

国际电信联盟

ITU-T

国际电信联盟
电信标准化部门

H.263

(01/2005)

H系列：视听和多媒体系统

视听业务的基础设施 — 活动图像编码

低比特率通信的视频编码

ITU-T H.263建议书

ITU-T



国际电信联盟

ITU-T H系列建议书
视听和多媒体系统

| | |
|--------------------|--------------------|
| 可视电话系统的特性 | H.100-H.199 |
| 视听业务的基础设施 | |
| 概述 | H.200-H.219 |
| 传输多路复用和同步 | H.220-H.229 |
| 系统概况 | H.230-H.239 |
| 通信规程 | H.240-H.259 |
| 活动图像编码 | H.260-H.279 |
| 相关系统概况 | H.280-H.299 |
| 视听业务的系统和终端设备 | H.300-H.349 |
| 视听和多媒体业务的号码簿业务体系结构 | H.350-H.359 |
| 视听和多媒体业务的服务质量体系结构 | H.360-H.369 |
| 多媒体的补充业务 | H.450-H.499 |
| 移动性和协作程序 | |
| 移动性和协作、定义、协议和程序概述 | H.500-H.509 |
| H系列多媒体系统和业务的移动性 | H.510-H.519 |
| 移动多媒体协作应用和业务 | H.520-H.529 |
| 移动多媒体应用和业务的安全性 | H.530-H.539 |
| 移动多媒体协作应用和业务的安全性 | H.540-H.549 |
| 移动性互通程序 | H.550-H.559 |
| 移动多媒体协作互通程序 | H.560-H.569 |
| 宽带和三网合一多媒体业务 | |
| 在VDSL上传送宽带多媒体业务 | H.610-H.619 |

欲了解更详细信息，请查阅ITU-T建议书目录。

ITU-T H.263建议书

低比特率通信的视频编码

摘 要

本建议书规定了一个能够用于低比特率压缩视听业务活动图像分量的编码再现。视频源编码算法的基本构形是根据ITU-T H.261建议书以及利用时域冗余度的图像间预测与减少空间冗余度的剩余信号变换编码的混合。源编码器能够在5种标准视频源格式上操作：子QCIF、QCIF、CIF、4CIF和16CIF以及也能够使用具有广泛使用范围的惯例视频格式操作。

译码器具有运动补偿能力，允许编码器中此技术的任选组合。运动补偿使用半像素精度，与使用全像素精度和环路滤波器的ITU-T H.261建议书相反。对传输符号采用可变长编码。

除基本的视频源编码算法外，作为改进的压缩性能与支持的附加能力包括18种可协商的编码选择。为增强显示能力及外部使用在比特流中也可以包括另外的辅助信息。

本第3版的H.263 纳入了以前已分别通过和出版的附件U、V、W和X，并做了如下一些更正和澄清：

- 更正图8；
- 对表1内容的澄清，关于BPPmaxKb表和图像内容；
- 对5.3.2内容的澄清，关于填充先导一个起始码的宏块；
- 关于H.263附件J和IDCT舍入误差之间交互的澄清；
- 澄清附件N，关于BCM中GN/MBA场的语法分析能力；
- 澄清附件O，关于直接双向预测的 MBs 和图像外推；
- 关于附件O中B图像与内部参考图像使用的澄清；
- 关于附件P与附件N使用的澄清；
- 附件U中图U.7的更正。

来 源

ITU-T第16研究组（2005-2008年）按照 ITU-T A.8建议书规定的程序，于2005年1月13日批准了ITU-T H.263（2005年）建议书。

前 言

国际电信联盟（ITU）是从事电信领域工作的联合国专门机构。ITU-T（国际电信联盟电信标准化部门）是国际电信联盟的常设机构,负责研究技术、操作和资费问题，并且为在世界范围内实现电信标准化，发表有关上述研究项目的建议书。

每四年一届的世界电信标准化全会（WTSA）确定ITU-T各研究组的研究课题，再由各研究组制定有关这些课题的建议书。

WTSA第1号决议规定了批准建议书须遵循的程序。

属ITU-T研究范围的某些信息技术领域的必要标准，是与国际标准化组织（ISO）和国际电工技术委员会（IEC）合作制定的。

注

本建议书为简要而使用的“主管部门”一词，既指电信主管部门，又指经认可的运营机构。

遵守本建议书的规定是以自愿为基础的，但建议书可能包含某些强制性条款（以确保例如互操作性或适用性等），只有满足所有强制性条款的规定，才能达到遵守建议书的目的。“应该”或“必须”等其它一些强制性用语及其否定形式被用于表达特定要求。使用此类用语不表示要求任何一方遵守本建议书。

知识产权

国际电联提请注意：本建议书的应用或实施可能涉及使用已申报的知识产权。国际电联对无论是其成员还是建议书制定程序之外的其它机构提出的有关已申报的知识产权的证据、有效性或适用性不表示意见。

至本建议书批准之日止，国际电联已经收到实施本建议书可能需要的受专利保护的知识产权的通知。但需要提醒实施者注意的是，这可能不是最新信息，因此大力提倡他们查询电信标准化局（TSB）的专利数据库。

© 国际电联 2005

版权所有。未经国际电联事先书面许可，不得以任何手段复制本出版物的任何部分。

目 录

| | 页 |
|-------------------------------|----|
| 1 范围 | 1 |
| 2 参考文献 | 1 |
| 2.1 规范性参考文献 | 1 |
| 2.2 资料性参考文献 | 1 |
| 3 简要说明 | 2 |
| 3.1 视频输入和输出 | 2 |
| 3.2 数字输出和输入 | 2 |
| 3.3 采样频率 | 2 |
| 3.4 源编码算法 | 2 |
| 3.5 比特速率 | 5 |
| 3.6 缓冲 | 6 |
| 3.7 对称传输 | 6 |
| 3.8 差错处理 | 6 |
| 3.9 多点操作 | 6 |
| 4 源编码器 | 7 |
| 4.1 源格式 | 7 |
| 4.2 视频源编码算法 | 9 |
| 4.3 编码控制 | 13 |
| 4.4 强制更新 | 13 |
| 4.5 起始码字节定位 | 13 |
| 5 句法和语义 | 13 |
| 5.1 图像层 | 22 |
| 5.2 块组层 | 32 |
| 5.3 宏块层 | 33 |
| 5.4 块层 | 40 |
| 6 译码处理 | 44 |
| 6.1 运动补偿 | 44 |
| 6.2 系数译码 | 46 |
| 6.3 块重构 | 48 |
| 附件 A — 反变换精度的技术规程 | 48 |
| 附件 B — 假想参考译码器 | 49 |
| 附件 C — 多点考虑 | 51 |
| C.1 冻结图像请求 | 51 |
| C.2 快速更新请求 | 51 |
| C.3 冻结图像释放 | 51 |
| C.4 连续出现多点和视频多路复用 (CPM) | 51 |

| | |
|---------------------------------------|----|
| 附件 D — 非受限运动矢量模式 | 53 |
| D.1 图像边界上的运动矢量 | 53 |
| D.2 运动矢量范围的外延 | 54 |
| 附件 E — 基于句法的算术编码模式 | 56 |
| E.1 引言 | 56 |
| E.2 SAC 编码器规格 | 57 |
| E.3 SAC 译码器规格 | 58 |
| E.4 句法 | 58 |
| E.5 PSC_FIFO | 59 |
| E.6 头部层符号 | 59 |
| E.7 宏块层和块层符号 | 60 |
| E.8 SAC 模型 | 60 |
| 附件 F — 先进的预测模式 | 64 |
| F.1 引言 | 64 |
| F.2 每宏块 4 个运动矢量 | 64 |
| F.3 亮度的交叠运动补偿 | 65 |
| 附件 G — PB 帧模式 | 68 |
| G.1 引言 | 68 |
| G.2 PB 帧和 INTRA 块 | 68 |
| G.3 块层 | 68 |
| G.4 PB 帧中 B 图像的矢量计算 | 69 |
| G.5 PB 帧中 B 块预测 | 69 |
| 附件 H — 视频编码信号的前向纠错 | 71 |
| H.1 引言 | 71 |
| H.2 纠错定帧 | 71 |
| H.3 纠错码 | 71 |
| H.4 纠错定帧的重新锁定时间 | 72 |
| 附件 I — 先进的 INTRA 编码模式 | 72 |
| I.1 引言 | 72 |
| I.2 句法 | 73 |
| I.3 译码处理 | 74 |
| 附件 J — 分块滤波器模式 | 80 |
| J.1 引言 | 80 |
| J.2 同 UMV 和 AP 模式的关系 (附件 D 和 F) | 81 |
| J.3 分块边缘滤波器定义 | 81 |
| 附件 K — 截面结构模式 | 85 |
| K.1 引言 | 85 |
| K.2 截面层结构 | 86 |

| | |
|-------------------------------------|-----|
| 附件 L — 增补的增强信息规范 | 88 |
| L.1 引言 | 88 |
| L.2 PSUPP 格式 | 88 |
| L.3 无事可做 | 89 |
| L.4 全图像冻结请求 | 89 |
| L.5 部分图像冻结请求 | 89 |
| L.6 尺寸再现部分图像冻结请求 | 90 |
| L.7 部分图像冻结释放请求 | 90 |
| L.8 全图像快拍标记 | 90 |
| L.9 部分图像快拍标记 | 91 |
| L.10 视频时间分段起始标记 | 91 |
| L.11 视频时间分段结束标记 | 91 |
| L.12 逐行精细分段起始标记 | 91 |
| L.13 逐行精细分段结束标记 | 91 |
| L.14 色度键控信息 | 92 |
| L.15 扩展的功能类型 | 94 |
| 附件 M — 改进的 PB 帧模式 | 94 |
| M.1 引言 | 94 |
| M.2 B _{PB} 宏块预测模式 | 95 |
| M.3 B 宏块双向预测的矢量计算 | 95 |
| M.4 MODE 表 | 95 |
| 附件 N — 参考图像选择模式 | 96 |
| N.1 引言 | 96 |
| N.2 视频源编码算法 | 97 |
| N.3 反向信道消息信道 | 97 |
| N.4 句法 | 98 |
| N.5 译码器处理 | 101 |
| 附件 O — 时间的、SNR 及空间的可量测性模式 | 102 |
| O.1 概述 | 102 |
| O.2 图像的传输序 | 106 |
| O.3 图像层句法 | 107 |
| O.4 宏块层句法 | 108 |
| O.5 运动矢量译码 | 112 |
| O.6 内插滤波器 | 112 |
| 附件 P — 参考图像再采样 | 115 |
| P.1 引言 | 115 |
| P.2 句法 | 118 |
| P.3 再采样算法 | 120 |
| P.4 操作实例 | 123 |
| P.5 4 因子再采样 | 126 |

| | |
|---------------------------------|-----|
| 附件 Q — 简约分辨率更新模式 | 130 |
| Q.1 引言 | 130 |
| Q.2 译码规程 | 131 |
| Q.3 参考图像的扩展 | 133 |
| Q.4 运动矢量重构 | 134 |
| Q.5 亮度的放大交叠运动补偿 | 136 |
| Q.6 简约或分辨率重构预测误差的上采样 | 138 |
| Q.7 块边界滤波器 | 141 |
| 附件 R — 独立分段译码模式 | 143 |
| R.1 引言 | 143 |
| R.2 模式操作 | 143 |
| R.3 用法限制 | 144 |
| 附件 S — 可供选择的 INTER VLC 模式 | 145 |
| S.1 引言 | 145 |
| S.2 系数的可供选择的 INTER VLC | 145 |
| S.3 CBPY 的可供选择的 INTER VLC | 146 |
| 附件 T — 修正的量化模式 | 146 |
| T.1 引言 | 146 |
| T.2 修正的 DQUANT 更新 | 146 |
| T.3 改动的色度系数量化步长尺寸 | 147 |
| T.4 修正的系数范围 | 148 |
| T.5 用法限制 | 148 |
| 附件 U — 增强参考图像选择模式 | 149 |
| U.1 引言 | 149 |
| U.2 视频源编码算法 | 150 |
| U.3 前向信道句法 | 151 |
| U.4 译码器处理 | 169 |
| U.5 反向信道信息 | 174 |
| 附件 V — 数据分裂截面模式 | 178 |
| V.1 范围 | 178 |
| V.2 数据分裂的结构 | 178 |
| V.3 与其他可选模式的关系 | 181 |
| 附件 W — 附加的补充增强信息规范 | 185 |
| W.1 范围 | 185 |
| W.2 参考文献 | 185 |
| W.3 附加的 FTYPE 取值 | 185 |
| W.4 PSUPP 字节的最大推荐数目 | 185 |
| W.5 定点离散余弦反变换 (IDCT) | 185 |
| W.6 图像消息 | 195 |

| | |
|---|-----|
| 附件 X — 档次和级别的定义 | 200 |
| X.1 范围 | 200 |
| X.2 优先模式的档次支持 | 200 |
| X.3 图像格式和图像时钟频率 | 204 |
| X.4 性能级别 | 205 |
| X.5 使用 ITU-T H.245 建议书时的通用能力定义 | 209 |
| 附录 I — 错误跟踪 | 213 |
| I.1 概述 | 213 |
| I.2 错误跟踪 | 213 |
| 附录 II — 推荐的可选增强 | 214 |

ITU-T H.263建议书

低比特率通信的视频编码

1 范围

本建议书规定一个能够用于低比特率压缩视听业务活动图像分量的编码再现。视频源编码算法的基本配置基于ITU-T H.261建议书。为改进性能和增加功能它包括18种可协商的编码选择。本建议书相当于ITU-T H.263建议书的版本2，它同原始建议书完全兼容，对原始建议书版本1的内容仅添加若干任选的特性。

2 参考文献

2.1 规范性参考文献

下列ITU-T建议书和其他参考文献的条款，通过在本建议书中的引用而构成本建议书的条款。在出版时，所指出的版本是有效的。所有的建议书和其他参考文献都面临修订，使用本建议书的各方应探讨使用下列建议书和其他参考文献最新版本的可能性。当前有效的ITU-T建议书清单定期出版。本建议书中引用某个独立文件，并非确定该文件具备建议书的地位。

- [1] ITU-R Recommendation BT.601-5 (1995), *Studio encoding parameters of digital television for standard 4:3 and wide-screen 16:9 aspect ratios.*

这里引用的参考文献 [1] 定义了依照本建议书设计的视频编译器所使用的图像的 (Y、C_B、C_R) 彩色空间域及其8比特整数表示。(参考文献 [1] 不用于规定本建议书的任何其他方面。)

2.2 资料性参考文献

作为本建议书中的说明，涉及以下附加的ITU-T建议书；

- [2] ITU-T Recommendation H.223 (2001), *Multiplexing protocol for low bit rate multimedia communication.*
- [3] ITU-T Recommendation H.242 (2004), *System for establishing communication between audiovisual terminals using digital channels up to 2 Mbit/s.*
- [4] ITU-T Recommendation H.245 (2005), *Control protocol for multimedia communication.*
- [5] ITU-T Recommendation H.261 (1993), *Video codec for audiovisual services at $p \times 64$ kbit/s.*
- [6] ITU-T Recommendation H.262 (2000) | ISO/IEC 13818-2:2000, *Information technology – Generic coding of moving pictures and associated audio information: Video.*
- [7] ITU-T Recommendation H.324 (2002), *Terminal for low bit-rate multimedia communication.*

3 简要说明

图1给出编译码器轮廓的方框图。

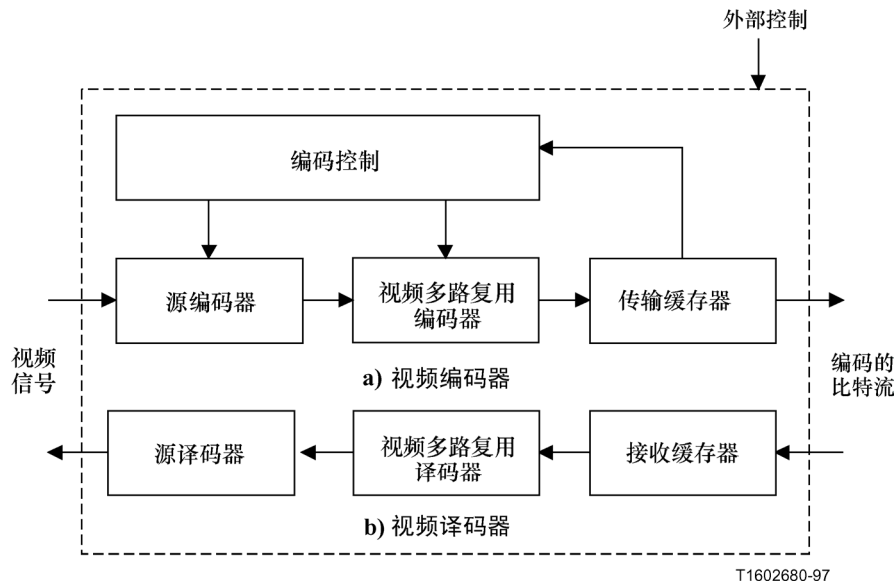


图 1/H.263—视频编译码器轮廓的方框图

3.1 视频输入和输出

为了允许单一建议书覆盖使用625行和525行电视标准区域内及区域间的使用，源编码器操作的标准源格式基于公共中间格式（CIF）。为能够使用具有广泛使用范围的任选的惯例源格式，它也可能使用外部协商（例如ITU-T H.245建议书）。输入和输出电视信号的标准可以是，例如，复合的或分量的，模拟的或数字的以及实施进入和来自源编码格式的任何必要的变换方法，此类方法本建议书不涉及。

3.2 数字输出和输入

视频编码器提供自包容的数字比特流，它可以同其他多功能信号相结合（例如ITU-T H.223建议书中规定的）。视频译码器实施相反处理。

3.3 采样频率

采用视频行速率的整数倍对图像采样。采样时钟和数字网络时钟不同步。

3.4 源编码算法

采用利用时域冗余度的像间预测与减少空间冗余度的剩余信号变换编码的混合。译码器具有运动补偿能力，允许编码器中此技术的任选组合。运动补偿使用半像素精度，与使用全像素精度和环路滤波器的ITU-T H.261建议书相反。对传输符号采用可变长编码。

另外，除核心H.263编码算法外，可以使用18种可协商的编码选择，所有这些选择或结合在一起，或单独使用（根据某种限制）。为增强显示能力及外部使用，比特流中也可以包括另外的辅助信息。提供适合于生成视频比特流的前向纠错方法，必要时提供使用。可协商的编码选择，前向纠错以及增补信息用法在随后的子节中描述。

3.4.1 连续出现多点和视频多路复用模式

在此任选模式中，至多4种分离的视频“子比特流”能够在同一视频信道内发送。本模式特性设计供连续出现多点应用或分离的逻辑信道均不可用而要求使用多路视频比特流的其他情形中使用（亦见附件C）。

3.4.2 非受限运动矢量模式

在此任选模式中，允许运动矢量指向图像的外部。使用边缘像素作为“不存在像素”的预测。若存在跨越图像边缘的运动，特别对较小的图像格式，用此模式可获取有效增量（亦见附件D）。另外，此模式包括运动矢量范围的一个外延，以至能够使用较大的运动矢量。在摄影机运动及大图像格式情形中，此模式特别有用。

