**开放媒体联盟 (Alliance for Open Media)**

文档编号: CWG-B078[o]\_v1

AV1 和 libaom 工具描述

日期:

2021年10月4日

状态:

输出文档

目的:

信息

作者:

赵鑫, 刘杉, Adrian Grange, Andrey Norkin

电子邮件:

xinzzhao@tencent.com, shanl@tencent.com, agrange@google.com, anorkin@netflix.com

来源:

腾讯, 谷歌, Netflix

摘要

本文档提供了 libaom 中主要编码功能的描述，libaom 是 AV1 标准规范的软件实现。本文档描述了规范的解码过程和一些关键的编码算法。

1 引言 3

2 缩写 4

3 工具描述 5

目录

摘要 ...... 1

1 引言 ...... 3

2 缩写 ...... 4

3 工具描述 ...... 4

3.1 块划分 ...... 4

3.1.1 编码块划分 ...... 4

3.1.2 变换块划分 ...... 5

3.2 帧内预测 ...... 7

3.2.1 方向性帧内预测 ...... 7

3.2.2 非方向性帧内预测 ...... 8

3.2.3 递归帧内预测 ...... 9

3.2.4 基于亮度的色度预测 (Chroma from Luma, CfL) ...... 10

3.2.5 帧内预测模式信号传递 ...... 10

3.3 帧间预测 ...... 11

3.3.1 参考帧系统 ...... 11

3.3.2 空间运动矢量预测 ...... 12

3.3.3 时间运动矢量预测 ...... 14

3.3.4 动态运动矢量预测 ...... 15

3.3.5 帧间预测模式信号传递 ...... 16

3.3.6 平移运动补偿 ...... 18

3.3.7 扭曲运动补偿 ...... 21

3.3.8 重叠块运动补偿 ...... 23

3.3.9 复合帧间预测 ...... 24

3.3.10 复合帧间-帧内预测 ...... 25

3.4 变换编码 ...... 26

3.4.1 核心变换 ...... 26

3.4.2 变换选择与信号传递 ...... 27

3.5 量化 ...... 29

3.6 熵编码 ...... 31

3.6.1 多符号算术编码引擎 ...... 31

3.6.2 系数编码 ...... 31

3.7 环路滤波和后处理 ...... 32

3.7.1 去块滤波器 ...... 32

3.7.2 约束方向增强滤波器 (CDEF) ...... 33

3.7.3 环路恢复滤波器 (Loop Restoration Filter) ...... 35

3.7.4 帧超分辨率 ...... 36

3.7.5 胶片颗粒合成 ...... 36

3.8 屏幕内容编码 ...... 37

3.8.1 帧内块复制 (Intra Block Copy) ...... 37

3.8.2 调色板模式 (Palette Mode) ...... 39

3.8.3 编码器内容类型检测 ...... 40

4 参考文献 ...... 40

# 1 引言

开放媒体联盟视频编码标准 AV1 的框架基于混合视频编码结构，包含几个主要功能模块，如预测、变换、量化、熵编码和环路滤波。每个功能模块使用某种类型的视频编码技术处理输入数据，其输出被传递到另一个功能模块或作为视频编解码器的最终输出。这些功能模块按照特定的设计连接并协同工作，以实现显著的数据压缩。AV1 参考编解码器 libaom [1] 中包含的功能模块总结如下，并在第 3 节中详细描述。

## 块划分

* 编码块划分 [2]
* 变换块划分 [2]

## 帧内预测

* 方向性帧内预测 [2]
* 非方向性帧内预测 [2]
* 递归帧内预测 [2]
* 基于亮度的色度预测 (CfL) [3]
* 帧内预测模式信号传递

## 帧间预测

* 参考帧系统 [2]
* 空间运动矢量预测
* 时间运动矢量预测
* 动态运动矢量预测 [4]
* 帧间预测模式信号传递
* 平移运动补偿
* 扭曲运动补偿 [5]
* 重叠块运动补偿 [6]
* 复合帧间预测
* 复合帧间-帧内预测

## 变换编码

* 核心变换
* 变换选择与信号传递

## 量化

## 熵编码

* 多符号算术编码引擎 [8]
* 系数编码 [9]

