与无损编码无关。

- Intra PCM (I_PCM)

- 处理原始像素。PCM应用于整个CU，并且可能不会大于 32×32 。亮度和色度可以具有单独的位深。

格式范围扩展

- 范围扩展中增加了**附加编码工具选项**，以支持每像素**位深超过10比特**，4: 2: 2和4: 4: 4色度格式以及4: 0: 0（单色）格式。
- **扩展精度处理**：使用扩展动态范围进行帧间预测插值和逆变换。
- **高精度加权预测**：增加加权预测的精度，提高高位深下淡化视频场景的编码效率。
- **跨分量预测**：使用色度/亮度分量之间的预测来提高编码效率。对于YCbCr 4: 4: 4视频，比特率最高可降低7%，RGB视频由于分量之间的相关性更高，其比特率降低可达26%。
- **禁用帧内平滑**：禁止使用在帧内预测中通常所用的邻域滤波处理。
- 修正变换跳过模式处理
- **CABAC旁路对齐**：允许在高吞吐量4: 4: 4 16 帧内档次中支持旁路解码之前将数据按照字节边界对齐。

档次, 级别和层级

- **档次, 级别和层级**定义了标准实现的一致性。
 - **档次**指定了解码器必须支持的HEVC工具的子集。
 - **级别**定义了对参数的约束, 如最大分辨率, 帧速率等。
 - **层级**定义了对最大比特率和编码图像缓冲区 (CPB) 的限制。
- HEVC标准的版本1, 2013年7月完成, 定义了**3个档次: 主要, 主要10, 主要静止图像**; **两个层级: 主要层和高层**。
 - 主要10档次可以提高视频质量并更好地支持广泛应用于UHDTV系统的ITU Rec. 2020色彩空间。它能够提供更高的动态范围和色彩保真度, 而不会造成比特率的损失。
 - 主要静止图像是主要档次的一个子集。对于4: 2: 0的8比特图像, 它的PSNR比JPEG 2000提高约20%, 比JPEG提高约60%。
- HEVC标准的版本2包括**范围扩展, 可伸缩扩展和多视图扩展**, 于2014年7月完成。版本2添加了21个扩展档次, 两个可伸缩扩展档次和一个多视图档次。

屏幕内容编码

- **屏幕内容编码**由MPEG定义为包含较大比例渲染（移动或静态）图形，文本或动画的视频，而不是摄像机捕获的视频场景，或附加在摄像机捕获的视频场景之上。
- 这样的混合视频越来越多地用于诸如桌面共享，**具有辅助材料的视频会议**（例如展示幻灯片），**网页滑动**，**视频游戏**以及广播的应用中。
- 提出用于屏幕内容编码的**新编码工具**：
 - 具有扩展搜索范围和各种分区方式、模式切换和位移矢量编码的**帧内块复制**；
 - 调色板模式具有**编码调色板本身和索引**的各个方面；
 - 基于字典编码的**字符串匹配**；
 - **交叉分量预测**和**自适应颜色变换**；
 - 对现有工具的其他小修改（如去块滤波等）。

HEVC 编码器设计

- 选择一个平台
 - 桌面CPU, CPU + GPU
 - 移动 CPU
 - DSP, FPGA
 - ASIC
- 如何在某些设计约束下做出最佳的编码器决策：
 - 选择一种运动估计方法
 - 为每个CU（相关的）选择CTB分区和编码模式
- 编码器设计约束：
 - 必须产生符合某档次的比特流
 - 必须满足所需的压缩比（比特率）要求
 - 必须满足所需的复杂度（功率）和吞吐量要求（实时？）
- 需要算法/架构创新

HEVC 编码器模式选择

- 确定变量

- CU 尺寸
- 帧内或帧间 CU
- 运动矢量
- 帧内 CU: 帧内模式选择
- 帧间 CU: PB 分区和 PB 模式

- 率失真代价函数:

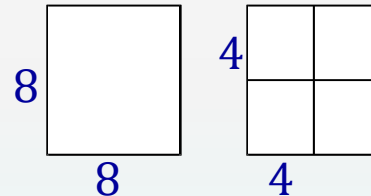
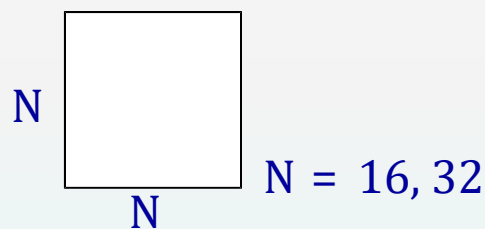
$$C = D + \lambda R$$