3.4.3 基于句法的算术编码模式

在此任选模式中，算术编码用于替代可变长编码。SNR和重构图像是相同的，但有意义的是将产生少数几个比特（亦见附件E）。

3.4.4 先进的预测模式

在此任选模式中，对P图像的亮度部分使用交叠的块运动补偿（OBMC）（亦见附件F）。图像中的某些宏块使用4个 8×8 矢量替代一个 16×16 矢量。编码器必须决定使用哪种类型的矢量。4个矢量占用更多的比特位，但可给出更好的预测。使用本模式一般可以给出可观的改进。由于OBMC导致较少的块效应，因此可获得主观增益。

3.4.5 PB帧模式

PB帧由两种编码图像作为一个单元组成。名称PB源于ITU-T H.262建议书中存在P图像与B图像的图像种类名称。这样，PB帧由一个P图像和一个B图像组成，P图像源于先前译码的P图像预测，B图像源于先前译码的和当前正在译码的两个P图像预测。选择名称B图像是由于B图像部分可以来自过去与未来的图像双向预测。用此种编码选择，图像速率能够可观地增加，但不增加很多的比特速率（亦见附件G）。然而，也提供改进的PB帧模式（亦见附件M）。在此保留原始的PB帧模式仅出于系统兼容的考虑，该系统是在采用改进的PB帧模式之前所生成的。

3.4.6 前向纠错

若必要，应保护视频比特流的要求，指定适当的前向纠错方法提供使用。供前向纠错所提供的方法也是ITU-T H.261建议书中指定的相同的BCH码方法（亦见附件H）。

3.4.7 先进的INTRA编码模式

在此任选模式中，编码之前INTRA块首先从相邻的INTRA块预测（亦见附件I）。对INTRA块定义单独的变长码（VLC）表。该技术适用于INTRA图像内的INTRA宏块，并适用于INTER图像内的INTRA宏块。此模式有效地改善核心H.263句法的INTRA编码压缩性能。

3.4.8 分块滤波器模式

在此任选模式中，采用跨越译码的I与P图像的8×8块边界的滤波器来减少块效应（亦见附件J）。使用该滤波器的考虑是减缓译码图像中块边缘效应的发生。该滤波器影响供后续图像预测所使用的图像，并因此处于运动预测环路内。

3.4.9 截面结构模式

在此任选模式中，“截面”层用GOB层的比特流句法来替代（亦见附件K）。此模式的目的是提供增强的差错复原能力，以构造更经得起检验的同基本的分组传输交付一道使用的比特流并且极小化视频延迟。截面类似于GOB，因为它是介于图像层与宏块层之间的句法层次。然而使用截面层允许灵活地划分图像，与GOB结构所要求的固定分化及固定传输序成为鲜明对照。

3.4.10 增补的增强信息

比特流中可以包括另外的增补信息标示增强的显示能力或提供外部使用的信息（亦见附件L）。此增补信息能够用于标示全图像或部分图像冻结或冻结释放请求具有或不具有尺寸再现，也能够用于给外部使用的视频流内的特殊图像或图像序列加标签，也能够用于传送视频合成的彩色密钥信息。即使译码器或许不具备提供使用它的或者甚至适当解释它的增强的能力，比特流中也可以出现增补信息。除非提供该请求能力的要求已经通过外部方式协商，否则不允许译码器轻易地丢弃增补信息。

3.4.11 改进的PB帧模式

此任选模式表示针对PB帧模式选择的改进（亦见附件G和M）。两种模式之间的主要差异是：在改进的PB帧模式中，每个B块可以使用单独的运动矢量前向预测或采用零矢量后向预测。在下标尺的P矢量不是好的B预测候选的情况下，该模式有效地改善了编码效率。当先前的P帧与PB帧之间存在景物切割时后向预测是特别有效的。

3.4.12 参考图像选择模式

通过允许来自图像而不是参考图像最近发送的时间预测，提供此任选模式以改进误差易出信道上实时视频通信的特性（亦见附件N）。此模式可以同反向信道状态消息一起使用，即发送回编码器通告有关其比特流是否正适当被接受的消息。在误差易出信道环境中，此模式允许编码器针对该信道的条件最佳化视频编码。

3.4.13 时间的、SNR和空间的可量测性模式

在此任选模式中，存在对时间的、SNR和空间的可量测性的支持（亦见附件O）。可量测性意味着比特流由基础层和一个或多个有关的增强层组成。基础层是单独可分离译码的比特流。增强层能够连带基础层一起译码来增强可感知的质量，通过或者增加图像速率、增强图像质量，或者增加图像的尺寸。SNR可量测性涉及增加信息来增强图像质量，而无需增加图像分辨率。空间可量测性涉及增加信息来增强图像质量，通过增加图像分辨率，或水平或垂直或两者兼有。也存在通过使用B图像对时间可量测性的支持。B图像作为可量测性增强（信息）包含来自参考层中的两个图像的双向预测，一个是当前图像时间域之前的图像，一个是时间域滞后的图像。B图像可以通过增强层信息来增加可感知的质量，通过增加显示的增强视频序列的图像速率。此模式对于具有可变带宽容量及连带纠错方案的异形网络而言是很有用处的。

3.4.14 参考图像再采样模式

提供句法支持供视频图像预测所使用的参考图像的一个任选模式，对当前的输入图像而言，在其用于组成预测器之前，该参考图像通过再采样操作处理（亦见附件P）。此模式允许视频编码的适当图像分辨率的有效动态选择，对于用作全局运动补偿器或特技信号发生器也能够支持图像扭曲变形。

3.4.15 简约分辨率更新模式

提供任选模式允许对具有较高分辨率的参考图像简约分辨率更新（亦见附件Q）。当编码高活动场景时，该模式将预期使用，并允许编码器在可以表示的场景活动部位增加图像速率的同时，在该场景的更多静止区域维持较高的分辨率表示。

3.4.16 独立分段译码模式

提供任选模式允许准备构造的模式没有任何跨越GOB或多GOB视频图像分段或截面边界的数据相关性（亦见附件R）。该模式通过防止差错数据跨越视频图像分段区域边界的传播提供误差鲁棒性。

3.4.17 可供选择的INTER VLC模式

当INTER图像中变化显著相当明显时，提供该任选模式以改进INTER图像编码的效率（亦见附件S）。此模式效率改进通过允许最初供INTRA图像所设计的VLC编码同样准备供某些INTER图像系数所使用来获取。

3.4.18 修正的量化模式

提供任选模式改进编码的比特率控制能力，减少色度量化的误差、扩展可表示的DCT系数范围，以及在系数值上设置某些限制（亦见附件T）。通过展宽步长尺寸范围可指定的变化，此模式修正比特流的差分量化步长尺寸参数的语义。它也减少供色度数据所使用的量化步长尺寸。展宽DCT系数等级的范围以确保任何可能的系数值均能够编码到步长尺寸所能允许的精度内。此模式中也在系数上设置了某些约束，因此增加了误差检测功能和极小化译码器的复杂度。

3.5 比特速率

外部提供传输时钟。视频比特速率可变。本建议书中，没有给出任何视频比特速率限制；限制将由终端或网络给出。

3.6 缓冲

编码器应控制它的输出比特流遵从附件B中规定的假想参考译码器要求。视频数据应在每个有效时钟周期提供。这可由MCBPC填充确保（见表7和表8），或当使用前向纠错时，亦可通过前向纠错填充帧确保（见附件H）。

任何单一图像编码产生的比特数应不超过以1024比特为单位度量的由BPPmaxKb参数指定的最大值。BPPmaxKb参数的最小可允许值取决于比特流中所使用的经协商的最大图像尺寸（见表1）。图像尺寸度量为亮度（Y）分量的图像宽度乘以图像高度，以像素为单位。编码器可以使用比表1中所指定的更大的BPPmaxKb值，提供的更大值首先通过外部手段协商，例如ITU-T H.245建议书。

当时间的、SNR和空间的可量测性模式（附件O）使用时，对于每个增强层中的每个图像而言，所发送的比特数均不应超过BPPmaxKb指定的最大值。

表 1/H.263—不同源图像格式的最小BPPmaxKb

| 以像素度量的Y图像尺寸 (=图像帧头中给出的宽×高) | 最小BPPmaxKb |
|-------------------------------|------------|
| 至多 25 344 (或 QCIF) | 64 |
| 25 360 到 101 376 (或 CIF) | 256 |
| 101 392 到 405 504 (或 4CIF) | 512 |
| 405 520 及以上 | 1024 |

3.7 对称传输

编译码器可供双向或单向的可视通信使用。

3.8 差错处理

差错处理将由外部手段提供（例如，ITU-T H.223建议书）。若外部手段不提供（例如，ITU-T H.221建议书中），则附件H中描述的任选的纠错码和定帧可以使用。

译码器能够发送指令对其下一个图像的一个或多个GOB（或者若使用附件K，则截面）采用INTRA模式编码，具有诸如避免缓存器溢出之类的编码参数。译码器也能发送指令仅传送非空GOB头，只要截面结构模式（见附件K）未使用。这些信号的传输方法通过外部手段协商（例如，ITU-T H.245建议书）。

3.9 多点操作

附件C中包括支持切换多点操作的必要性能。

4 源编码器

4.1 源格式

源编码器操作于具有以下形式定义的源格式非隔行图像上：

- 1) 图像格式，由每行像素数、每个图像的行数及像素宽高比所确定；和
- 2) 图像间计时，由图像时钟频率（PCF）所确定。例如公共中间格式（CIF）具有每行352像素、288行、12:11的像素宽高比及每秒30 000/1001幅图像的图像时钟频率。

源编码器操作在每秒以30 000/1001（近似29.97）次图像时钟频率（PCF）发生的非隔行图像上，称之为CIF PCF。经由外部手段协商使用任选的惯例PCF也是可能的。惯例PCF由1 800 000/（时钟参数*时钟变换因子）给出，其中时钟参数可取1-127之间的值并且时钟变换因子可以或为1000或为1001。图像时钟频率的容限为 $\pm 50 \times 10^{-6}$ 。

图像按一个亮度分量和两个色差分量编码（Y，C_B和C_R）。这些分量及表示其采样值的代码如ITU-R BT.601-5建议书的规定。

- 黑=16；
- 白=235；
- 零色差=128；
- 峰值色差=16和240。

这些值是标称值并且编码算法运行于1到254之间的输入值。

存在5种标准图像格式：子QCIF、QCIF、CIF、4CIF和16CIF。协商惯例图像格式也是可能的。对所有的这些图像格式而言，亮度采样结构是正交排列的每行dx像素，每图像dy行。两个色差分量中的每一分量采样亦为正交排列，每行dx/2像素，每图像dy/2行。对于每一标准的图像格式，dx、dy、dx/2、dy/2的值在表2中给出。

表 2/H.263 — 每个标准的H.263图像格式的行数和每行的像素数

| 图像格式 | 亮度行的像素数 (dx) | 亮度行的数目 (dy) | 色度行的像素数 (dx/2) | 色度行的数目 (dy/2) |
|--------|-----------------|----------------|-------------------|------------------|
| 子 QCIF | 128 | 96 | 64 | 48 |
| QCIF | 176 | 144 | 88 | 72 |
| CIF | 352 | 288 | 176 | 144 |
| 4CIF | 704 | 576 | 352 | 288 |
| 16CIF | 1408 | 1152 | 704 | 576 |

对于所有的图像格式，色差采样定位的方法是使其块边界恰好与亮度块边界重合，如图2所示。对每一标准的图像格式而言，像素宽高比是相同的，它与ITU-T H.261建议书中对QCIF和CIF规定的宽高比相同；(288/3) : (352/4)，以互为素数形式表示简化为12:11。除子QCIF图像格式外，由其他所有标准图像格式所覆盖的图像区域均有4:3的宽高比。

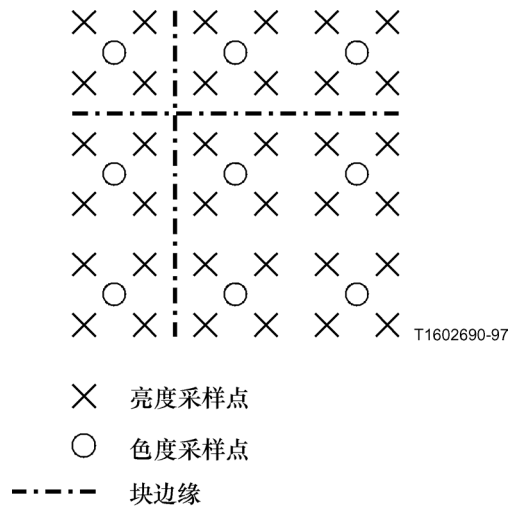


图 2/H.263—亮度样点与色度样点的位置关系

若惯例像素宽高比使用首次通过外部手段协商，则惯例图像格式可以有惯例像素宽高比，如表3所描述的。惯例图像格式可以有任意数目的行和每行任意数目的像素，只要提供的行的数目被4除尽且在区间[4, ..., 1152]中，以及只要提供的每行的像素数目也可被4除尽且在区间[4, ..., 2048]中。对具有宽度或高度不被16除尽的图像格式而言，以假定图像宽度或高度具有被16除尽的下一个最大尺寸的相同方式译码该图像，然后在图像右部和底部截断图像为仅出于显示考虑的、适当的宽度或高度。

表 3/H.263—惯例像素宽高比

| 像素宽高比 | 像素宽度：像素高度 |
|------------------|-------------------|
| 正方形 | 1 : 1 |
| CIF | 12 : 11 |
| 525 型的 4 : 3 图像 | 10 : 11 |
| CIF 的 16 : 9 图像 | 16 : 11 |
| 525 型的 16 : 9 图像 | 40 : 33 |
| 扩展的 PAR | m : n, m 和 n 互为素数 |

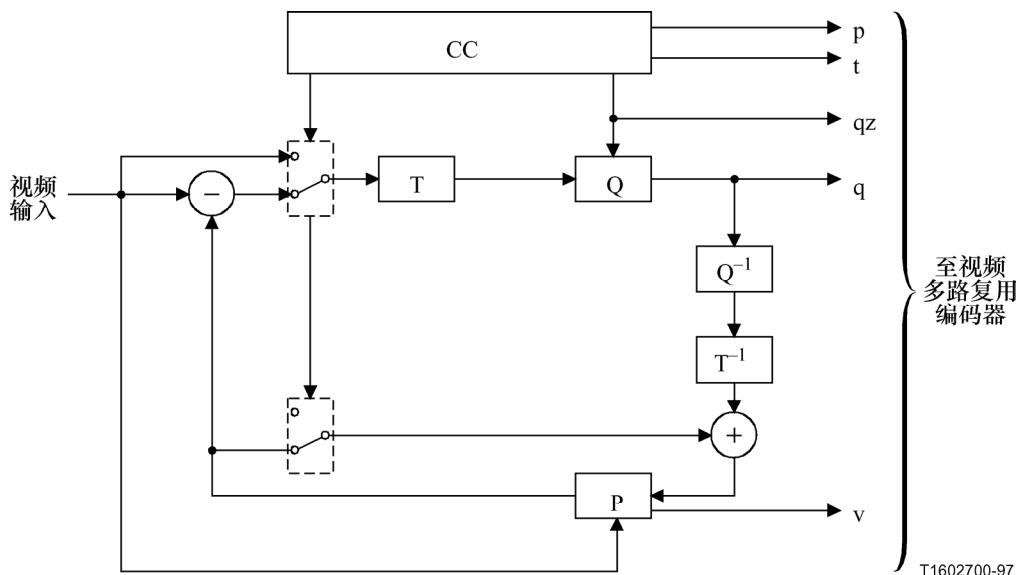
所有译码器和编码器均应能够使用CIF图像时钟频率操作。一些译码器和编码器也可以支持惯例图像时钟频率操作。所有译码器均应能够使用子QCIF图像格式操作。所有译码器也应能够使用QCIF图像格式操作。一些译码器也可以采用CIF、4CIF或16CIF或惯例图像格式操作。编码器应能够采用子QCIF和QCIF图像格式之一操作。编码器决定使用这两种格式中的哪一种格式，并且不强迫能够采用两种格式操作。一些编码器也能采用CIF、4CIF、16CIF或惯例图像格式操作。哪种任选格式和哪种图像时钟频率能够由译码器掌握通过外部手段标示，例如ITU-T H.245建议书。对于可能存在的图像格式和视频编码算法的一个完整的概括，涉及终端种类，例如ITU-T H.324建议书。

注 — 对CIF出于实用的考虑，每行像素数与对525行或625行视频源的亮度信号和色差信号的有效部分分别以6.75 MHz和3.375 MHz进行采样是兼容的。这些频率同ITU-R BT.601-5建议书中的频率有一简单关系。

通过传输图像间所具有的非传输图像的最小数值，提供了限制编码器其最大图像速率的手段。这个最小数值的选择应通过外部手段进行（例如ITU-T H.245建议书）。PB帧模式中，在对非传输图像的最小数值计算时，PB帧单元的P图像和B图像被当作两个单独的图像。

4.2 视频源编码算法

图3显示源编码器的一般形式。主要环节是预测、块变换和量化。



- T 变换
- Q 量化器
- P 具有运动补偿可变延迟的图像存储器
- CC 编码控制
- p INTAR/INTER方式标记
- t 传输与否标记
- qz 量化器指示
- q 变换系数的量化索引号
- v 运动矢量

图 3/H.263—源编码器

4.2.1 GOB、截面、宏块和块

每个图像被分成或块组（GOB）或截面。

块组（GOB）由最多 $K \cdot 16$ 行组成，其中 K 依赖于图像格式中行的数目并依赖于任选的简约分辨率更新模式是否使用（见附件Q）。相关性在表4中显示。若行数小于或等于400并且任选的简约分辨率更新模式未使用，则 $K=1$ 。若行数小于或等于800并且任选的简约分辨率更新模式在使用，或者行数大于400，则 $K=2$ 。若行数大于800，则 $K=4$ 。当使用惯例图像尺寸时，在最后（最底部）的GOB中行数可以小于 $K \cdot 16$ ，只要图像中行数不被 $K \cdot 16$ 除尽。然而，作为每个标准图像格式行的数目是 $K \cdot 16$ 的整数倍，因此在每个标准图像格式中，每个GOB均为 $K \cdot 16$ 行。这样，例如，若任选的简约分辨率更新模式未使用，则每个图像的GOB数目对子QCIF是6，对QCIF是9，对CIF、4CIF和16CIF是18。GOB编号通过使用GOB的垂直扫描来实现，起始于最上面的GOB（编号0）并结束于最下面的GOB。图4给出CIF图像格式的图像中GOB

排列的实例。对每个GOB而言，数据由GOB头（可以是空的）和后随的宏块数据组成。对GOB而言，以递增的GOB编号形式传输每个GOB数据。

表 4/H.263—GOB尺寸的参数K定义

| 行的数目dy | 当未在RRU模式时K的值 | 当在RRU模式时K的值 |
|------------------|--------------|-------------|
| 4,, 400 | 1 | 2 |
| 404,, 800 | 2 | 2 |
| 804,, 1152 | 4 | 4 |

截面结构模式在附件K中描述。截面类似于GOB，它们均为句法的多宏块层，但截面有比GOB更为灵活的形状和用法，并在某些条件下截面可以任何顺序在比特流中出现。

| |
|----|
| 0 |
| 1 |
| 2 |
| 3 |
| 4 |
| 5 |
| 6 |
| 7 |
| 8 |
| 9 |
| 10 |
| 11 |
| 12 |
| 13 |
| 14 |
| 15 |
| 16 |
| 17 |