## 环路滤波和后处理

* 去块滤波器
* 约束方向增强滤波器 (CDEF) [10]
* 环路恢复滤波器 [11]
* 帧超分辨率
* 胶片颗粒合成 [12]

## 屏幕内容编码

* 帧内块复制 [13]
* 调色板模式
* 编码器内容类型检测

本文档中描述的每个功能模块的编码特性都包含在 AV1 编解码器的 libaom [1] 实现中。

在本文档中，语法元素使用 Courier 字体表示，例如语法元素 base\_q\_idx。

# 2 缩写

本文档中使用的缩写如下(中文为机翻)：

ARF 交替参考帧 (Alternate Reference Frame)

AV1 开放媒体联盟视频编码标准 (AOMedia Video 1)

BV 块矢量 (Block Vector)

CDEF 约束定向增强滤波器 (Constrained Directional Enhancement Filter)

CfL 基于亮度的色度预测 (Chroma from Luma)

DRL 动态参考列表 (Dynamic Reference List)

EOB 块结束 (End of Block)

FIR 有限脉冲响应 (Finite Impulse Response)

IntraBC 帧内块复制 (Intra Block Copy)

LR 环路重建 (Loop Restoration)

LRU 环路重建滤波器 (Loop Restoration Unit)

MV 运动矢量 (Motion Vector)

OBMC 重叠块运动补偿 (Overlapped Block Motion Compensation)

SGF 自导向滤波器 (Self-Guided Filter)

# 3 工具描述

## 3.1 块划分

### 3.1.1 编码块划分

AV1 中使用不同大小的编码块。最大的编码块称为超级块 (Superblock)，大小为 128x128 或 64x64，默认大小为 128x128。大小在序列头中信号传递。最小的编码块大小为 4x4。

超级块可以进一步划分为更小的编码块。划分策略在比特流中信号传递。除了无划分模式 (PARTITION\_NONE) 外，还有多达九种支持的划分模式（见图 1）。这些包括三种 4 划分模式：PARTITION\_SPLIT、PARTITION\_VERT\_4 和 PARTITION\_HORZ\_4；四种 3 划分（T 形）模式：PARTITION\_HORZ\_A、PARTITION\_HORZ\_BPARTITION\_VERT\_A 和 PARTITION\_VERT\_B；以及两种 2 划分模式：PARTITION\_HORZ 和 PARTITION\_VERT。

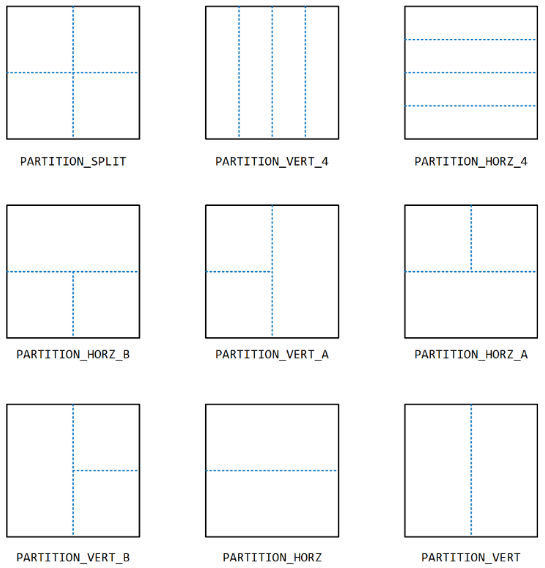


Fig.1 编码块划分模式

在所有的划分模式中，只有PARTITION\_SPLIT允许递归划分，即子划分块可以进一步划分。对于所有其他划分模式，子划分块不能进一步划分。此外，对于8x8或128x128块尺寸，不允许使用PARTITION\_VERT\_4和PARTITION\_HORZ\_4模式，而对于8x8块尺寸，不允许使用T形划分模式。

### 3.1.2 变换块划分

无论是内部还是帧间编码块，都可以进一步划分成多个变换块，其划分深度最多可以达到两个级别。变换块的尺寸由变换划分模式和编码块尺寸决定。从当前深度的变换尺寸到下一个深度的变换尺寸的映射如表1所示。