- D = 原始视频与压缩/未压缩视频之间的失真，通常为平方误差和 (SSE) 或绝对误差和 (SAD) 或绝对变换差值和 (SATD)
- R = 符号的速率，熵或压缩视频的总比特数
- λ = 拉格朗日乘子，用于表示 R 与 D 的相对重要性。

- 全搜索 RDO vs. 快速 RDO

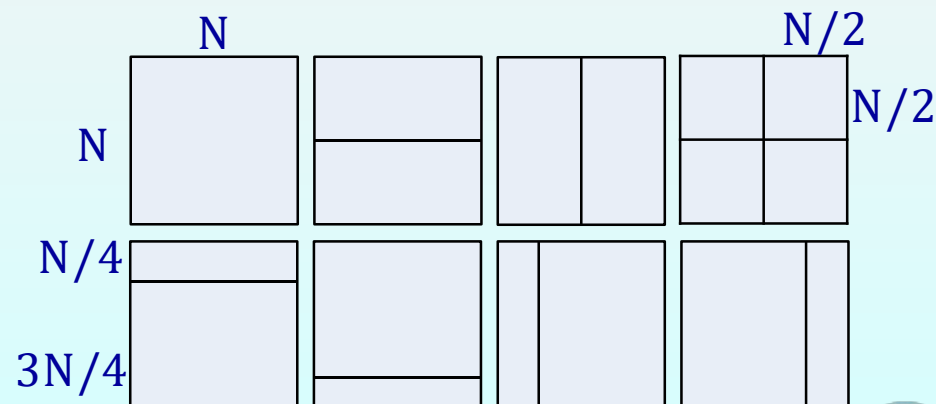
HEVC 编码器模式选择 (cont'd)

- HEVC-HM实现**全搜索RDO** (每个CTU有太多选项)
- 用于 **64×64 CTU的CU选项** - $64 \times 64, 32 \times 32, 16 \times 16, 8 \times 8$
- **帧内CU**可以是 $32 \times 32, 16 \times 16, 8 \times 8$, 并作为一个PB, 只有 8×8 的帧内CU可以继续分区。



每个PB有35个模式
 4×35 不同的编码

- 用于帧间CU的PB分区

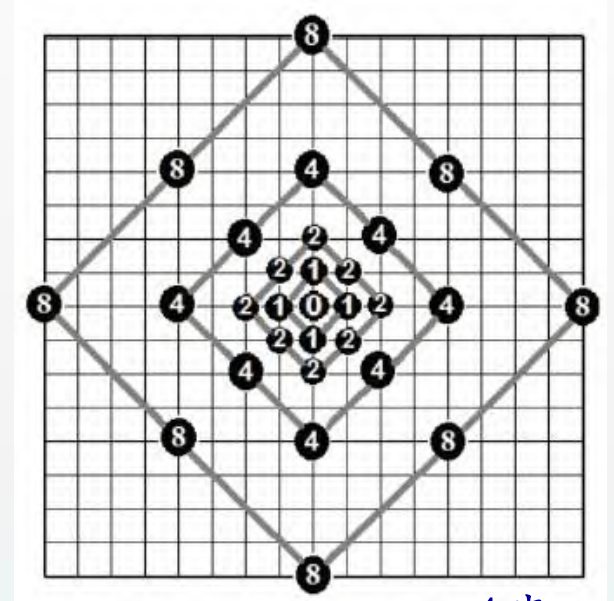


不允许 4×4 ,
仅允许单向 8×4 和
 4×8

HEVC 编码器运动估计

- 整数MV搜索

- 搜索中心是MV预测
- 菱形搜索 (搜索范围= ± 64 , 7步)
- 如果搜索中心最佳MV > 5像素, 执行光栅扫描搜索 (5像素步长)
- 在步骤2或3搜索得到的最佳MV周围进行菱形搜索。如果找到新的最佳MV, 重复此步骤。



4步

- 半像素 MV 搜索

- 搜索围绕最佳整数MV的8个点

- 四分之一像素MV搜索

- 搜索围绕最佳半像素MV的8个点

- 搜索合并/跳过的候选位置

习题

- 8.1
- 8.2
- 8.3