图 4/H.263—CIF图像中块组的排列

每个GOB被分成宏块。宏块结构依赖于任选的简约分辨率更新（RRU）模式是否使用（见附件Q）。除非在RRU模式中，否则每个宏块涉及16像素×16行的Y分量与空间相应的8像素×8行的C_B和C_R分量。更进一步，一个宏块由4个亮度块和两个空间相应的色差块组成，如图5所示。这样每个亮度或色度块均涉及Y、C_B或C_R的8像素×8行的块。除非在RRU模式中，否则对于QCIF、QCIF、CIF格式GOB由1个宏块行组成，对4CIF格式由2个宏块行组成，对16CIF格式由4个宏块行组成。

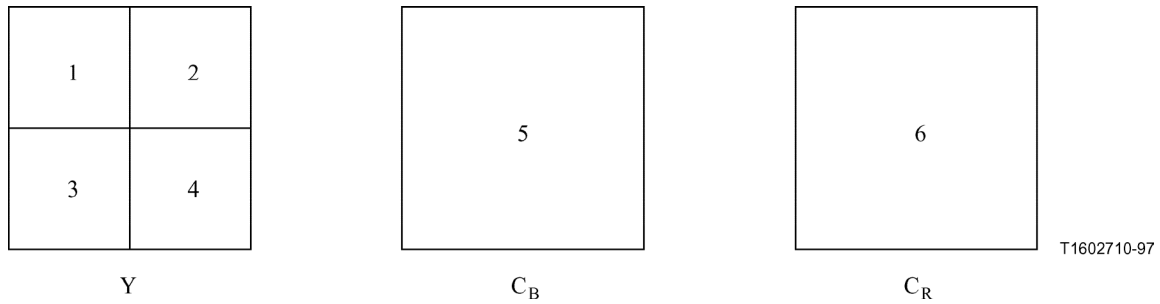


图 5/H.263—宏块中的块排列

在RRU模式时，宏块涉及32像素×32行的Y分量与空间相应的16像素×16行的C_B和C_R分量，并且每个亮度或色度块均涉及Y、C_B或C_R的16像素×16行的块。对CIF和4CIF格式而言，GOB由1个宏块行组成，对16CIF格式由2个宏块行组成。

宏块编号通过宏块行从左至右的水平扫描实现，起始于最上面的宏块行，结束于最下面的宏块行。对宏块而言，以递增宏块编号的形式传输每个宏块数据。对块而言，以递增块编号的形式传输每个块数据（见图5）。

模式和传输块选择准则不受建议书约束，并且作为编码控制策略的一部分可以动态的变化。变换传输块并对生成的变换系数进行量化和熵编码。

4.2.2 预测

预测的基本形式在图像之间并可以加入运动补偿（见4.2.3）。在采用时间预测方式中，编码模式称之为INTER；若未采用任何时间预测方式，则编码模式称之为INTRA。INTRA编码模式能够在图像级（对I图像INTRA或对P图像INTER）或者P图像中的宏块级上标示。在任选的PB帧模式中，B图像总是用INTER模式编码。B图像部分地被双向预测（参见附件G）。

总计H.263有七种基本的图像类型（其中仅头两个图像类型是强制的）按照其预测结构的形式基本定义如下：

- 1) INTRA：对预测而言，该图像不具有任何参考图像（也称I图像）；
- 2) INTER：该图像使用时间域过去的参考图像（也称P图像）；
- 3) PB：表示两个图像并具有时间域过去的参考图像的帧（见附件G）；
- 4) 改进的PB：功能类似但通常比PB帧更好的帧（见附件M）；
- 5) B：该图像具有两个参考图像，其中一个时间域超前B图像，以及另一个时间域滞后B图像，且具有相同的图像尺寸（见附件O）；
- 6) EI：该图像具有时间域同时存在的参考图像，其中该参考图像具有或者相同或者较小的图像尺寸（见附件O）；

7) EP: 该图像具有两个参考图像, 其中一个时间域超前EP图像, 以及另一个时间域同时存在, 且具有或者相同或者较小的图像尺寸(见附件O);

作为这里使用的, “参考”或“参照物”图像是指包含能够作为另一个图像译码根据通过参考使用的数据的图像。这种通过参考的使用也称之为“预测”, 虽然有时它或许实际指示用于相反的时间方向。

4.2.3 运动补偿

译码器将接受每宏块一个矢量或者若使用先进的预测模式或分块滤波器模式, 则译码器将接受每宏块1个或4个矢量(见附件F和J)。若使用PB帧模式, 为适应B宏块预测的运动矢量, 则每宏块可以传输一个附加的 Δ (增量)矢量。类似的, 改进的PB帧宏块(见附件M)可以包括附加的前向运动矢量。B图像宏块(见附件O)能够同前向和后向运动矢量一起传输, 以及EP图像能够同前向运动矢量一起传输。

运动矢量的水平分量和垂直分量均为整数或半整数值。在缺省预测模式中, 这些值受限于 $[-16, 15.5]$ 的范围(对B图像的前向和后向运动矢量分量而言, 这也是有效的)。

然而, 在非受限运动矢量模式中, 矢量分量的最大范围将增加。若PLUSPTYPE缺席, 则区间范围为 $[-31.5, 31.5]$, 具有约束条件为只要预测值处于 $[-15.5, 16]$ 范围内, 围绕每个运动矢量分量的预测值仅在 $[-16, 15.5]$ 范围内的值可以达到。若PLUSPTYPE缺席, 并且预测值处于 $[-15.5, 16]$ 范围之外, 则在区间 $[-31.5, 31.5]$ 范围内具有同预测值相同符号的所有值加上零值能够达到。若PLUSPTYPE存在, 则运动矢量值极少限制(亦见附件D)。

在简约分辨率更新模式中, 运动矢量范围放大到近似双倍尺寸, 并且限制每个矢量分量仅具有半整数值或零值。因此在缺省简约分辨率更新模式中, 每个运动矢量分量的范围为 $[-31.5, 30.5]$ (见附件Q)并且若非受限运动矢量模式也使用则该运动矢量分量的范围更大(亦见附件D)。

运动矢量的水平分量或垂直分量的正值表明预测由来自空间位置处于正被预测像素右方或下方的参考图像中的像素构成。

运动矢量的限制条件是被它们参考的所有像素都在编码图像区域内, 当使用非受限运动矢量模式、先进的预测模式或分块滤波器模式时(见附件D、F及J)或在时间的、SNR及空间可量测性模式的B和EP图像中(见附件O)除外。

4.2.4 量化

除非使用任选的先进的INTRA编码模式或修正的量化模式, 否则对INTRA块的首系数量化器的编号为1, 对所有其他系数的量化器编号为31。宏块内除INTRA块的首系数外, 对所有系数均使用同一个量化器。判决水平不规定。INTRA块的首系数通常为变换DC值采用步长尺寸8的均匀量化。其他31个量化器中的每一个均使用等空间重构电平, 具有围绕零的中心死区并具有步长尺寸是2到62之间的偶数值。精确公式参见6.2。对于使用先进的INTRA编码模式的量化参见附件I。对于使用修正的量化模式的量化参见附件T。

注一 对于较小的量化步长尺寸, 不能表示变换系数的全动态范围, 除非使用任选的修正的量化模式。

4.3 编码控制

可以通过改变若干参数来控制编码视频数据生成的速率。这些包括源编码器之前的处理、量化器、块重要性判决和时域子采样。这些措施在整个控制策略中所占的比重不是本建议书的课题。

一旦引用，时域子采样通过丢弃完整图像来实现。

对于视频信号的空间域和时间域分辨率间的某种权衡，译码器能够标示它的选择。在呼叫起始，编码器应标示其缺省的权衡并指示它是否具备响应译码器请求改变这一权衡的能力。这些信号的传输方法通过外部手段进行（例如，ITU-T H.245建议书）。

4.4 强制更新

这种功能通过强制使用INTRA模式的编码算法来实现。更新模式不规定。为控制反变换失配差错积累，当P图像中每个宏块的系数传输时，每132次中至少有一次应使用INTRA模式编码此宏块。当使用任选的EP图像时（见附件O）类似的要求适用，即当每个宏块的系数传输时，每132次中至少有一次此宏块应使用INTRA或向上的模式编码。

4.5 起始码字节定位

起始码字节定位通过起始码之前插入由少于8个0比特组成的填充码字来实现，以致起始码的首比特是字节的第一（最高有效）比特位。若起始码的最高有效比特位置从H.263比特流中的第一比特起为是8比特的一个倍数，则该起始码被字节定位。所有图像、截面及EOSBS起始码应字节定位并且GOB和EOS起始码可以字节定位。

注1 — 对某个图像而言，占用的比特数目是可变的，但总是8比特的倍数。

注2 — H.324要求H.263编码器调整图像起始码具有传送到适配层（AL_SDU）逻辑信息单元的起始。

5 句法和语义

视频句法以分级结构排列成具有4个基本层的分级结构。从顶到底，该层为：

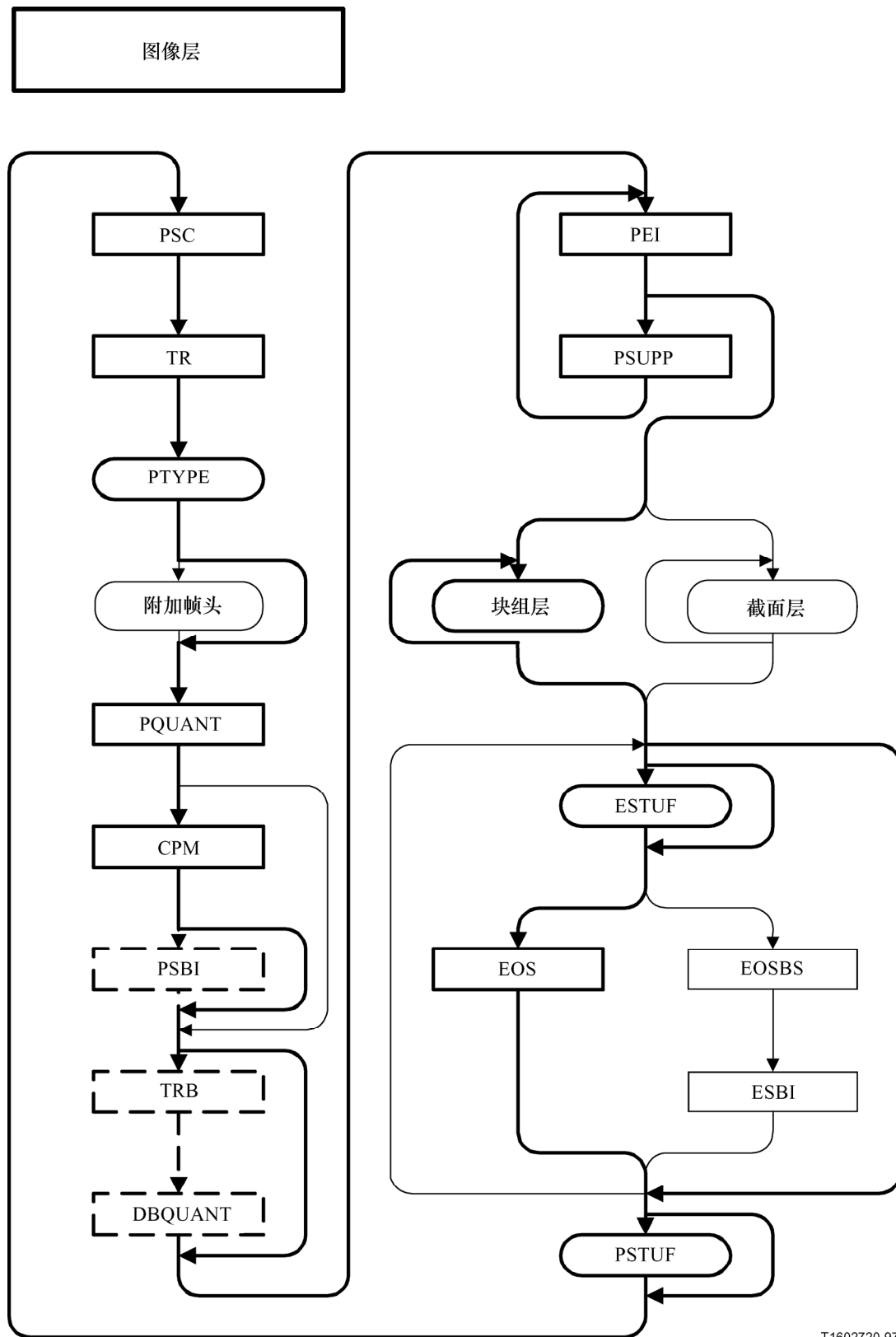
- 图像；
- 块组或截面或视频图像分段；
- 宏块；
- 块。

句法框图在图6中显示，以下组成该框图的解释指南：

- 1) 箭头路径显示句法单元的可能流向。对于箭头路径图解式含义而言，任何具有零长度的句法单元被认为缺席（这样，例如由于PSTUF场的长度可以为零，因此存在箭头路径绕行PSTUF，尽管PSTUF具有强制性质）。
- 2) 每个句法单元的缩写和语义在以后各节中规定。
- 3) 使用粗实线显示的句法单元箭头路径集代表操作的句法流的“基线”模式，无须使用任何任选的增强。（此句法在本建议书的版本1中也存在，并在任何情形中保持不变。）

- 4) 使用粗虚线显示的句法单元箭头路径集代表本建议书的版本1和版本2中均曾出现的任选的增强句法流中的附加单元。（该句法在任何情形中保持不变。）
- 5) 使用细实线显示的句法单元箭头路径集代表特定于版本2中增加的附加任选性质的任选的增强句法流中的附加新单元。（该句法在版本1中不存在。）
- 6) 具有矩形框边界显示的句法单元场指示固定长度场，具有圆形边界显示的句法单元场指示可变长度场。一个句法单元（DQUANT）若具有两种类型边界显示，则由于它具有或可变或固定的长度。
- 7) 固定长度场定义为这样的场，即场的长度不依赖于场自身内容中的数据。此场的长度或者总是相同或者由句法流中的前导数据确定。
- 8) 术语“层”用于涉及可以被解释和图解化为独特实体的任何句法部分。

除非另有规定，否则最高有效比特总是首先被传输。此为比特1即是本建议书代码表中的最左比特。除非另有规定，否则所有未用或空闲比特均置于“1”。空闲比特的功能未被ITU指定之前不应使用。