Table 1: 变换划分尺寸设置

|  |  |  |  |
| --- | --- | --- | --- |
| Current depth | | Next depth | |
| Enumerator | Transform size | Enumerator | Transform size |
| TX\_4X4 | 4×4 | TX\_4X4 | 4×4 |
| TX\_8X8 | 8×8 | TX\_4X4 | 4×4 |
| TX\_16X16 | 16×16 | TX\_8X8 | 8×8 |
| TX\_32X32 | 32×32 | TX\_16X16 | 16×16 |
| TX\_64X64 | 64×64 | TX\_32X32 | 32×32 |
| TX\_4X8 | 4×8 | TX\_4X4 | 4×4 |
| TX\_8X4 | 8×4 | TX\_4X4 | 4×4 |
| TX\_8X16 | 8×16 | TX\_8X8 | 8×8 |
| TX\_16X8 | 16×8 | TX\_8X8 | 8×8 |
| TX\_16X32 | 16×32 | TX\_16X16 | 16×16 |
| TX\_32X16 | 32×16 | TX\_16X16 | 16×16 |
| TX\_32X64 | 32×64 | TX\_32X32 | 32×32 |
| TX\_64X32 | 64×32 | TX\_32X32 | 32×32 |
| TX\_4X16 | 4×16 | TX\_4X8 | 4×8 |
| TX\_16X4 | 16×4 | TX\_8X4 | 8×4 |
| TX\_8X32 | 8×32 | TX\_8X16 | 8×16 |
| TX\_32X8 | 32×8 | TX\_16X8 | 16×8 |
| TX\_16X64 | 16×64 | TX\_16X32 | 16×32 |
| TX\_64X16 | 64×16 | TX\_32X16 | 32×16 |

表1中的变换类型可以概括为：对于1:1方块，下一层的变换划分将创建四个1:1方块变换块；对于1:2或2:1非方块，下一层的变换划分将创建两个1:1方块变换块；对于1:4或4:1非方块，下一层的变换划分将分别创建两个1:2或2:1变换块。

对于帧内编码块的亮度颜色分量，所有的变换块必须具有相同的大小。例如，对于一个32×16编码块，一级变换划分将创建两个16x16变换块，二级变换划分将创建八个8x8变换子块。内部编码块变换划分的示例如图2所示。变换块的编码顺序通过箭头表示。