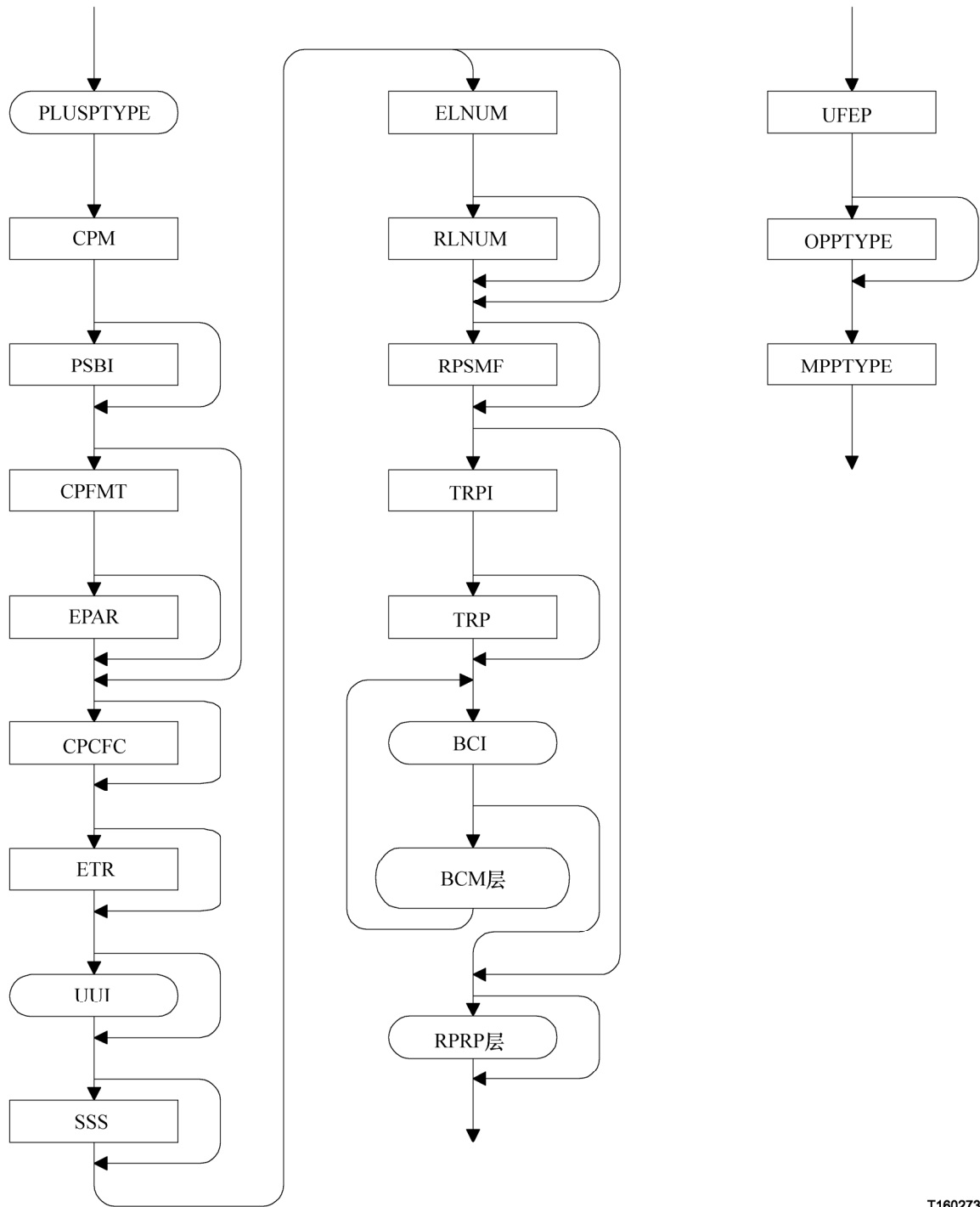


T1602720-97

图 6/H.263—视频比特流的句法框图（共7张，第1张）

附加帧头

附加类型



T1602730-97

图 6/H.263—视频比特流的句法框图（共7张，第2张）

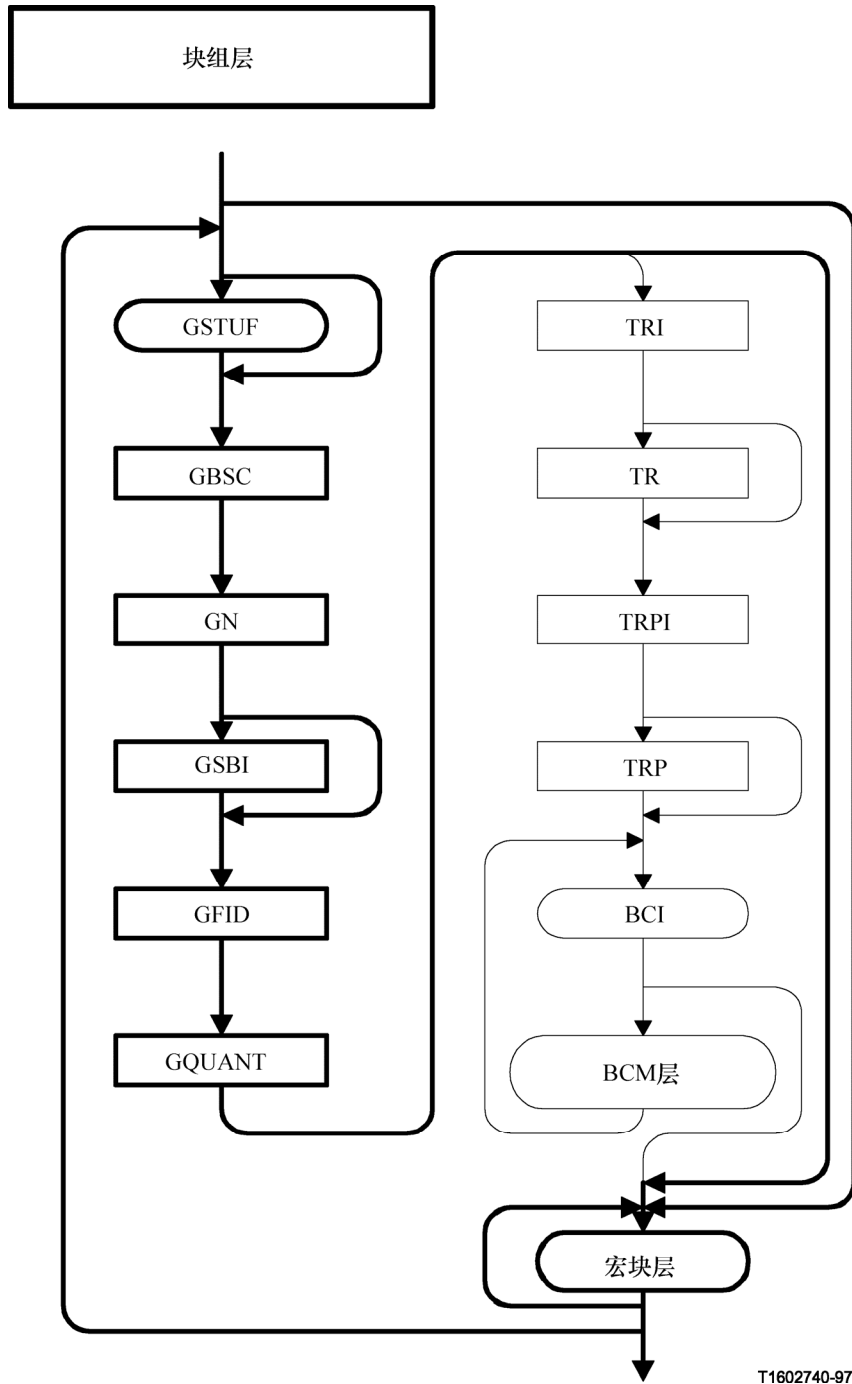
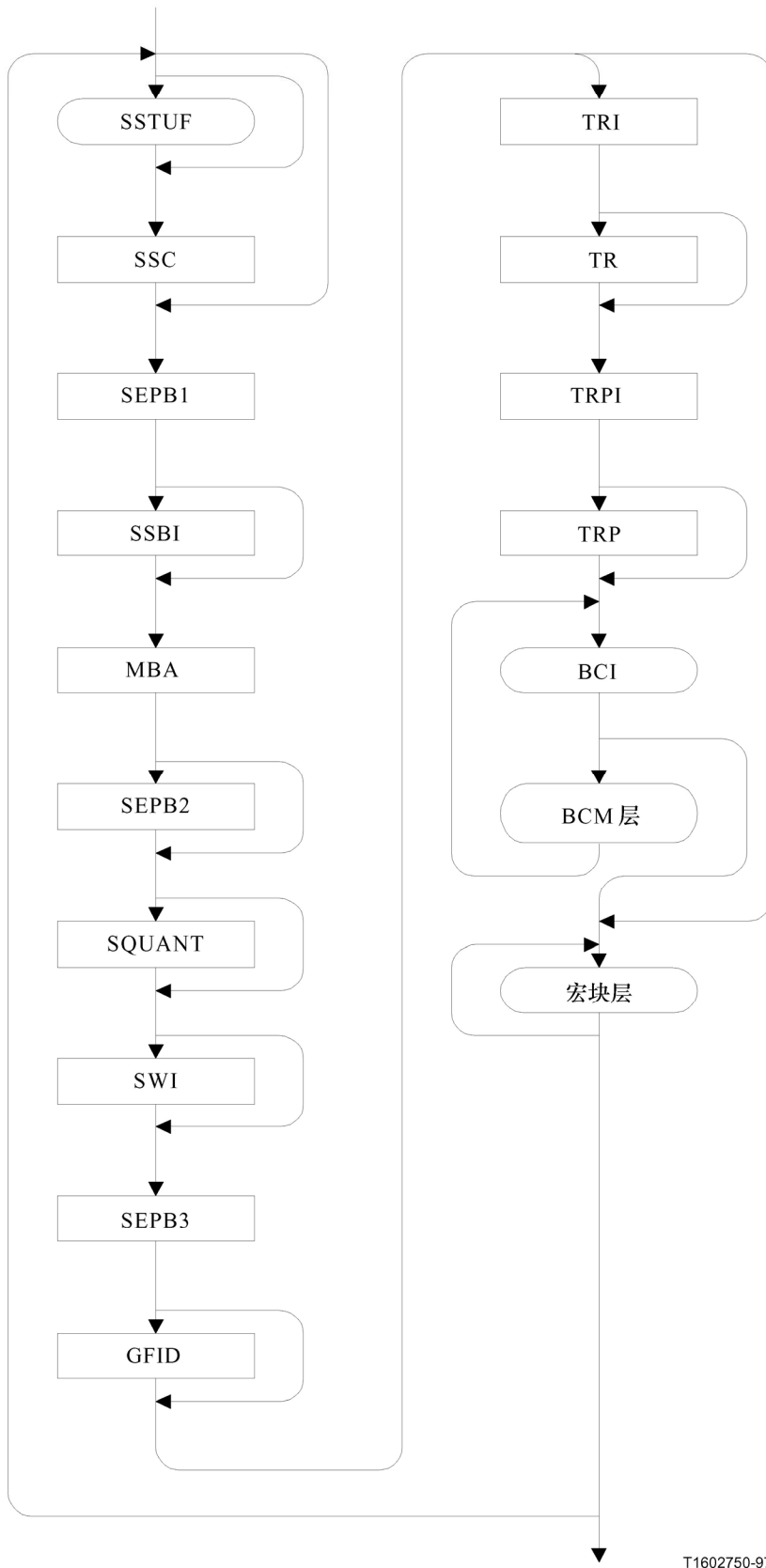


图 6/H.263—视频比特流的句法框图（共7张，第3张）

截面层



T1602750-97

图 6/H.263—视频比特流的句法框图（共7张，第4张）

BCM可分离逻辑信道层

BCM层

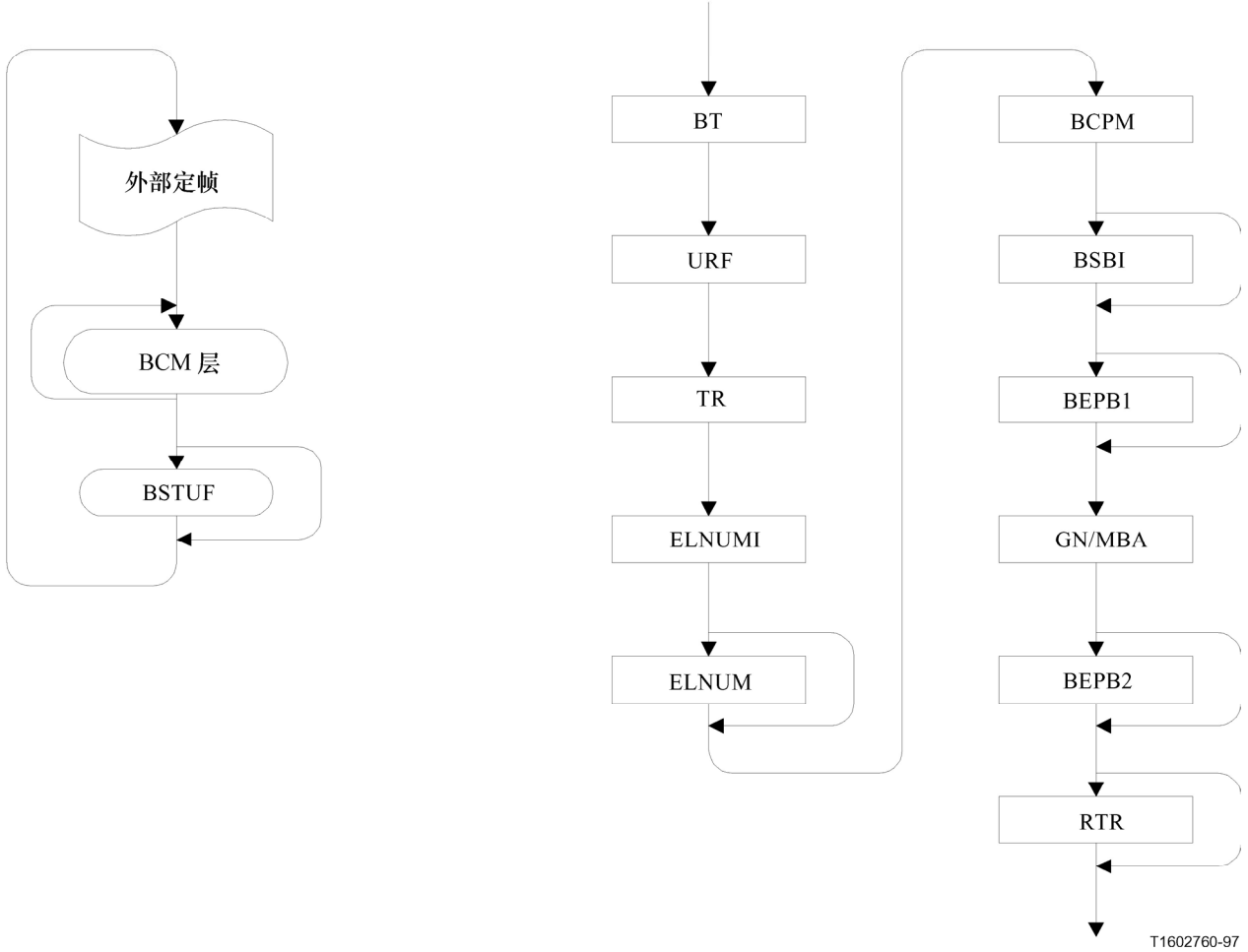
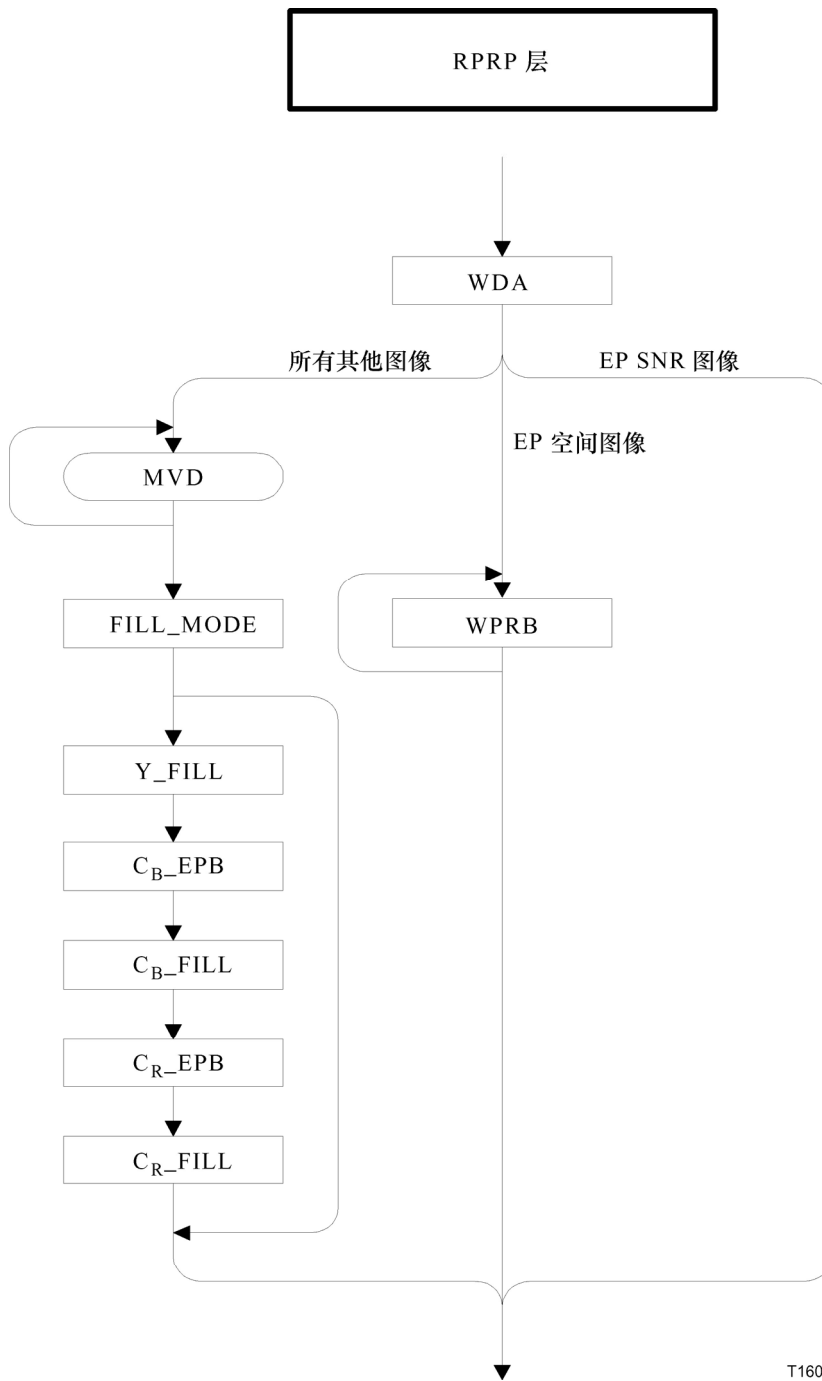
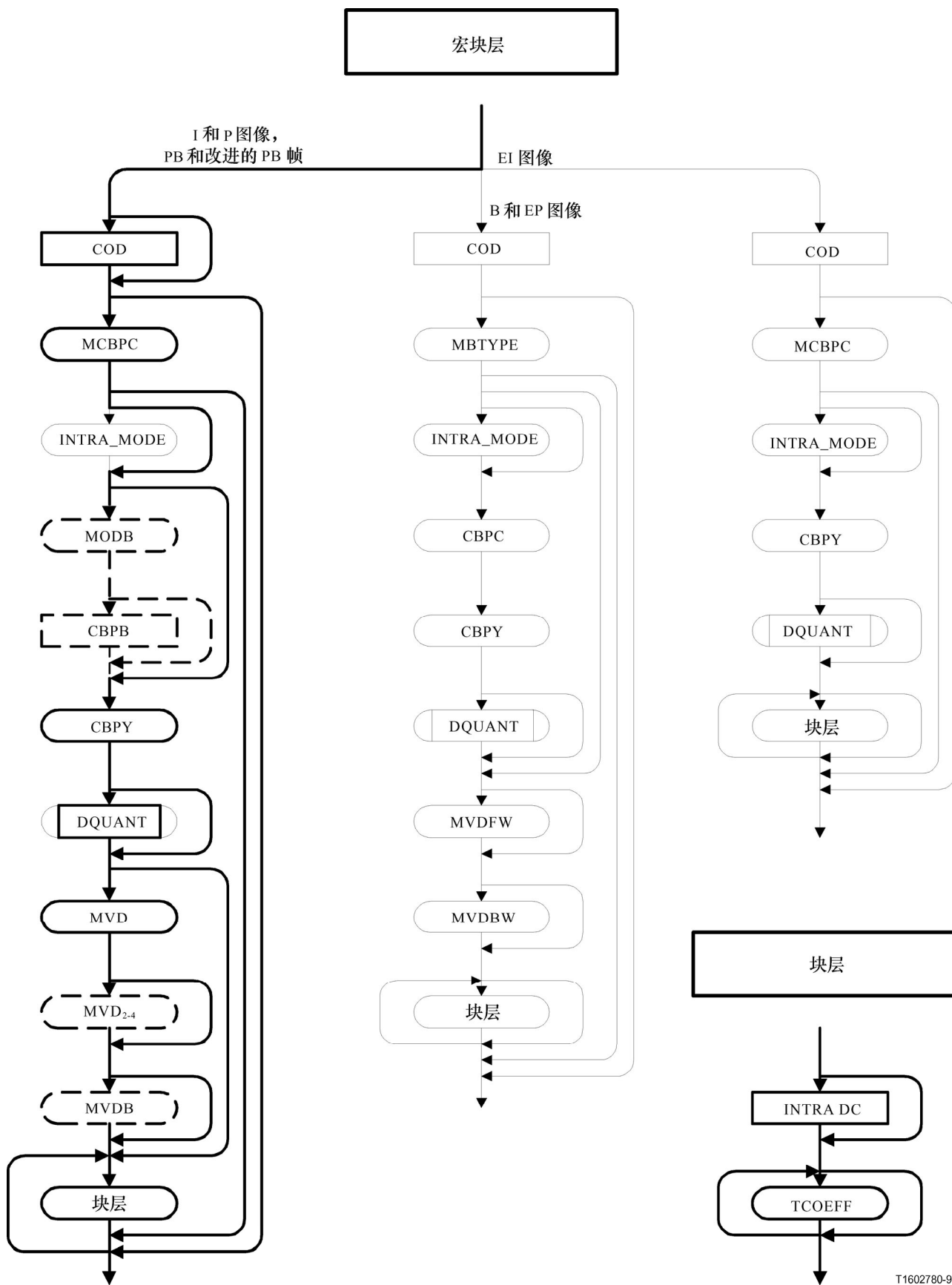


图 6/H.263—视频比特流的句法框图（共7张，第5张）



T1602770-97

图 6/H.263—视频比特流的句法框图（共7张，第6张）



T1602780-97

图 6/H.263—视频比特流的句法框图（共7张，第7张）

5.1 图像层

每帧图像数据由图像头和跟随的块组或截面数据组成，最后跟随着任选的序列结束码和填充比特。其结构在图7中显示，其中该图像不包括任选的PLUSPTYPE数据场，只要由CPM指示，PSBI就存在。只要PTYPE指示使用“PB帧”模式，TR_B和DBQUANT就存在（除非PLUSPTYPE场存在并在那点指示使用DBQUANT）。

如PTYPE的比特6-8中所指示的那样，任选的PLUSPTYPE场存在。只要存在，比特流中就包括一个附加的数据集，该数据集立即紧随PTYPE并且优先于PQUANT。另外，当PLUSPTYPE存在时，在图像头部中CPM和PSBI场前移，以至它们立即出现在PLUSPTYPE之后，而不是定位在PQUANT之后。跟随PLUSPTYPE的附加数据格式在图8中显示。PLUSPTYPE之后在此附加图像头部数据中所有场均为任选的，依赖于在PLUSPTYPE中它们的存在是否被指明。当使用截面结构模式时（见附件K），对图7中所示的GOB位置以截面替代。

PSUPP和PEI的组合可以不存在，并且一旦存在可以重复。EOS和EOSBS+ESBI可以不存在，而仅当EOS或EOSBS存在时，ESTUF才可以存在。EOS不应被重复，除非每对EOS码之间至少出现一个图像起始码。下线的图像其图像头部不传输。

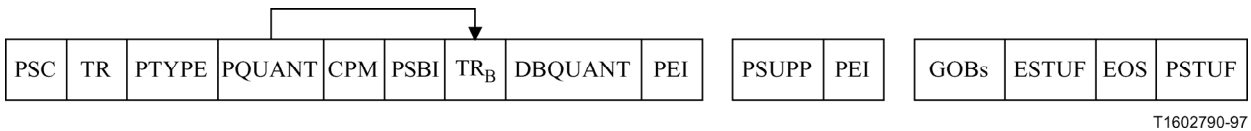


图 7/H.263—图像层结构（无任选的PLUSPTYPE相关场）

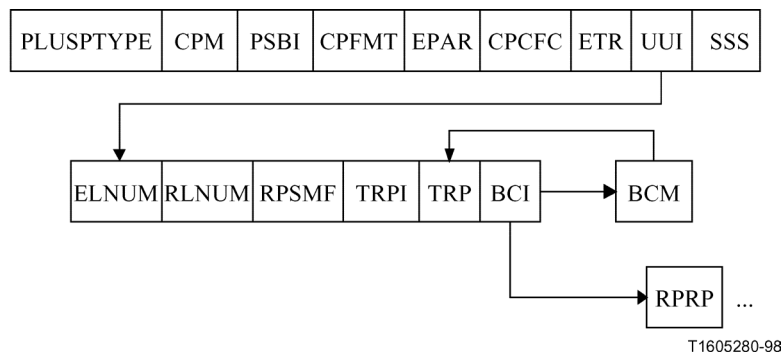


图 8/H.263—任选的PLUSPTYPE相关场结构（只要存在立即位于PTYPE之后）

5.1.1 图像起始码（PSC）（22比特）

PSC是22比特字。它的值为0000 0000 0000 0000 1 00000。所有图像起始码应字节定位。如有必要，这可通过起始码前插入PSTUF实现，以至起始码的首比特是字节的第一（最高有效）比特。

5.1.2 时域参考（TR）（8比特）

时域参考（TR）值通过增加其在时域过去参考图像头部中的值组成，即自以前传输图像以来以图像时钟频率跳过的或非参考图像的数目加1。TR的解释依赖于有效图像时钟频率。在标准CIF图像时钟频率下，TR为8比特数，可以有256个可能值。实施运算仅取8个LSB。若标示惯例图像时钟频率在使用，则5.1.8中扩展的TR和TR_B构成10比特数，其中TR_B存储八个最低有效比特（LSB）及ETR存储两个最高有效比特（MSB）。在此情形中，实施运算取10个LSB。在任选的PB帧或改进的PB帧模式中，TR仅设址P图像；对PB帧或改进的PB帧的B图像部分的时域参考，参见5.1.22。

5.1.3 类型信息 (PTYPE) (可变长度)

有关完整图像的信息:

- 比特1: 总是“1”, 为避免起始码仿真。
- 比特2: 总是“0”, 为同ITU-T H.261建议书区分。
- 比特3: 分屏指示符, “0”断, “1”通。
- 比特4: 文件摄像指示符, “0”断, “1”通。
- 比特5: 全图像冻结释放, “0”断, “1”通。
- 比特6-8: 源格式, “000”禁用, “001”子QCIF, “010”QCIF, “011”CIF, “100”4CIF, “101”16CIF, “110”保留, “111”扩展的PTYPE。

若比特6-8不等于指示扩展的PTYPE (PLUSPTYPE) 的“111”, 则下列五个比特在PTYPE中也存在:

- 比特9: 图像编码类型, “0”INTRA (I图像), “1”INTER (P图像)。
- 比特10: 任意的非受限运动矢量模式 (见附件D), “0”断, “1”通。
- 比特11: 任意的基于句法的算术编码模式 (见附件E), “0”断, “1”通。
- 比特12: 任意的先进的预测模式 (见附件F), “0”断, “1”通。
- 比特13: 任意的PB帧模式 (见附件G), “0”正常的I或P图像, “1”PB帧。

分屏指示符是指示译码的图像上、下半屏能够并排显示的信号。这个比特对图像的编码或译码没有任何直接影响。

全图像冻结释放是来自编码器的一个信号, 它应答分组转发请求 (即使不确认) 或快速更新请求 (亦见附件C), 或图像冻结请求 (亦见附件L), 以及允许译码器从其冻结图像模式退出且以正常方式显示译码的图像。

若比特6-8指示与先前图像帧头不同的源格式, 那么当前图像应是一个I图像, 除非比特6-8指示扩展的PTYPE并且使用任意的参考图像再采样模式 (见附件P) 的能力已经外部协商 (例如ITU-T H.245建议书)。

比特10-13涉及任意的模式, 仅用于编译器之间协商以后 (亦分别见附件D、E、F和G)。若比特9置“0”, 比特13同样亦应置“0”。

比特6-8不应具有指示扩展的PTYPE (PLUSPTYPE) 存在的值“111”, 除非能力已经外部协商 (例如ITU-T H.245建议书), 只有通过使用扩展的PTYPE才允许使用惯例源格式或一个或多个可用的其他的任选模式 (见附件I到K及附件M到T)。每当比特6-8不具有值“111”时, 只有通过使用扩展的PTYPE才可用的所有的附加模式均将认为已设置到一个“断”状态, 并且应推断保持“断”状态, 除非稍后在比特流中明确切换到“通”状态。

5.1.4 另加的PTYPE (PLUSPTYPE) (可变长度)

仅当在TPYPE的比特6-8中指示扩展的PTYPE时，12或30比特字才存在。

PLUSPTYPE由最多三个子场UFEP、OPPTYPE和MPPTYPE组成。OPPTYPE仅当UFEP有特殊值时才存在。

5.1.4.1 更新全扩展的PTYPE (UFEP) (3比特)

仅当在TPYPE的比特6-8中指示“扩展的PTYPE”时，3比特的固定长度码字才存在。当设置为“000”时，它指示包括在当前图像头部中那些在每个图像头部中均需要标示的扩展PTYPE场(MPPTYPE)。当设置为“001”时，它指示包括在当前图像头部中的所有扩展PTYPE场。若图像类型为INTRA或EI，则此场应设置为“001”。

另外，若PLUSPTYPE在每个连续图像序列中存在，则此场应设置为“001”，设置周期至少五秒或五幅图像所指定的暂停周期，无论那种情况，允许较大的时间间隔。更准确的说，暂停周期要求UFEP=“001”在第一幅图像头部的PLUSPTYPE场中出现（若PLUSPTYPE在每个插入图像中存在的话），该图像具有时域参考指示时间间隔自最后发生UFEP=“001”以来大于或等于五秒，或者最后发生UFEP=“001”之后的第五幅图像的PLUSPTYPE场中出现。（无论哪种要求，允许时域参考所测量的较大的时间周期。）

在误差易出环境中，编码器将更为经常的设置UFEP为“001”。与“000”和“001”不同的UFEP值被保留。

5.1.4.2 PLUSPTYPE的任选部分 (OPPTYPE) (18比特)

若UFEP为“001”，则以下比特在PLUSPTYPE中存在：

- 比特1-3 源格式，“000”保留，“001”子QCIF，“010”QCIF，“011”CIF，“100”4CIF，“101”16CIF，“110”惯例源格式，“111”保留的；
- 比特4 任选的惯例PCF，“0”CIF PCF，“1”惯例PCF；
- 比特5 任选的非受限运动矢量(UMV)模式(见附件D)，“0”断，“1”通；
- 比特6 任选的基于句法的算术编码(SAC)模式(见附件E)，“0”断，“1”通；
- 比特7 任选的先进的预测(AP)模式(见附件F)，“0”断，“1”通；
- 比特8 任选的先进的INTRA编码(AIC)模式(见附件I)，“0”断，“1”通；
- 比特9 任选的分块滤波器(DF)模式(见附件J)，“0”断，“1”通；
- 比特10 任选的截面结构(SS)模式(见附件K)，“0”断，“1”通；
- 比特11 任选的参考图像选择(RPS)模式(见附件N)，“0”断，“1”通；
- 比特12 任选的独立分段译码(ISD)模式(见附件R)，“0”断，“1”通；
- 比特13 任选的可选择的INTER VLC(AIV)模式(见附件S)，“0”断，“1”通；
- 比特14 任选的修正的量化(MQ)模式(见附件T)，“0”断，“1”通；

- 比特15 等于“1”防止起始码仿真；
- 比特16 保留的，应等于“0”；
- 比特17 保留的，应等于“0”；
- 比特18 保留的，应等于“0”。

5.1.4.3 PLUSPTYPE存在时，PLUSPTYPE的强制部分（MPPTYPE）（9比特）

不管UFEP的值，以下9比特总应在PLUSPTYPE中存在：

- 比特1-3 图像类型代码：
 - “000” I图像（INTRA），
 - “001” P图像（INTER），
 - “010” 改进的PB帧（见附件M），
 - “011” B图像（见附件O），
 - “100” EI图像（见附件O），
 - “101” EP图像（见附件O），
 - “110” 保留的，
 - “111” 保留的；
- 比特4 任选的参考图像再采样（RPR）模式（见附件P），“0”断，“1”通；
- 比特5 任选的简约分辨率更新（RRU）模式（见附件Q），“0”断，“1”通；
- 比特6 舍入类型（RTYPE）（见6.1.2）；
- 比特7 保留的，应等于“0”；
- 比特8 保留的，应等于“0”；
- 比特9 等于“1”防止起始码仿真；

编码器将控制舍入类型以至P图像、改进的PB帧及EP图像具有同供其运动补偿的参考图像不同的比特6（P图像的舍入类型）的值。若参考图像为I或EI图像，则比特6可以取任何值。仅当比特1-3指示P图像、改进的PB帧或EP图像时，比特6才可以置于“1”。对于其他类型的图像，此比特应总是置于“0”。

5.1.4.4 PLUSPTYPE的语义

PLUSPTYPE的强制部分由可能伴随逐个图像变化的特征组成。根本上讲，这些特征是指示在I、P、改进的PB、B、EI和EP之间图像类型的比特。（注意附件G的PB帧模式不能同PLUSPTYPE存在一起使用—附件M的改进的PB帧模式将替代使用。）然而，它们也包括使用RPR和RRU模式的指示，同样作为指示这些比特可以从图像到图像变化。

从一个图像到另一个图像可能继续使用而不改变的特征（5.1.4.5中所讨论的那些显而易见的方式除外），将放置在PLUSPTYPE的任选部分。当UFEP为000时，从先前的具有UFEP=001的PLUSPTYPE中的图像类型和所发送的模式信息中可以推断丢失的模式信息。

若PLUSPTYPE存在，而UFEP为000，那么：

- 1) 对P图像和改进的PB帧而言（见附件M），像素宽高比、图像宽度及图像高度与参考图像的那些参数相比不变化。

- 2) 对增强层中时域可量测性B图像而言（见附件O），只要增强层内所发送的最后图像是EI或EP图像，则参考层编号（RLNUM）应与增强层编号（ELNUM）相同。若增强层内所发送的最后图像是B图像，则参考层编号与最后B图像的参考层编号应相同。像素宽高比、图像宽度及图像高度与时域后续参考层图像的那些参数相比不变化。

注意若在与B图像相同的增强层内，时域附近EI或EP图像存在。则RLNUM（明确的或隐含的）总应等于ELNUM。也要注意B图像的像素宽高比、图像宽度及图像高度（明确的或隐含的）总应等于其时域后续参考层图像的那些参数。

- 3) 对SNR/空间可量测性EP图像而言（见附件O），像素宽高比、图像宽度及图像高度与同一增强层内的时域先前参考图像的那些参数相比不变化。

5.1.4.5 某些图像类型的模式限制和模式推理准则

某些模式不适用于某些图像类型。特别的，以下这些限制适用：

- 1) 在I（INTRA）图像内，下列模式不适用：非受限的运动矢量（见附件D）、先进的预测（见附件F）、可选择的INTER VLC（见附件S）以及简约分辨率更新（见附件Q）。
- 2) 在B图像内（见附件O），下列模式不适用：基于句法的算术编码（见附件E）、分块滤波器（见附件J）以及先进的预测（见附件F）。
- 3) 在EI图像内（见附件O），下列模式不适用：非受限的运动矢量（见附件D）、基于句法的算术编码（见附件E）、先进的预测（见附件F）、参考图像再采样（见附件P）、简约分辨率更新（见附件Q）以及可选择的INTER VLC（见附件S）。
- 4) 在EP图像内（见附件O），下列模式不适用：基于句法的算术编码（见附件E）以及先进的预测（见附件F）。

在以上四个条款目录中列举的一个或多个模式或许在对那个模式（I、B、EI或EP类型）禁止的图像类型内的PLUSPTYPE的任选部分中存在值“1”的模式标记。此种情形可以允许并且应被理解为假定遵循下一段落中的模式推理准则。

模式状态从属于下列模式推理准则：

- 1) 在PLUSPTYPE的任选部分中，一旦模式标记已经置于“1”，则比特流中当前图像和每个随后的图像应对那个模式指派“通”状态。
- 2) 推测的“断”状态应指派给这样的模式，它不适用于具有当前图像类型码的图像。然而对那个模式而言，比特流中每个随后的图像应有推测的“通”状态（除非这将导致非常明显的冲突——应以相同的方式来解决）。在分层可标度比特流情形中（见附件O），模式状态应仅从该比特流的同一层内推断。
- 3) 状态的推理应继续直至同一层内或者包含PLUSPTYPE 任选部分或者一点不包含PLUSPTYPE的图像被发送时为止。若发送包含PLUSPTYPE 任选部分的新图像，则在新消息中发送的状态应压倒老的状态。若发送不包括PLUSPTYPE的图像（PTYPE的比特6-8不是“111”的图像），则“断”状态应指派给在PTYPE场中没有明确设置为“通”的所有模式，并且所有模式应连续具有推断的“断”状态直至包含PLUSPTYPE 任选部分的新图像被发送时为止。

- 4) 两种模式不需要模式状态推理，由于这些模式的模式标记在PLUSPTYPE的强制部分中出现。这些模式是参考图像再采样模式（附件P）和简约分辨率更新模式（附件Q）。这些模式的每一个模式标记不应发送，除非当前图像允许使用该模式。例如简约分辨率更新模式比特不应在INTRA图像中发送。

5.1.4.6 模式交互限制

某些模式不能用于同一些其他模式结合。

- 1) 基于句法的算术编码模式（见附件E）不应同可选择的INTER VLC模式（见附件S）或修正的量化模式（见附件T）一起使用。
- 2) 若PLUSPTYPE存在，非受限运动矢量（见附件D）不应同基于句法的算术编码模式（见附件E）一起使用。
- 3) 独立分段译码模式（见附件R）不应同图像再采样模式（见附件P）一起使用。
- 4) 如果没有截面结构模式的矩形截面子模式（见附件K）的同时使用，独立分段译码模式（见附件R）不应同截面结构模式一起使用。

5.1.4.7 图像头部CPM（1比特）和PSBI（2比特）的定位

在图像头部CPM和PSBI场的定位依赖于PLUSPTYPE是否存在（见5.1.20和5.1.21）。若PLUSPTYPE存在，那么图像头部中PLUSPTYPE之后CPM立即紧随，若PLUSPTYPE不存在，那么图像头部中PQUANT之后CPM立即紧随。PSBI总是立即紧随CPM（只要CPM=“1”）。

5.1.5 惯例图像格式（CPFMT）（23比特）

仅当惯例图像格式的使用在PLUSPTYPE中标示且UFEP为“001”时，23比特的固定长度码字才存在。只要存在，CPFMT构成如下：

- 比特1-4 像素宽高比代码：表5中4比特字指示出PAR值。对于扩展的PAR而言，精确的像素宽高比应在EPAR中指明（见5.1.6）；
- 比特5-13 图像宽度指示：范围[0, ..., 511]，每行像素数 = (PWI+1) * 4；
- 比特14 等于“1”防止起始码仿真；
- 比特15-23 图像高度指示：范围[1, ..., 288]，行的数目 = PHI * 4；

表 5/H.263—PAR代码定义

| PAR代码 | 像素宽高比 |
|-----------|-------------------------------|
| 0000 | 禁用 |
| 0001 | 1 : 1 (正方形) |
| 0010 | 12 : 11 (4 : 3 的 CIF 图像) |
| 0011 | 10 : 11 (4 : 3 的 525 类型图像) |
| 0100 | 16 : 11 (16 : 9 的 CIF 展宽图像) |
| 0101 | 40 : 33 (16 : 9 的 525 类型展宽图像) |
| 0110-1110 | 保留的 |
| 1111 | 扩展的 PAR |

5.1.6 扩展的像素宽高比 (EPAR) (16比特)

仅当CPFMF存在且这里指示扩展的PAR时，16比特的固定长度码字才存在。只要存在，EPAR构成如下：

- 比特1-8 PAR宽度：“0”为禁用。该PAR宽度的自然二进制表示；
- 比特9-16 PAR高度：“0”为禁用。该PAR高度的自然二进制表示。

PAR宽度和PAR高度应互素。

5.1.7 惯例图像时钟频率代码 (CPCFC) (8比特)

仅当PLUSPTYPE存在且UFEP为“001”以及惯例图像时钟频率在PLUSPTYPE中标示时，8比特的固定长度码字才存在。只要存在，CPCFC构成如下：

- 比特1 时钟变换码：“0”指示时钟变换因子为1000及“1”指示为1001；
- 比特2-8 时钟因子：“0”为禁用。该时钟因子的自然二进制表示。

惯例图像时钟频率由 $1\ 800\ 000 / (\text{时钟参数} \times \text{时钟变换因子})$ Hz给出。

时域参考计数器应以该图像时钟频率的逆，以秒为单位计数。当PCF与对先前图像所指派的那个PCF相比变化时，当前图像的时域参考按照先前的PCF度量，以至于新的PCF仅对未来图像的时域参考度量生效。

5.1.8 扩展的时域参考 (ETR) (2比特)

仅当惯例图像时钟频率在使用时（不管UFEP的值），2比特的固定长度码字才存在。它是5.1.2中定义的10比特数的两个MSB。

5.1.9 无限制的非受限运动矢量指示符 (UII) (可变长度)

仅当任意的非受限运动矢量模式在PLUSPTYPE中指示且UFEP为001时，1或2比特的可变长度码字才存在。当UII存在时，它指示正在使用的运动矢量范围的有效界限。

- UII=“1”运动矢量范围依照表D.1和D.2受限。
- UII=“01”运动矢量范围不受限，除由图像尺寸限制外。

5.1.10 截面结构的子模式比特 (SSS) (2比特)

仅当任选的截面结构模式（见附件K）在PLUSPTYPE中指示且UFEP为001时，2比特的固定长度码字才存在。若截面结构模式在使用而UFEP不为001，则SSS的最后发送值应保持有效。

- 比特1 矩形截面，“0”指示自由运行截面，“1”指示矩形截面；
- 比特2 任意截面排序，“0”指示顺序次序，“1”指示任意次序。

5.1.11 增强层编号 (ELNUM) (4比特)

仅当任选的时域的、SNR及空间可量测性模式在使用时（不管UFEP的值），4比特的固定长度码字才存在。特殊的增强层通过增强层编号ELNUM标识。层间的图像对应经由时域参考完成。每个增强层内的图像尺寸或者使用存在的源格式场指示，或者通过同参考层的关系推断。基础层之上的第一个增强层被指派为增强层编号2，而基础层为编号1。

5.1.12 参考层编号 (RLNUM) (4比特)

仅当任选的时域的、SNR及空间可量测性模式在使用（见附件D）且UFEP为001时，4比特的固定长度码字才存在。充当参考参照物所使用的图像层编号通过参考层编号 (RLNUM) 来标识。层间时间对应经由时域参考完成。

注意若在与B图像相同的增强层内，时域附近EI或EP图像存在，则RLNUM应等于ELNUM（见附件O）。

5.1.13 参考图像选择模式标记 (RPSMF) (3比特)