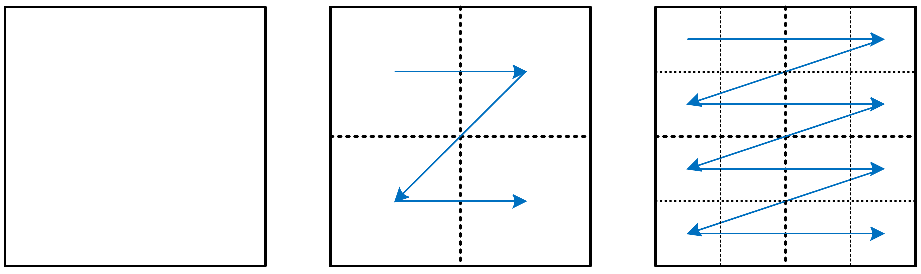


Fig.2 帧内编码块的变换划分示例

对于帧间编码块的亮度颜色分量，在Level 1变换划分之后，每个子块可以在Level 2中独立划分。因此，帧间编码块可以具有可变的变换块大小，如图3所示。

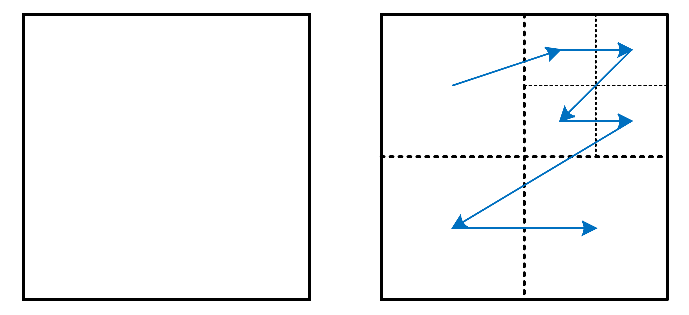


Fig.3 帧间编码块的变换划分示例

对于色度颜色分量，变换块的大小与相应的色度编码块大小相同，除非色度编码块的宽度或高度大于32，在这种情况下，色度变换块的宽度或高度设置为32。

如果编码块的大小小于或等于64x64，上述变换块划分方案从编码块大小开始执行。然而，如果编码块的宽度W或高度H大于64（以亮度样本为单位），则首先将其隐式划分为多个最小(W, 64)x最小(H, 64)子块，此步骤不需要额外地标记。变换块划分方案从每个最小(W, 64)x最小(H, 64)子块开始适用，并且划分会显式地信号化。例如，对于一个128x64编码块，残差块首先隐式划分为两个64x64子块，然后每个64x64子块可以进一步划分成更小的变换块，划分模式显式地记录在码流中。

## 3.2 帧内预测

### 3.2.1 方向帧内预测

方向帧内预测使用一组边缘方向来建模局部纹理。有八种标准方向帧内预测模式，每种模式都有一组角度偏移值（angle delta），索引值在-3到+3之间，标准角度值位于0位置。通过将角度偏移值添加到标准帧内角度以导出预测方向。总共有56种方向帧内预测模式。图4展示了八种标准模式（实箭头）和在D67\_PRED标准模式周围的一组角度偏移值（虚箭头）的例子，当角度偏移值不为零时。

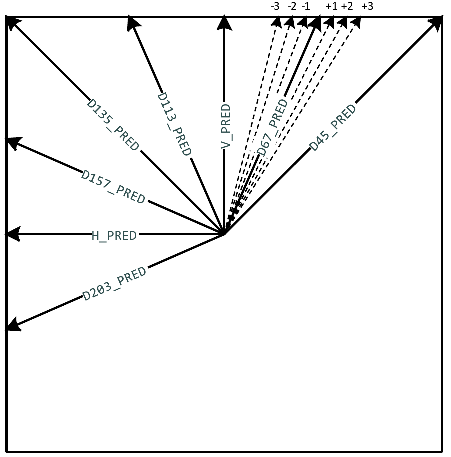


图4：方向帧内预测模式。实箭头表示标准帧内预测模式，虚箭头表示具有非零角度偏移的帧内预测模式

标准模式和相关的角度偏移索引都会被信号化，标准模式索引在相关角度偏移索引之前被信号化。对于较小的块大小，即4x4、4x8和8x4，延长帧内预测角度精度的额外编码增益通常是有限的；因此，在这种情况下，仅使用标准模式，没有角度偏移被应用或信号化。

3.2.2 非方向帧内预测

除了方向帧内预测模式外，还有五种非方向帧内预测模式，分别是DC\_PRED、SMOOTH、SMOOTH\_H、SMOOTH\_V和Paeth，通常用于平滑区域的预测。

* DC\_PRED模式通过平均来自上方和左侧邻近块的重建样本获取当前块的预测样本。
* SMOOTH\_V和SMOOTH\_H模式分别沿垂直和水平方向使用二次插值生成预测值，而SMOOTH模式使用沿两个方向的二次插值结果的平均值生成预测值。用于二次插值的样本包括来自上方和左侧邻近重建块的重建样本以及由上方和左侧重建样本估计的右侧和底部边界的样本。
* Paeth预测模式根据其上方（T）、左侧（L）和左上（TL）参考样本来预测每个样本，如图5所示。这些参考样本中，值最接近(T + L - TL)的那个被选为预测样本。

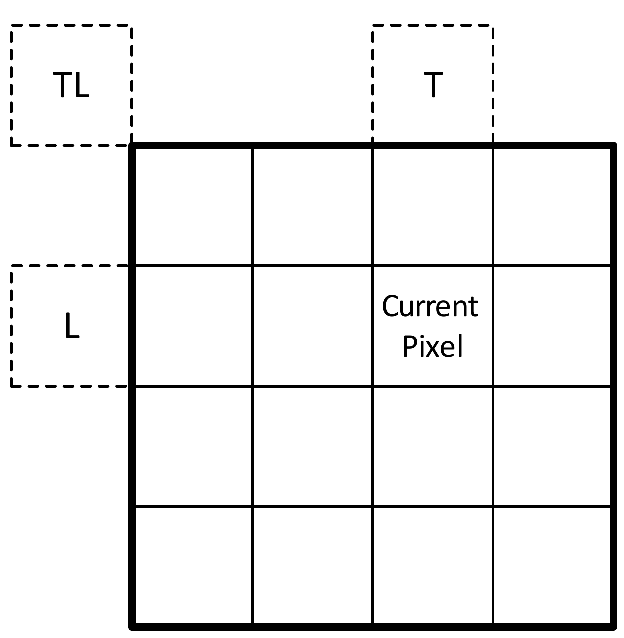


Fig.5 使用Paeth预测模式的参考像素

### 3.2.3 递归帧内预测

AV1中定义了五种递归帧内预测模式。每种模式指定一组八个7抽头滤波器。给定选择的递归帧内预测模式索引（0到4），当前块被分为多个4x2子块。对于每个4x2子块，每个样本通过使用来自上方和左侧块的七个邻近样本作为输入的7抽头插值进行预测。不同的滤波器应用于每个4x2子块内不同坐标处的样本。这个预测过程对每个4x2子块逐一进行，并且每个4x2子块生成的预测样本可以用于预测后续的4x2子块。

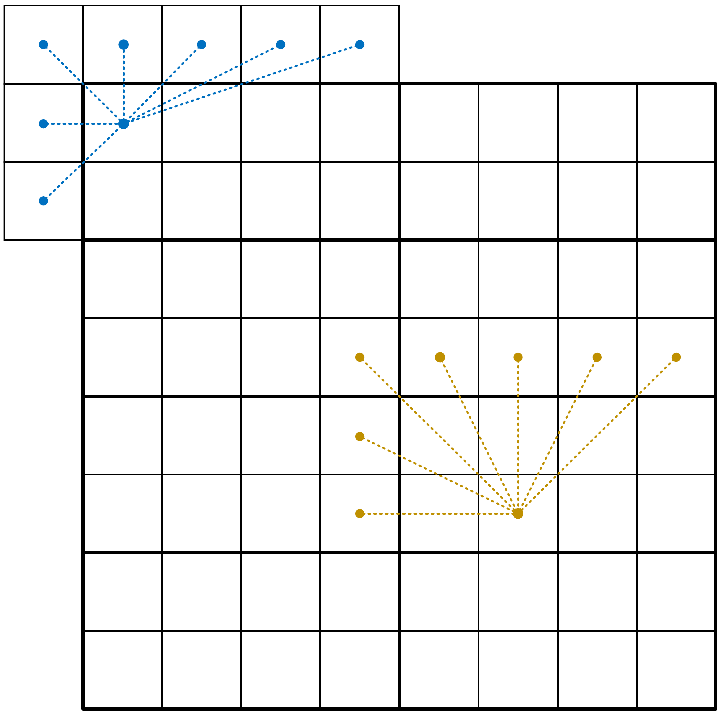


Fig.6 递归帧内预测模式示意

### 3.2.4 基于亮度的色度预测

基于亮度的色度（CfL）预测是一种仅适用于色度编码块的帧内预测模式。CfL预测模式通过线性模型使用同位重建亮度样本推导色度预测样本。当相应的亮度和色度分辨率不同时，例如对于4:2:0和4:2:2色度抽样格式，需要在输入到CfL模式之前对重建亮度样本进行采样。预测块是色度DC分量和缩放的亮度AC分量之和。块的DC分量是块的平均值，移除DC分量便可得到块的AC分量。在CfL模式中，诸如用于亮度交流分量的缩放因子等参数，在编码过程中计算并将记录在码流中。图7展示了CfL预测模式的流程图。