仅当参考图像选择模式在使用且UFEP为001时，3比特的固定长度码字才存在，只要存在，RPSMF指示那种类型的反向信道消息被该编码器所必需。若参考图像选择模式在使用而RPSMF不存在，则发送的RPSMF的最后值保持有效。

- 100: 即不需要ACK也不需要NACK信号；
- 101: 需要ACK信号返回；
- 110: 需要NACK信号返回；
- 111: 需要ACK和NACK信号返回；
- 000-011: 保留的。

5.1.14 预测的时域参考指示 (TRPI) (1比特)

仅当任选的参考图像选择模式在使用时（不管UFEP的值），1比特的固定长度码字才存在。只要存在，TRPI指示以下TRP场的存在：

- 0: TRP场不存在；
- 1: TRP场存在。

每当图像头部指示I或EI图像时，TRPI应为0。

5.1.15 预测的时域参考 (TRP) (10比特)

只要存在 (如TRPI中指示的), TRP指示供编码预测所使用的时域参考, B图像的情形除外。对B图像而言, 具有时域参考TRP的图像供前向预测使用。(逆时间方向预测总量使用即时时域后续图像。) TRP为10比特数。对参考图像而言, 若惯例图像时钟频率未使用, 则TRP的两个MSB为0且LSB包含参考图像的图像头部中所出现的8比特TR。若参考图像的惯例图像时钟频率在使用, 则TRP是10比特数由来自该参考图像头部的联系在一起的ETR和TR组成。

当TRP不存在时, 最近的时域先前参照物图像应供预测使用, 如不在参考图像选择模式中那样。下一个PSC、GSC或SSC出现之前TRP一直有效。

5.1.16 后向信道消息指示 (BCI) (可变长度)

仅当任选的参考图像选择模式在使用时, 1或2比特的可变长度场才存在。当置于“1”, 此比特标示以下任选的视频后向信道消息 (BCM) 场的存在。“01”指示视频后向信道消息场缺席或结束。BCM和BCI的结合可以不存在, 并且只要存在就可以重复。若任选的参考图像选择模式的视频最大子模式未在使用, 则BCI应置于“01”。

5.1.17 后向信道消息 (BCM) (可变长度)

仅当先导的BCI场存在且置于“1”时, 具有N.4.2中所指示句法的后向信道消息才存在。

5.1.18 参考图像再采样参数 (RPRP) (可变长度)

仅当任选的参考图像再采样模式比特在PLUSPTYPE中设置时, 可变长度场才存在。该场携带参考图像再采样模式的参数 (见附件P)。注意参考图像再采样模式也能够由INTER编码图像的图像头部出现来隐含要求, 只要该INTER图像具有不同于先前编码图像的图像尺寸, 在此情形RPRP场不存在并且参考图像再采样模式比特不设置。

5.1.19 量化器信息 (PQUANT) (5比特)

5比特固定长度码字指示供该图像使用的量化器直至其被任何随后的GQUANT或DQUANT更新时为止。码字为QUANT值的自然二进制表示, 具有半个步长尺寸, 范围从1到31。

5.1.20 连续出现多点和视频多路复用 (CPM) (1比特)

1比特的码字标示任选的连续出现多点和视频多路复用 (CPM) 的使用; “0”断, “1”通。对于CPM的使用, 参阅附件C。若PLUSPTYPE存在, 则CPM紧随PLUSPTYPE之后, 若PLUSPTYPE不存在, 则跟随图像头部中的PQUANT。

5.1.21 图像子比特流指示符 (PSBI) (2比特)

为2比特的固定长码字, 仅当连续出现多点和视频多路复用模式被CPM指示时, 它才存在。对于图像帧头和所有跟随的直至下一图像帧或GOB起始码的信息而言, 码字是子比特流编号的自然二进制表示 (亦见附件C)。若CPM为“1”, PSBI紧随CPM之后 (图像帧头中CPM和PSBI的位置依赖于PLUSPTYPE是否存在)。

5.1.22 PB帧中B图像的时域参考 (TR_B) (3/5比特)

若PTYPE或PLUSPTYPE指示“PB帧”或“改进的PB帧”，则TR_B存在（亦见附件G和M），它指示自最后的P或I图像或者PB帧或改进的PB帧的P图像部分以来和PB帧或改进的PB帧的B图像部分之前所有非传输或非参考图像的数目（速率29.97 Hz或在CPCFC中指示的惯例图像时钟频率）。码字为非传输图像的数目加1的自然二进制表示。对标准CIF图像时钟频率它为3比特长并且当惯例图像时钟频率在使用时扩展为5比特。对标准CIF图像时钟频率非传输图像的最大数目为6，当惯例图像时钟频率在使用时为30。

5.1.23 PB帧中B图像的量化信息 (DBQUANT) (2比特)

若PTYPE或PLUSPTYPE指示PB帧或改进的PB帧，则DBQUANT存在（亦见附件G和M）。在译码处理中，对每个宏块获取量化参数QUANT。采用PB帧QUANT用于P块，同时对B块使用不同的量化参数BQUANT。QUANT取值范围1到31。DBQUANT指示QUANT和BQUANT之间的关系，如表6中规定。此表中，“/”代表舍位除法。BQUANT取值范围1到31；若从表6产生的BQUANT的值大于31，则它被限幅到31。

表 6/H.263—DBQUANT代码和QUANT与BQUANT间的关系

| DBQUANT | BQUANT |
|---------|-------------------------------|
| 00 | $(5 \times \text{QUANT}) / 4$ |
| 01 | $(6 \times \text{QUANT}) / 4$ |
| 10 | $(7 \times \text{QUANT}) / 4$ |
| 11 | $(8 \times \text{QUANT}) / 4$ |

5.1.24 额外插入信息 (PEI) (1比特)

为一个比特，置“1”标示下述任选数据场的存在。

5.1.25 增补的增强信息 (PSUPP) (0/8/16……比特)

若PEI置“1”，那么其后跟随9比特，其中前8比特为数据 (PSUPP)，第9比特为另一个PEI比特，它指示其后是否跟随进一步的9比特，如此继续。编码器应如附件L中所指出的那样使用PSUPP。不支持附件L中描述的扩展能力的译码器应设计为在PEI置“1”时丢弃PSUPP。这将使附件L的扩展能力后向兼容以至于利用该扩展能力的比特流也能够被不支持那些能力的译码器无须改动而使用。

5.1.26 填充 (ESTUF) (可变长度)

由少于8个零比特组成的可变长码字。编码器可在EOS码字之前直接插入此码字。若必要，编码器应插入该码字以实现EOSBS码字之前的直接强制字节定位。若ESTUF存在，则ESTUF的最后比特应是字节的最后（最低有效）比特，以致EOS或EOSBS码字的起始被字节定位。译码器应设计为丢弃ESTUF。对EOSBS及其使用的描述见附件C。

5.1.27 序列结束 (EOS) (22比特)

22比特码字。它的值是0000 0000 0000 0000 1 11111。编码器决定是否插入此码字。EOS可以被字节定位。这可通过EOS码之前插入ESTUF来实现，以致EOS码的首比特是字节的第一（最高有效）比特。EOS不应重复除非每对EOS代码之间至少出现一个图像起始码。

5.1.28 填充 (PSTUF) (可变长度)

由少于8个零比特组成的可变长码字。编码器插入此码字可供下一个PSC的字节定位。PSTUF的最后比特应是字节的最后（最低有效）比特，以致H.263比特流中包括PSTUF的视频比特流从第一比特起是8比特的一个倍数。译码器应设计丢弃PSTUF。

若为了某种理由，在某个确定的时间周期编码器停止编码图像并在以后恢复编码，则编码器停止之前应传输PSTUF，以防止先前图像的最后的最后最多7比特未被发送，直到编码器恢复编码时为止。

5.2 块组层

每个块组 (GOB) 数据由GOB头和跟随其后的宏块数据组成。结构显示在图9。每个GOB包含一行或多行宏块。对于每个图像帧中的第一个GOB (具有编号0)，不传输任何GOB头。对所有其他的GOB，GOB头可以是空的，取决于编码器策略。通过外部手段，例如ITU-T H.245建议书，译码器能够标示远端编码器仅传输非空的GOB头。当GBSC存在时，GSTUF可以存在。当GBSC存在时，GN、GFID 和GQUANT存在。在图像头部中当CPM为1时，GSBI存在。

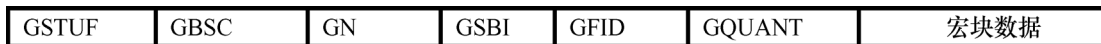


图 9/H.263—GOB层结构

5.2.1 填充 (GSTUF) (可变长度)

由少于8个零比特组成的可变长码字。在GBSC码字之前，编码器可直接插入此码字。若GSTUF存在，GSTUF的最后比特应是字节的最后（最低有效）比特，以致GBSC码字的起始被字节定位。译码器应设计为丢弃GSTUF。

5.2.2 块组起始码 (GBSC) (17比特)

为一个17比特字。它的值是0000 0000 0000 0000 1。GOB起始码可以字节定位。通过起始码前插入GSTUF能够实现字节定位以致起始码的首比特是字节的第一（最高有效）比特。

5.2.3 块组编号 (GN) (5比特)

为5比特固定长码字。该比特是块组编号的二进制表示。对编号0的GOB，包括GSTUF、GBSC、GN、GSBI、GFID和GQUANT的GOB头是空的；作为组编号0用于PSC中。组编号1到17用在标准图像格式的GOB头部中，组编号1到24用在惯例图像格式的GOB头部中，当CPM=“0”时组编号16到28在截面格式头部中使用（见附件K），当CPM=“1”时组编号25到27及29在截面格式头部中使用（见附件K），组编号31用在EOS代码中，组编号30用在EOSBS代码中。

5.2.4 GOB子比特流指示符 (GSBI) (2比特)

在图像头部仅当CPM为1, 2比特的固定长码字才存在。对GOB头和所有跟随的直至下一个图像或GOB起始码为止的信息而言, 码字是子比特流编号的自然二进制表示 (亦见附件C)。

5.2.5 GOB帧ID (GFID) (2比特)

为2比特的固定长码字。在给定图像的每个GOB (或截面) 头部中, GFID应有相同的值。更进一步, 如图像头部中所指示的若PTYPE同先前传输图像的PTYPE相同, 那么GFID应有该先前图像中的同样值, 假设PLUSPTYPE不存在。然而, 若某图像头部中的PTYPE不同于先前传输图像头部中的PTYPE, 则该图像中的GFID值也应与先前图像的GFID值不同。

若PLUSPTYPE存在, 则GFID的值应同先前图像 (在同一层内) 的该值相同, 就先前图像而言, 只要该PTYPE及PLUSPTYPE, 和CPFMT、EPAP、CPCFC、SSS、ELNUM、RLNUM、UII、RPSMF和RPRP之间的所有现存场继续有效; 否则, GFID应与先前图像的该值不同。

5.2.6 量化器信息 (GQUANT) (5比特)

为5比特的固定长码字, 指示供图像剩余部分所使用的量化器QUANT直至被任何后续的GQUANT或DQUANT更新时为止。码字是QUANT值的自然二进制表示, 具有半步长尺寸, 范围为1到31。

5.3 宏块层

每个宏块数据由宏块头和跟随的块数据组成。结构显示在图10。对这些图像的每个宏块而言, 在不是“INTRA”模式类型的图像中, COD才存在。当用COD指示或者当该图像为“INTRA”类型时, MCBPC存在。若PTYPE指示PB帧, 对宏块类型0-4, MODB存在。当用MCBPC指示时, CBPY、DQUANT、MVD和MVD₂₋₄存在。仅当由MODB指示时, CBPB和MVDB才存在。当用MCBPC和CBPY指示时, 块数据存在。MVD₂₋₄仅在先进预测模式 (参见附件F) 或分块滤波器模式 (参见附件J) 中存在。MODB、CBPB和MVDB仅在PB帧模式 (参见附件G) 中存在。在基于句法的算术编码模式中, 对于符号的编码参见附件E。对B、EI及EP图像中宏块层的编码见附件O。

| | | | | | | | | | | | |
|-----|-------|------|------|------|--------|-----|------------------|------------------|------------------|------|-----|
| COD | MCBPC | MODB | CBPB | CBPY | DQUANT | MVD | MVD ₂ | MVD ₃ | MVD ₄ | MVDB | 块数据 |
|-----|-------|------|------|------|--------|-----|------------------|------------------|------------------|------|-----|

图 10/H.263—宏块层结构

5.3.1 编码的宏块指示 (COD) (1比特)

一个比特, 置“0”时该比特标示宏块编码。若置“1”, 对该宏块没有任何进一步的信息传输; 在此情况下, 译码器应把该宏块当作一个INTER宏块处理具有整块的运动矢量等于0并且无任何系数数据。对这些图像中的每个宏块而言, 在不是“INTRA”模式类型的图像中, COD才存在。

注 — 在先进预测模式 (见附件F) 中, 若COD置“1”, 交叠块运动补偿亦实施, 在分块滤波器模式中 (见附件J), 分块滤波器也能够影响具有COD置于“1”的宏块的某些像素值。

5.3.2 色度宏块类型和编码块模式 (MCBPC) (可变长度)

MCBPC是可变长度码字给出有关色度宏块类型和编码块模式的信息。MCBPC的码字在表7和表8中给出。MCBPC总是包括在编码的宏块中。

表 7/H.263—MCBPC的VLC表 (对I图像)

| 索引 | 宏块类型 | CBPC (56) | 比特数目 | 代码 |
|----|------|-----------|------|-------------|
| 0 | 3 | 00 | 1 | 1 |
| 1 | 3 | 01 | 3 | 001 |
| 2 | 3 | 10 | 3 | 010 |
| 3 | 3 | 11 | 3 | 011 |
| 4 | 4 | 00 | 4 | 0001 |
| 5 | 4 | 01 | 6 | 0000 01 |
| 6 | 4 | 10 | 6 | 0000 10 |
| 7 | 4 | 11 | 9 | 0000 11 |
| 8 | 填充 | — | 9 | 0000 0000 1 |

表 8/H.263—MCBPC的VLC表 (对P图像)

| 索引 | 宏块类型 | CBPC (56) | 比特数目 | 代码 |
|----|------|-----------|------|-------------|
| 0 | 0 | 00 | 1 | 1 |
| 1 | 0 | 01 | 4 | 0011 |
| 2 | 0 | 10 | 4 | 0010 |
| 3 | 0 | 11 | 6 | 0001 01 |
| 4 | 1 | 00 | 3 | 011 |
| 5 | 1 | 01 | 7 | 0000 111 |
| 6 | 1 | 10 | 7 | 0000 110 |
| 7 | 1 | 11 | 9 | 0000 0010 1 |
| 8 | 2 | 00 | 3 | 010 |
| 9 | 2 | 01 | 7 | 0000 101 |
| 10 | 2 | 10 | 7 | 0000 100 |
| 11 | 2 | 11 | 8 | 0000 0101 |
| 12 | 3 | 00 | 5 | 0001 1 |
| 13 | 3 | 01 | 8 | 0000 0100 |
| 14 | 3 | 10 | 8 | 0000 0011 |
| 15 | 3 | 11 | 7 | 0000 011 |
| 16 | 4 | 00 | 6 | 0001 00 |
| 17 | 4 | 01 | 9 | 0000 0010 0 |
| 18 | 4 | 10 | 9 | 0000 0001 1 |
| 19 | 4 | 11 | 9 | 0000 0001 0 |

表 8/H.263—MCBPC的VLC表（对P图像）

| 索引 | 宏块类型 | CBPC (56) | 比特数目 | 代码 |
|----|------|-----------|------|------------------|
| 20 | 填充 | — | 9 | 0000 0000 1 |
| 21 | 5 | 00 | 11 | 0000 0000 010 |
| 22 | 5 | 01 | 13 | 0000 0000 0110 0 |
| 23 | 5 | 10 | 13 | 0000 0000 0111 0 |
| 24 | 5 | 11 | 13 | 0000 0000 0111 1 |

对比特填充而言，该表中一个额外的码字是可用的。该码字应被译码器丢弃。若由MPPTYPE比特1-3指示改进的PB帧以及在OPPTYPE比特1-3中指示惯例源格式，那么在该图像的第一个宏块之前宏块A不应指示填充（为防止起始码仿真）。

注 — 译码器应设计为允许宏块类型在一个图像、GOB或比特流中的截面起始码的定位前立即指示比特填充。然而，在这种方式中译码器不应使用宏块层填充（用于与明确需要对此支持之前已设计完成的译码器间的互用性）。

宏块类型给出有关宏块及其那些数据单元存在的信息。宏块类型及其包括的单元在表9和表10中列举。除非在图像头部存在扩展的PTYPE（PLUSPTYPE）以及或者先进的预测模式（见附件F）或分块滤波器模式（见附件J）在使用，否则宏块类型5（表8中索引号21-24）应不存在，并且对图像的第一个宏块而言宏块类型5应不存在。为了防止起始码仿真，编码器也不应允许宏块类型5的MCBPC代码立即紧随在比特流中的7个连续零（如能够由特殊的INTRADC代码紧随COD=0产生）。当必须实施此项要求时（对图像的非第一宏块而言），宏块类型5代码能够由填充前导。

表 9/H.263—正常图像的宏块类型和所包含的数据元素

| 图像类型 | 宏块类型 | 名称 | COD | MCBPC | CBPY | DQUANT | MVD | MVD ₂₋₄ |
|-------|------|-----------|-----|-------|------|--------|-----|--------------------|
| INTER | 不编码 | — | X | | | | | |
| INTER | 0 | INTER | X | X | X | | X | |
| INTER | 1 | INTER+Q | X | X | X | X | X | |
| INTER | 2 | INTER4V | X | X | X | | X | X |
| INTER | 3 | INTRA | X | X | X | | | |
| INTER | 4 | INTRA+Q | X | X | X | X | | |
| INTER | 5 | INTER4V+Q | X | X | X | X | X | X |
| INTER | 填充 | — | X | X | | | | |
| INTRA | 3 | INTRA | | X | X | | | |
| INTRA | 4 | INTRA+Q | | X | X | X | | |
| INTRA | 填充 | — | | X | | | | |

注 — “X” 代表宏块中此项存在。

表 10/H.263—PB帧的宏块类型和所包含的数据单元

| 图像类型 | 宏块类型 | 名称 | COD | MCBPC | MODB | CBPY | CBPB | DQUANT | MVD | MVDB | MVD ₂₋₄ |
|-------|------|-----------|-----|-------|------|------|------|--------|-----|------|--------------------|
| INTER | 不编码 | — | X | | | | | | | | |
| INTER | 0 | INTER | X | X | X | X | (X) | | X | (X) | |
| INTER | 1 | INTER+Q | X | X | X | X | (X) | X | X | (X) | |
| INTER | 2 | INTER4V | X | X | X | X | (X) | | X | (X) | X |
| INTER | 3 | INTRA | X | X | X | X | (X) | | X | (X) | |
| INTER | 4 | INTRA+Q | X | X | X | X | (X) | X | X | (X) | |
| INTER | 5 | INTER4V+Q | X | X | X | X | (X) | X | X | (X) | X |
| INTER | 填充 | — | X | X | | | | | | | |

注 1 — “X” 代表宏块中此项存在。

注 2 — 若 MODB 指示，则 CBPB 和 MVDB 才存在。

注 3 — 在 INTER 模式，B 块总被编码，即使 PB 宏块的宏块类型指示 INTER。

当至少一个非INTRADC变换系数被传输时（INTRADC是INTRA模式块的DC系数，见5.4.1），除非任选的先进INTRA编码模式在使用，否则色度的编码块模型意味着C_B和/或C_R块。对编码块模型中的CBPC₅和CBPC₆而言，只要块N的任何非INTRADC系数存在，就有CBPC_N=1，否则CBPC_N=0。若先进INTRA编码模式在使用，则用法类似，但指示INTRADC系数的方式与指示其他系数的方式相同（见附件I）。图5中给出块编号。当MCBPC=填充时，宏块层的剩余部分被跳过。在这种情况下，前面的COD=0同任何编码的和非编码的宏块无关，并因此宏块编号不增加。对P图像，通过COD=0和MCBPC=填充的多路设置可实现多路填充。见表7和表8。

5.3.3 B块的宏块模式（MODB）（可变长度）

若PTYPE指示PB帧，对宏块类型0-4，MODB存在，它是一个可变长码字，指示CBPB（指示该宏块的B系数传输）是否存在和/或MVDB是否存在。表11中规定了MODB码字。对改进的PB帧而言，MODB有不同的编码，如附件M所规定的。

表 11/H.263—MODB的VLC表

| 索引 | CBPB | MVDB | 比特数目 | 代码 |
|----|------|------|------|----|
| 0 | | | 1 | 0 |
| 1 | | X | 2 | 10 |
| 2 | X | X | 2 | 11 |

注 — “X” 代表宏块中此项存在。

5.3.4 B块的编码块模型（CBPB）（6比特）

PB帧模式中，若MODB指示，则CBPB才存在。在编码块模型中，对每个比特CBPB_N，若B块N的任何系数存在，则CBPB_N=1，否则CBPB_N=0。图5中给出块编号，CBPB的最远的左比特相应于块编号1。

5.3.5 亮度信号的编码块模型（CBPY）（可变长度）

除非先进的INTRA编码模式在使用，否则对于至少一个非INTRADC变换系数被传输的宏块而言（INTRADC是对INTRA块的DC系数，见5.4.1），给定模型编号的变长码字指示宏块中那些Y块应予传输。若先进的INTRA编码在使用，则指示INTRADC的方式与指示其他系数的方式相同（见附件I）。

对编码块模型中的每个比特 $CBPY_N$ 而言，若块N的任何非INTRADC系数存在，则 $CBPY_N=1$ ，否则 $CBPY_N=0$ 。图5中给出块编号， $CBPY$ 的最远的左比特相应于块编号1。对确定模式 $CBPY_N$ ，不同的码字供INTER和INTRA宏块使用，如表12规定。

表 12/H.263— $CBPY$ 的VLC表

| 索引 | CBPY (INTRA) (12, 34) | CBPY (INTER) (12, 34) | 比特数目 | 代码 |
|----|--------------------------|--------------------------|------|---------|
| 0 | 00 | 11 | 4 | 0011 |
| | 00 | 11 | | |
| 1 | 00 | 11 | 5 | 0010 1 |
| | 01 | 10 | | |
| 2 | 00 | 11 | 5 | 0010 0 |
| | 10 | 01 | | |
| 3 | 00 | 11 | 4 | 1001 |
| | 11 | 00 | | |
| 4 | 01 | 10 | 5 | 0001 1 |
| | 00 | 11 | | |
| 5 | 01 | 10 | 4 | 0111 |
| | 01 | 10 | | |
| 6 | 01 | 10 | 6 | 0000 10 |
| | 10 | 01 | | |
| 7 | 01 | 10 | 4 | 1011 |
| | 11 | 00 | | |
| 8 | 10 | 01 | 5 | 0001 0 |
| | 00 | 11 | | |
| 9 | 10 | 01 | 6 | 0000 11 |
| | 01 | 10 | | |
| 10 | 10 | 01 | 4 | 0101 |
| | 10 | 01 | | |
| 11 | 10 | 01 | 4 | 1010 |
| | 11 | 00 | | |
| 12 | 11 | 00 | 4 | 0100 |
| | 00 | 11 | | |
| 13 | 11 | 00 | 4 | 1000 |
| | 01 | 10 | | |
| 14 | 11 | 00 | 4 | 0110 |
| | 10 | 01 | | |
| 15 | 11 | 00 | 2 | 11 |
| | 11 | 00 | | |

5.3.6 量化器信息 (DQUANT) (2比特/可变长度)

若修正的量化模式未使用，则DQUANT为2比特码来规定QUANT中的变化。表13中给出不同码字的差值。QUANT取值范围从1到31；添加差值后，若QUANT的值小于1或大于31，则被分别限幅到1和31。若修正的量化模式在使用，则DQUANT为可变长度码字，如附件T中所规定的。

表 13/H.263—QUANT的DQUANT码和差值

| 索引 | 差 值 | DQUANT |
|----|-----|--------|
| 0 | - 1 | 00 |
| 1 | - 2 | 01 |
| 2 | 1 | 10 |
| 3 | 2 | 11 |

5.3.7 运动矢量数据 (MVD) (可变长度)

所有的INTER宏块都包括MVD (在PB帧模式中也包括INTRA宏块)，它由水平分量的变长码字后跟垂直分量的变长码字组成。表14中给出变长码。若使用非受限运动矢量模式且PLUSPTYPE存在，则运动矢量使用表D.3替代表14编码 (见附件D)。

表 14/H.263—MVD的VLC表

| 索引 | 向 量 | 差 | 比特数目 | 代 码 |
|----|--------|------|------|------------------|
| 0 | - 16 | 16 | 13 | 0000 0000 0010 1 |
| 1 | - 15.5 | 16.5 | 13 | 0000 0000 0011 1 |
| 2 | - 15 | 17 | 12 | 0000 0000 0101 |
| 3 | - 14.5 | 17.5 | 12 | 0000 0000 0111 |
| 4 | - 14 | 18 | 12 | 0000 0000 1001 |
| 5 | - 13.5 | 18.5 | 12 | 0000 0000 1011 |
| 6 | - 13 | 19 | 12 | 0000 0000 1101 |
| 7 | - 12.5 | 19.5 | 12 | 0000 0000 1111 |
| 8 | - 12 | 20 | 11 | 0000 0001 001 |
| 9 | - 11.5 | 20.5 | 11 | 0000 0001 011 |
| 10 | - 11 | 21 | 11 | 0000 0001 101 |
| 11 | - 10.5 | 21.5 | 11 | 0000 0001 111 |
| 12 | - 10 | 22 | 11 | 0000 0010 001 |
| 13 | - 9.5 | 22.5 | 11 | 0000 0010 011 |
| 14 | - 9 | 23 | 11 | 0000 0010 101 |
| 15 | - 8.5 | 23.5 | 11 | 0000 0010 111 |
| 16 | - 8 | 24 | 11 | 0000 0011 001 |
| 17 | - 7.5 | 24.5 | 11 | 0000 0011 011 |
| 18 | - 7 | 25 | 11 | 0000 0011 101 |
| 19 | - 6.5 | 25.5 | 11 | 0000 0011 111 |

表 14/H.263—MVD的VLC表（续）

| 索引 | 向量 | 差 | 比特数目 | 代码 |
|----|------|-------|------|----------------|
| 20 | -6 | 26 | 11 | 0000 0100 001 |
| 21 | -5.5 | 26.5 | 11 | 0000 0100 011 |
| 22 | -5 | 27 | 10 | 0000 0100 11 |
| 23 | -4.5 | 27.5 | 10 | 0000 0101 01 |
| 24 | -4 | 28 | 10 | 0000 0101 11 |
| 25 | -3.5 | 28.5 | 8 | 0000 0111 |
| 26 | -3 | 29 | 8 | 0000 1001 |
| 27 | -2.5 | 29.5 | 8 | 0000 1011 |
| 28 | -2 | 30 | 7 | 0000 111 |
| 29 | -1.5 | 30.5 | 5 | 0001 1 |
| 30 | -1 | 31 | 4 | 0011 |
| 31 | -0.5 | 31.5 | 3 | 011 |
| 32 | 0 | | 1 | 1 |
| 33 | 0.5 | -31.5 | 3 | 010 |
| 34 | 1 | -31 | 4 | 0010 |
| 35 | 1.5 | -30.5 | 5 | 0001 0 |
| 36 | 2 | -30 | 7 | 0000 110 |
| 37 | 2.5 | -29.5 | 8 | 0000 1010 |
| 38 | 3 | -29 | 8 | 0000 1000 |
| 39 | 3.5 | -28.5 | 8 | 0000 0110 |
| 40 | 4 | -28 | 10 | 0000 0101 10 |
| 41 | 4.5 | -27.5 | 10 | 0000 0101 00 |
| 42 | 5 | -27 | 10 | 0000 0100 10 |
| 43 | 5.5 | -26.5 | 11 | 0000 0100 010 |
| 44 | 6 | -26 | 11 | 0000 0100 000 |
| 45 | 6.5 | -25.5 | 11 | 0000 0011 110 |
| 46 | 7 | -25 | 11 | 0000 0011 100 |
| 47 | 7.5 | -24.5 | 11 | 0000 0011 010 |
| 48 | 8 | -24 | 11 | 0000 0011 000 |
| 49 | 8.5 | -23.5 | 11 | 0000 0010 110 |
| 50 | 9 | -23 | 11 | 0000 0010 100 |
| 51 | 9.5 | -22.5 | 11 | 0000 0010 010 |
| 52 | 10 | -22 | 11 | 0000 0010 000 |
| 53 | 10.5 | -21.5 | 11 | 0000 0001 110 |
| 54 | 11 | -21 | 11 | 0000 0001 100 |
| 55 | 11.5 | -20.5 | 11 | 0000 0001 010 |
| 56 | 12 | -20 | 11 | 0000 0001 000 |
| 57 | 12.5 | -19.5 | 12 | 0000 0000 1110 |

表 14/H.263—MVD的VLC表（续完）

| 索引 | 向量 | 差 | 比特数目 | 代码 |
|----|------|--------|------|------------------|
| 58 | 13 | - 19 | 12 | 0000 0000 1100 |
| 59 | 13.5 | - 18.5 | 12 | 0000 0000 1010 |
| 60 | 14 | - 18 | 12 | 0000 0000 1000 |
| 61 | 14.5 | - 17.5 | 12 | 0000 0000 0110 |
| 62 | 15 | - 17 | 12 | 0000 0000 0100 |
| 63 | 15.5 | - 16.5 | 13 | 0000 0000 0011 0 |

5.3.8 运动矢量数据 (MVD₂₋₄) (可变长度)

若由PTYPE和MCBPC指示, 则包括3个码字MVD₂₋₄, 其中每一个由每个矢量的水平分量的变长码字后跟垂直分量的变长码字组成。表14中给出变长码。仅当在先进的预测模式(参见附件F)或分块滤波器模式(见附件J)中时, MVD₂₋₄才存在。

5.3.9 B宏块的运动矢量数据 (MVDB) (可变长度)

PB帧或改进的PB帧模式中若由MODB指示, 则MVDB才存在, 它由每个矢量的水平分量的变长码字后跟垂直分量的变长码字组成。表14中给出变长码。对于MVDB的使用, 参见附件G和M。

5.4 块层

若不在PB帧模式中, 则宏块由4个亮度块和两色差的各自一个块组成(见图5)。块层结构如图11所示。若MCBPC指示宏块类型3或4, 则该宏块的每个块的INTRADC均存在(见表7和表8)。若由MCBPC或CBPY指示, 则TCOEF存在。

在PB帧模式中, 宏块包括12个块。首先作为在缺省的H.263模式中传输的6个P块数据, 然后是传输的6个B块数据。若MCBPC指示宏块类型3或4, 则宏块的任何P块的INTRADC均存在(见表7和表8)。对B块INTRADC不存在。若由MCBPC或CBPY指示, 则TCOEF对P块存在; 若由CBPB指示, 则TCOEF对B块存在。

在基于句法的算术编码模式中对于符号的编码, 参见附件E。

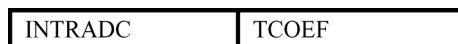


图 11/H.263—块层结构

5.4.1 INTRA块的DC系数 (INTRADC) (8比特)

8比特码字。代码0000 0000不用。代码1000 0000也不用，1024的重构电平编码为1111 1111 (见表15)。

表 15/H.263—INTRA模式DC系数的重构电平

| 索引 | FLC | 进入反变换的重构电平 |
|-----|-----------------|------------|
| 0 | 0000 0001 (1) | 8 |
| 1 | 0000 0010 (2) | 16 |
| 2 | 0000 0011 (3) | 24 |
| ... | ... | ... |
| 126 | 0111 1111 (127) | 1016 |
| 127 | 1111 1111 (255) | 1024 |
| 128 | 1000 0001 (129) | 1032 |
| ... | ... | ... |
| 252 | 1111 1101 (253) | 2024 |
| 253 | 1111 1110 (254) | 2032 |

5.4.2 变换系数 (TCOEF) (可变长度)

最经常发生的EVENT使用表16给出的变长码编码。最后比特“S”代表电平符号，“0”正“1”负。

EVENT是最后非零系数指示 (LAST; “0”：此块中存在较多非零系数，“1”：这是此块中最后的非零系数)、编码系数前面的连续零个数 (RUN)，以及编码系数的非零值 (LEVEL) 的组合。

(LAST, RUN, LEVEL) 的剩余组合用22比特字编码，它由7比特ESCAPE、1比特LAST、6比特RUN和8比特的LEVEL组成。对表16中所列的编码组合不禁止使用这22比特字。对LEVEL的8比特字，代码0000 0000禁用，除非修正的量化模式在使用 (见附件T)，否则代码1000 0000也禁用。表17中给出RUN和LEVEL代码。

表 16/H.263—TCOEF的VLC表

| 索引 | LAST | RUN | LEVEL | 比特数目 | VLC代码 |
|----|------|-----|-------|------|----------------|
| 0 | 0 | 0 | 1 | 3 | 10s |
| 1 | 0 | 0 | 2 | 5 | 1111s |
| 2 | 0 | 0 | 3 | 7 | 0101 01s |
| 3 | 0 | 0 | 4 | 8 | 0010 111s |
| 4 | 0 | 0 | 5 | 9 | 0001 1111s |
| 5 | 0 | 0 | 6 | 10 | 0001 0010 1s |
| 6 | 0 | 0 | 7 | 10 | 0001 0010 0s |
| 7 | 0 | 0 | 8 | 11 | 0000 1000 01s |
| 8 | 0 | 0 | 9 | 11 | 0000 1000 00s |
| 9 | 0 | 0 | 10 | 12 | 0000 0000 111s |

表 16/H.263—TCOEF的VLC表（续）

| 索引 | LAST | RUN | LEVEL | 比特数目 | VLC代码 |
|----|------|-----|-------|------|-----------------|
| 10 | 0 | 0 | 11 | 12 | 0000 0000 110s |
| 11 | 0 | 0 | 12 | 12 | 0000 0100 000s |
| 12 | 0 | 1 | 1 | 4 | 110s |
| 13 | 0 | 1 | 2 | 7 | 0101 00s |
| 14 | 0 | 1 | 3 | 9 | 0001 1110s |
| 15 | 0 | 1 | 4 | 11 | 0000 0011 11s |
| 16 | 0 | 1 | 5 | 12 | 0000 0100 001s |
| 17 | 0 | 1 | 6 | 13 | 0000 0101 0000s |
| 18 | 0 | 2 | 1 | 5 | 1110s |
| 19 | 0 | 2 | 2 | 9 | 0001 1101s |
| 20 | 0 | 2 | 3 | 11 | 0000 0011 10s |
| 21 | 0 | 2 | 4 | 13 | 0000 0101 0001s |
| 22 | 0 | 3 | 1 | 6 | 0110 1s |
| 23 | 0 | 3 | 2 | 10 | 0001 0001 1s |
| 24 | 0 | 3 | 3 | 11 | 0000 0011 01s |
| 25 | 0 | 4 | 1 | 6 | 0110 0s |
| 26 | 0 | 4 | 2 | 10 | 0001 0001 0s |
| 27 | 0 | 4 | 3 | 13 | 0000 0101 0010s |
| 28 | 0 | 5 | 1 | 6 | 0101 1s |
| 29 | 0 | 5 | 2 | 11 | 0000 0011 00s |
| 30 | 0 | 5 | 3 | 13 | 0000 0101 0011s |
| 31 | 0 | 6 | 1 | 7 | 0100 11s |
| 32 | 0 | 6 | 2 | 11 | 0000 0010 11s |
| 33 | 0 | 6 | 3 | 13 | 0000 0101 0100s |
| 34 | 0 | 7 | 1 | 7 | 0100 10s |
| 35 | 0 | 7 | 2 | 11 | 0000 0010 10s |
| 36 | 0 | 8 | 1 | 7 | 0100 01s |
| 37 | 0 | 8 | 2 | 11 | 0000 0010 01s |
| 38 | 0 | 9 | 1 | 7 | 0100 00s |
| 39 | 0 | 9 | 2 | 11 | 0000 0010 00s |
| 40 | 0 | 10 | 1 | 8 | 0010 110s |
| 41 | 0 | 10 | 2 | 13 | 0000 0101 0101s |
| 42 | 0 | 11 | 1 | 8 | 0010 101s |
| 43 | 0 | 12 | 1 | 8 | 0010 100s |
| 44 | 0 | 13 | 1 | 9 | 0001 1100s |
| 45 | 0 | 14 | 1 | 9 | 0001 1011s |
| 46 | 0 | 15 | 1 | 10 | 0001 0000 1s |
| 47 | 0 | 16 | 1 | 10 | 0001 0000 0s |

表 16/H.263—TCOEF的VLC表（续）

| 索引 | LAST | RUN | LEVEL | 比特数目 | VLC代码 |
|----|------|-----|-------|------|-----------------|
| 48 | 0 | 17 | 1 | 10 | 0000 1111 1s |
| 49 | 0 | 18 | 1 | 10 | 0000 1111 0s |
| 50 | 0 | 19 | 1 | 10 | 0000 1110 1s |
| 51 | 0 | 20 | 1 | 10 | 0000 1110 0s |
| 52 | 0 | 21 | 1 | 10 | 0000 1101 1s |
| 53 | 0 | 22 | 1 | 10 | 0000 1101 0s |
| 54 | 0 | 23 | 1 | 12 | 0000 0100 010s |
| 55 | 0 | 24 | 1 | 12 | 0000 0100 011s |
| 56 | 0 | 25 | 1 | 13 | 0000 0101 0110s |
| 57 | 0 | 26 | 1 | 13 | 0000 0101 0111s |
| 58 | 1 | 0 | 1 | 5 | 0111s |
| 59 | 1 | 0 | 2 | 10 | 0000 1100 1s |
| 60 | 1 | 0 | 3 | 12 | 0000 0000 101s |
| 61 | 1 | 1 | 1 | 7 | 0011 11s |
| 62 | 1 | 1 | 2 | 12 | 0000 0000 100s |
| 63 | 1 | 2 | 1 | 7 | 0011 10s |
| 64 | 1 | 3 | 1 | 7 | 0011 01s |
| 65 | 1 | 4 | 1 | 7 | 0011 00s |
| 66 | 1 | 5 | 1 | 8 | 0010 011s |
| 67 | 1 | 6 | 1 | 8 | 0010 010s |
| 68 | 1 | 7 | 1 | 8 | 0010 001s |
| 69 | 1 | 8 | 1 | 8 | 0010 000s |
| 70 | 1 | 9 | 1 | 9 | 0001 1010s |
| 71 | 1 | 10 | 1 | 9 | 0001 1001s |
| 72 | 1 | 11 | 1 | 9 | 0001 1000s |
| 73 | 1 | 12 | 1 | 9 | 0001 0111s |
| 74 | 1 | 13 | 1 | 9 | 0001 0110s |
| 75 | 1 | 14 | 1 | 9 | 0001 0101s |
| 76 | 1 | 15 | 1 | 9 | 0001 0100s |
| 77 | 1 | 16 | 1 | 9 | 0001 0011s |
| 78 | 1 | 17 | 1 | 10 | 0000 1100 0s |
| 79 | 1 | 18 | 1 | 10 | 0000 1011 1s |
| 80 | 1 | 19 | 1 | 10 | 0000 1011 0s |
| 81 | 1 | 20 | 1 | 10 | 0000 1010 1s |
| 82 | 1 | 21 | 1 | 10 | 0000 1010 0s |
| 83 | 1 | 22 | 1 | 10 | 0000 1001 1s |
| 84 | 1 | 23 | 1 | 10 | 0000 1001 0s |
| 85 | 1 | 24 | 1 | 10 | 0000 1000 1s |