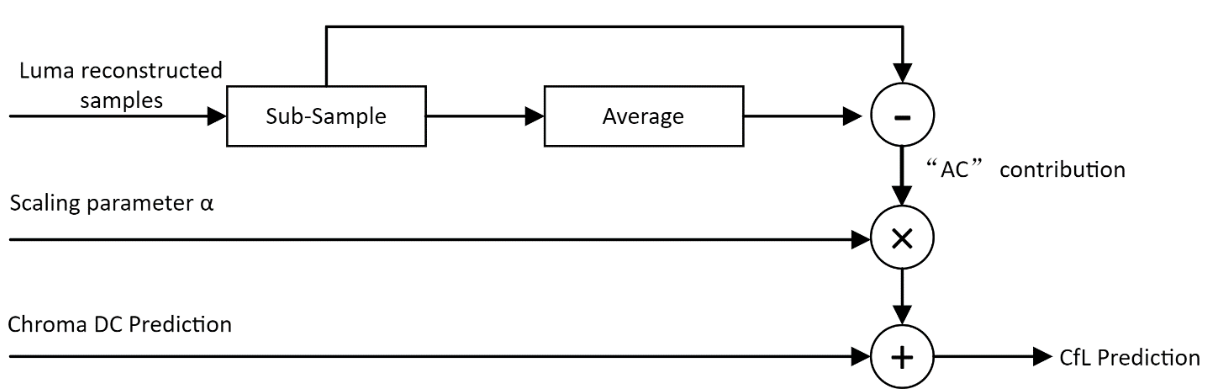


Fig.7 CfL预测模式

3.2.5 帧内预测模式signaling

对于亮度颜色分量，帧内预测模式包括56种方向帧内预测模式、5种非方向预测模式和5种递归滤波模式。以下过程用于标记编码块的帧内预测模式：

* 句法*y\_mode*首先被记录，以指示是应用八种标准方向帧内预测模式之一还是五种非方向预测模式之一。
* 如果块尺寸大于8x8且*y\_mode*指示方向帧内预测模式，则进一步记录句法*angle\_delta\_y*以指示角度增量索引，范围为-3到+3。
* 否则，如果亮度预测模式是DC\_PRED且编码块的最大宽度和高度小于或等于32，则存储标志*use\_filter\_intra*至码流以指示是否应用递归帧内预测模式。
* 如果*use\_filter\_intra*被标记为1，则进一步记录*filter\_intra\_mode*以指示应用哪种递归帧内预测模式。

类似地，对于色度颜色分量，帧内预测模式包括56种方向帧内预测模式、5种非方向预测模式和CfL预测模式。以下过程用于信号化编码块的帧内预测模式：

* 句法*uv\_mode*指示是应用八种标准方向帧内预测模式、五种非方向预测模式之一，还是应用CfL预测模式。
* 如果块尺寸大于8x8且*uv\_mode*指示方向帧内预测模式，则进一步记录语法元素*angle\_delta\_uv*以指示角度增量索引，范围为-3到+3。
* 否则，如果*uv\_mode*指示CfL模式作为帧内预测模式，则为Cb和Cr颜色分量进一步记录缩放参数alpha。

y\_mode, uv\_mode以及帧内预测模式的映射关系参考表2。

表2. y\_mode, uv\_mode以及帧内预测模式的映射关系

|  |  |  |
| --- | --- | --- |
| y\_mode | uv\_mode | Intra prediction mode |
| 0 | 0 | DC\_PRED |
| 1 | 1 | V\_PRED |
| 2 | 2 | H\_PRED |
| 3 | 3 | D45\_PRED |
| 4 | 4 | D135\_PRED |
| 5 | 5 | D113\_PRED |
| 6 | 6 | D157\_PRED |
| 7 | 7 | D203\_PRED |
| 8 | 8 | D67\_PRED |
| 9 | 9 | SMOOTH |
| 10 | 10 | SMOOTH\_V |
| 11 | 11 | SMOOTH\_H |
| 12 | 12 | Paeth |
| N/A | 13 | CfL |

intra和inter帧采用不同的方式获取*y\_mode*。intra帧中，y\_mode通过相邻亮度帧内预测模式推导得出inter帧中则是通过当前编码块大小得到。而uv\_mode则是使用同位块的亮度帧内预测模式。

## 3.3 帧间预测

### 3.3.1 参考帧系统

最多可以在解码帧缓冲区中存储八个帧。编码每个帧时，可以从解码帧缓冲区中选择最多两个参考帧，用作帧间预测的参考帧。单一参考帧间预测只使用一个帧作为参考帧。复合帧间预测使用两个参考帧，并通过结合来自这些参考帧的两个预测块生成一个预测。

Uni和Bi复合预测是复合帧间预测的特殊情况。单向复合预测是指选定的两个参考帧在显示顺序上都是当前帧的前导或后继帧，而双向复合预测是指一个参考帧在当前帧的前面，另一个参考帧在显示顺序上在当前帧之后。对于单向复合预测，适用的参考帧组合仅限于四种组合，而对于双向复合预测，所有12种可能的参考帧组合都受支持。在一个帧的编码完成后，编码器决定需要替换解码帧缓冲区中哪一个参考帧，并且此更新操作在码流中明确信号化。

解码帧缓冲区存储四种类型的参考帧，分别是：1) 最近帧：最近显示的帧；2) BWD<后向?>帧：将来显示的帧；3) 黄金帧：在较远的过去显示的帧；4) 替代参考帧（ARF）：来自过去或未来的帧。ARF 也可以选择不显示。编码器可以通过在多个原始帧的运动轨迹上进行时间过滤来合成 ARF。

3.3.2 空间运动矢量预测

可以通过利用空间邻近块，包括当前块的直接邻近块和非直接邻近块，来识别空间运动矢量（MV）预测器。如图8所示，其中每个空间邻近块是一个8x8块。

1. 检查顶部相邻行从左到右。

2. 检查左侧相邻列从上到下。

3. 检查右上相邻块。

4. 检查左上相邻块。

5. 检查第一行非相邻行从左到右。

6. 检查第一列非相邻列从上到下。

7. 检查第二行非相邻行从左到右。

8. 检查第二列非相邻列从上到下。

单一参考帧间预测中，空间 MV 预测器是通过识别使用与当前块相同的单一参考帧预测的空间邻近块，并使用它们的相关 MV 作为空间 MV 预测器生成的。

复合预测中，当一个空间邻近块使用与当前块相同的参考帧通过复合预测模式进行预测时，相关的 MV 可以用作空间 MV 预测器。

对于使用两个参考帧的复合预测，在派生 MV 预测器时不检查非相邻空间邻居。

### 3.3.3 时间运动矢量预测

除了空间邻近块外，还可以使用参考图片的同位置块导出称为时间 MV 预测器的 MV 预测器。为生成时间 MV 预测器，首先存储参考帧的 MV 与其各自的参考帧索引相关联。之后，对于当前帧的每个8x8块，识别和存储一个通过当前块的参考帧的 MV 及其相关的参考帧索引在时间 MV 缓冲区中。

3.3.4 动态运动矢量预测

运动矢量可以通过当前帧中的空间邻近块或参考帧中的时间邻近块进行预测。通过检查所有这些块，可以识别最多四个运动矢量预测器。对于单一参考帧间预测，每个参考帧都有单独的运动矢量列表。对于复合帧间预测，每个参考对使用运动矢量对列表来导出运动矢量预测器。比特流中，单一参考预测的动态参考列表索引信号化用于指定哪种运动矢量预测模式将被使用。

3.3.5 帧间预测模式信令

帧间预测模式信令时，首先信号化一个跳过模式标志以确定是否对当前编码块应用跳过模式。如果启用了跳过模式，复合帧间预测使用两个参考帧和平移运动，所有的运动矢量和参考帧都隐含派生而不是信号化。未启用跳过模式时，信号化参考帧信息。

3.3.6 平移运动补偿

平移运动模型假设一个编码块内的所有像素共享相同的单一运动矢量。运动矢量可表示为1/16像素精度。

3.3.7 变形运动补偿

与传统的假设参考和目标块之间平移运动的运动补偿不同，变形运动利用仿射模型。

3.3.8 重叠块运动补偿

重叠块运动补偿适用于单一参考预测的交织编码块。

3.3.9 复合帧间预测

使用两个参考帧的帧间预测称为复合预测。根据权重因子的导出和预测块的不同，不同的复合运动预测模式被定义。

3.3.10 复合交织内部预测

复合交织内部预测模式中，预测块是内部预测和帧间预测的组合。