表 16/H.263—TCOEF的VLC表（续完）

| 索引 | LAST | RUN | LEVEL | 比特数目 | VLC代码 |
|-----|--------|-----|-------|------|-----------------|
| 86 | 1 | 25 | 1 | 11 | 0000 0001 11s |
| 87 | 1 | 26 | 1 | 11 | 0000 0001 10s |
| 88 | 1 | 27 | 1 | 11 | 0000 0001 01s |
| 89 | 1 | 28 | 1 | 11 | 0000 0001 00s |
| 90 | 1 | 29 | 1 | 12 | 0000 0100 100s |
| 91 | 1 | 30 | 1 | 12 | 0000 0100 101s |
| 92 | 1 | 31 | 1 | 12 | 0000 0100 110s |
| 93 | 1 | 32 | 1 | 12 | 0000 0100 111s |
| 94 | 1 | 33 | 1 | 13 | 0000 0101 1000s |
| 95 | 1 | 34 | 1 | 13 | 0000 0101 1001s |
| 96 | 1 | 35 | 1 | 13 | 0000 0101 1010s |
| 97 | 1 | 36 | 1 | 13 | 0000 0101 1011s |
| 98 | 1 | 37 | 1 | 13 | 0000 0101 1100s |
| 99 | 1 | 38 | 1 | 13 | 0000 0101 1101s |
| 100 | 1 | 39 | 1 | 13 | 0000 0101 1110s |
| 101 | 1 | 40 | 1 | 13 | 0000 0101 1111s |
| 102 | ESCAPE | | | 7 | 0000 011 |

表 17/H.263—RUNS和LEVELS的FLC表

| 索引 | RUN | 代 码 |
|----|-----|---------|
| 0 | 0 | 000 000 |
| 1 | 1 | 000 001 |
| 2 | 2 | 000 010 |
| . | . | . |
| . | . | . |
| 63 | 63 | 111 111 |

| 索引 | LEVEL | 代 码 |
|-----|-------|-----------|
| — | -128 | 见正文 |
| 0 | -127 | 1000 0001 |
| . | . | . |
| 125 | -2 | 1111 1110 |
| 126 | -1 | 1111 1111 |
| — | 0 | 禁止 |
| 127 | 1 | 0000 0001 |
| 128 | 2 | 0000 0010 |
| . | . | . |
| 253 | 127 | 0111 1111 |

6 译码处理

6.1 运动补偿

本子节中，描述缺省的H.263预测模式的运动补偿。有关非受限运动矢量模式中运动补偿的描述，参见附件D。有关先进的预测模式中运动补偿的描述，参见附件F。有关简约分辨率更新模式中运动补偿的描述，参见附件Q。

6.1.1 差分运动矢量

通过预测值和MVD指示的矢量差相加（见表14和表D.3）可获得宏块矢量。每宏块采用4个矢量的差分编码，参见附件F。在每宏块1个矢量的情形中，该差分编码的候选预测值取自3个周围的宏块，如图12所示。将分别计算预测值的水平和垂直分量。

在当前GOB、截面或图像边界的特殊情形中，以递增顺序使用以下判决准则：

- 1) 当相应的宏块用INTRA模式编码（若不在具有双向预测的PB帧模式中）或不编码时（COD=1），则候选预测值置零。
- 2) 若相应宏块处于图像或截面的外部（在左边），则候选预测值MV1置零。
- 3) 然后，若相应宏块处于图像的外部（在顶上）或GOB的外部（在顶上）且该当前GOB的GOB头非空，或者在截面结构模式时，它在截面的外部；则候选预测值MV2和MV3设置为MV1。
- 4) 然后，若相应宏块处于图像的外部（在右边），则候选预测值MV3置零。

对每个分量，预测值是该分量的3个候选预测值的中值。

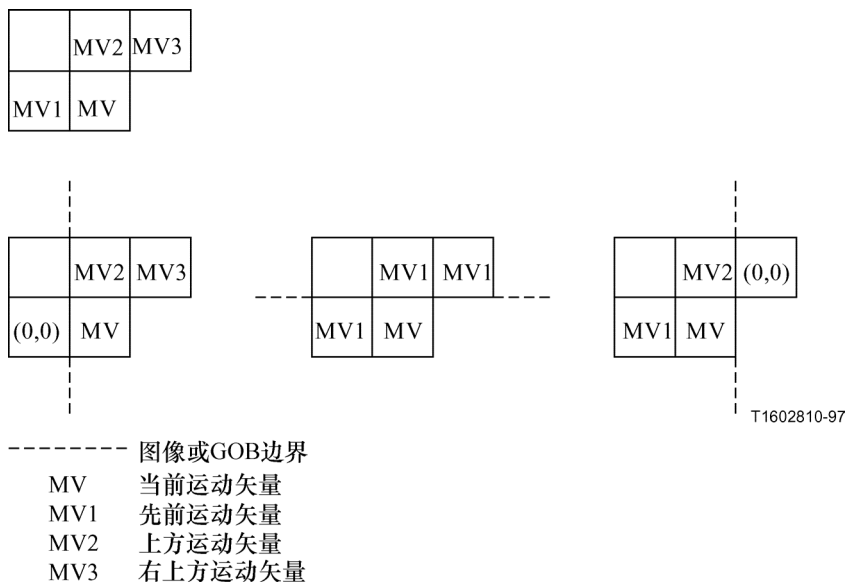


图 12/H.263—运动矢量预测

利用运动矢量分量值的范围受限的这一事实。MVD的每个VLC字代表一对差分值。仅该对差分值中的一个值将产生落入到可允许范围 $[-16, 15.5]$ 内的宏块矢量分量。运动矢量的水平分量或垂直分量的正值表明预测是由空间位置处于被预测像素右方或下方的先前图像中的像素组成的。若使用非受限运动矢量模式（见附件D），则运动矢量的译码应如D.2中的规定那样实施。

该运动矢量供宏块中所有4个亮度块的所有像素使用。归于较低色度格式，通过该宏块矢量各分量值除以2得到两个色度块的运动矢量。产生的1/4像素分辨率矢量的各分量值被修改朝向最近的半像素位置，如表18所示。

表 18/H.263—1/4像素分辨率色度矢量分量的修正

| | | | | | |
|---------|---|-----|-----|-----|---|
| 1/4像素位置 | 0 | 1/4 | 1/2 | 3/4 | 1 |
| 生成的位置 | 0 | 1/2 | 1/2 | 1/2 | 1 |

6.1.2 子像素的插值预测

使用图13中所述的双线性插值可以求出半像素值。“/”表示舍位除法。

PType中当源格式场（比特6-8）指示“扩展的PType”时，RCONTROL的值等于MPPTYPE中舍入类型（RType）比特（比特6）的值（见5.1.4.3）。否则RCONTROL有隐含值0，对改进的PB帧的B图像部分而言，不管RType比特的值，RCONTROL的值置于0（见附件M）。

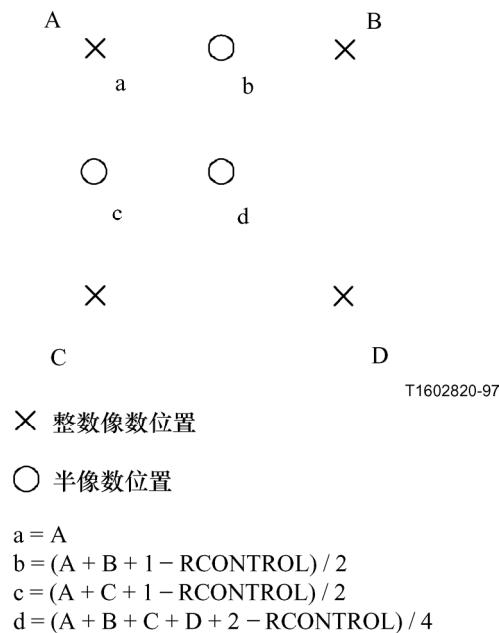


图 13/H.263—双线性插值的半像素预测

6.2 系数译码

6.2.1 反向量化

本子节中描述反向量化处理，任选的先进的INTRA编码模式使用时除外（见附件I）。若LEVEL=“0”，则重构电平REC=“0”。表15给出INTRADC的重构电平。非INTRADC系数的所有非零系数的重构电平由以下公式给出：

$$|REC| = QUANT \cdot (2 \cdot |LEVEL| + 1) \quad \text{若QUANT为“奇”}$$

$$|REC| = QUANT \cdot (2 \cdot |LEVEL| + 1) - 1 \quad \text{若QUANT为“偶”}$$

注意到这个处理不允许偶取值的数。它可以防止IDCT失配差错的累积。|REC|计算以后，添加符号可获取REC: REC=符号 (LEVEL) • |REC|。

符号 (LEVEL) 由TCOEF编码的最后比特 (见表16) 或表17给出。

6.2.2 重构电平的限幅

反向量化后，非INTRADC的所有系数的重构电平限幅到-2048到2047范围。

6.2.3 折线定位

依照图14中给出的顺序安置量化的变换系数于8×8块中，除非任选的先进的INTRA编码模式在使用 (见附件I)。系数1是DC系数。

| | | | | | | | |
|----|----|----|----|----|----|----|----|
| 1 | 2 | 6 | 7 | 15 | 16 | 28 | 29 |
| 3 | 5 | 8 | 14 | 17 | 27 | 30 | 43 |
| 4 | 9 | 13 | 18 | 26 | 31 | 42 | 44 |
| 10 | 12 | 19 | 25 | 32 | 41 | 45 | 54 |
| 11 | 20 | 24 | 33 | 40 | 46 | 53 | 55 |
| 21 | 23 | 34 | 39 | 47 | 52 | 56 | 61 |
| 22 | 35 | 38 | 48 | 51 | 57 | 60 | 62 |
| 36 | 37 | 49 | 50 | 58 | 59 | 63 | 64 |

图 14/H.263 - 量化变换系数的折线定位

6.2.4 反变换

系数反向量化和折线定位后，生成的8×8块通过一个大小8×8的可分的二维反离散余弦变换处理。限幅后反变换输出范围从-256到+255，用9比特表示。给出的反变换的传递函数为：

$$f(x, y) = \frac{1}{4} \sum_{u=0}^7 \sum_{v=0}^7 C(u)C(v)F(u, v) \cos\left[\pi(2x+1)\frac{u}{16}\right] \cos\left[\pi(2y+1)\frac{v}{16}\right]$$

$$u, v, x, y = 0, 1, 2, \dots, 7$$

其中：

x, y = 像素域中的空间坐标；

u, v = 变换域中坐标；

$C(u) = 1/\sqrt{2}$ 当 $u=0$ ，否则 $C(u) = 1$ ；

$C(v) = 1/\sqrt{2}$ 当 $v=0$ ，否则 $C(v) = 1$ 。

注一 变换块内， $x = 0$ 和 $y = 0$ 分别代表最靠近图像左边缘和顶边缘的像素。

计算反变换的计算程序不定义，但应满足附件A指定的误差容限。

6.3 块重构

6.3.1 求和

运动补偿和系数译码后（包括反变换），对每个亮度和色度块建立重构。对INTRA块，重构等于反变换的结果。对INTER块，重构通过预测和反变换的结果求和建立。求和在像素基上实施。对简约分辨率更新模式中的求和，参见附件Q。

6.3.2 限幅

为防止变换系数幅度的量化畸变以免引起在编码器环路和译码器环路中的算术溢出，插入限幅功能。预测和重构的预测误差求和后，限幅器在小于0或大于255的生成像素值上操作，分别限定它们到0和255。

附 件 A

反变换精度的技术规程

A.1 按照下面给定的随机数发生器（“C”型），产生范围从-L到+H的随机整数像素数据值。排列成8×8的块。包含各10 000块的数据集应在（L=256，H=255）、（L=H=5）和（L=H=300）的情况下产生。

A.2 对每个8×8块，使用至少64比特浮点精度的方法，进行可分的、正交的、多重矩阵的正向离散余弦变换。

$$f(u,v) = \frac{1}{4} C(u)C(v) \sum_{x=0}^7 \sum_{y=0}^7 f(x,y) \cos \left[\pi(2x+1) \frac{u}{16} \right] \cos \left[\pi(2y+1) \frac{v}{16} \right]$$

$$u, v, x, y = 0, 1, 2, \dots, 7$$

其中：

x, y = 像素域中的空间坐标；

u, v = 变换域中坐标；

$C(u) = 1/\sqrt{2}$ 当 $u=0$ ，否则 $C(u) = 1$ ；

$C(v) = 1/\sqrt{2}$ 当 $v=0$ ，否则 $C(v) = 1$ 。

A.3 对每个块，以四舍五入的方法将64个变换系数取成最接近的整数值。然后限幅它们到-2048到+2047的范围。这是到反变换的12比特输入数据。

A.4 对于由A.3产生的每个8×8的12比特数据块，用至少64比特浮点精度，进行可分离的、正交的、多重矩阵的离散余弦反变换（IDCT）。将得到的像素四舍五入成最接近的整数并限幅到-256到+255的范围。这些8×8的像素块是参考IDCT输出数据。

A.5 对于由A.3产生的每个8×8块，在测试条件下进行IDCT，并将输出限幅到-256到+255的范围。这些8×8的像素块是测试IDCT输出数据。

A.6 对64个IDCT输出像素中的每个像素，以及对上面产生的10 000块数据集的每个集，在参考数据和测试数据之间测量峰值、平均值和均方误差。

- A.7
 - 对任何像素，峰值误差幅度不超过1。
 - 对任何像素，均方误差不超过0.06。
 - 总均方误差不超过0.02。
 - 对任何像素，平均误差幅度不超过0.015。
 - 总平均误差幅度不超过0.0015。
- A.8 全零输入应产生全零输出。
- A.9 使用与A.1完全相同的数据值但改变每个像素的符号重新测量。

随机数发生器的“C”程序

```

/* L and H shall be long, that is 32 bits */
long rand      (L,H)
long          L,H;
{
    static long randx = 1;                /* long is 32 bits          */
    static double z = (double) 0x7fffffff;
    long      i,j;
    double    x;                          /* double is 64 bits      */
    randx = (randx * 1103515245) + 12345;
    i = randx & 0x7fffffff;              /* keep 30 bits           */
    x = ( (double)i ) / z;                /* range 0 to 0.99999 ... */
    x *= (L+H+1);                         /* range 0 to < L+H+1    */
    j = x;                                 /* truncate to integer     */
    return(j - L);                        /* range -L to H         */
}

```

附 件 B

假想参考译码器

假想参考译码器（HRD）定义如下：

- B.1 HRD与编码器有相同的时钟频率及相同的图像时钟频率，二者同步操作。
- B.2 HRD接收缓存器的尺寸是（ $B+BPP_{max}Kb*1024$ 比特），其中（ $BPP_{max}Kb*1024$ ）是比特流中已协商使用的每个图像的最大比特数（见3.6）。B值定义如下：

$$B = 4 \cdot R_{max} / PCF$$

其中PCF为有效图像时钟频率， R_{max} 以bit/s为单位是连接期间的最大视频比特速率。生效的图像时钟频率是标准的CIF图像时钟频率，除非在图像帧头的CPCFC场中指定惯例PCF。对B而言这个值是最小的。编码器可以使用较大的B值，提供的较大数值首先通过外部手段协商，例如ITU-T H.245建议书。

对 R_{max} 值取决于系统配置（例如，GSTN或ISDN，单路或多路链接）并可以等于物理链路所支持的最大比特速率。 R_{max} 的协商通过外部手段实现，例如ITU-T H.245建议书。

B.3 HRD的初始化为空。

B.4 HRD缓存器按图像时钟间隔（1000/PCF ms）进行检查。若缓存器中至少有一帧完整的编码图像，则按比特流序该帧前最早图像的所有数据立即去除（例如如图B.1中 t_{n+1} 的情形）。去除上面的数据后，缓存器即时占有量必须低于B。这是对包括编码图像数据和MCBPC和STUF填充数据的编码器输出比特流的要求，但这一比特流不含附件H中描述的纠错定帧比特、填充指示符（FI）、填充比特或纠错奇偶校验信息。

出于这个规定的考虑，一个完整的编码图像应是正规的I或P图像或PB帧或改进的PB帧，除非任选的时间的、SNR和空间的可量测性模式在使用。

当时间的、SNR及空间的可量测性模式在使用时（见附件O），给每个增强层一个附加的HRD，对该HRD而言，完整的编码图像为一幅EI、EP或B图像。当图像到达时，基础层缓冲器应掌握该图像帧头的比特，直至足够数量的图像帧头已经到达足以确定图像是否为基础层或增强层图像，以及增强层的编号。当它能够确定到达的图像属于增强层时，该图像的所有比特应立即传输到适当的增强层HRD，并且任何稍后到达的比特继续放置到该增强层的HRD中，直到足够数量的某些新的图像帧头已经到达足以确定该比特流应再一次按规定的路线到另一个HRD缓存器时为止。增强层识别的处理是即时和同步的，独立于图像时钟间隔检测时间。

为满足此要求，对第 $n+1$ 编码图像帧的比特数 d_{n+1} 必须满足：

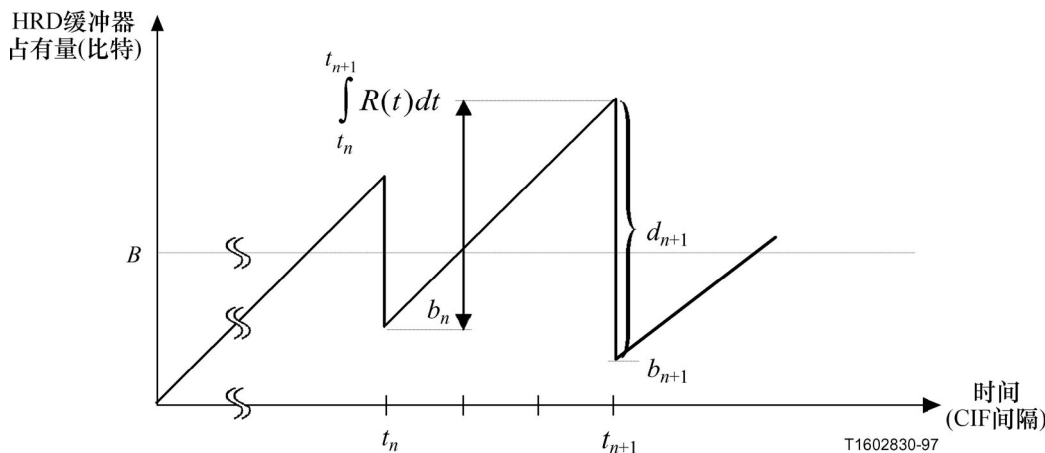
$$d_{n+1} \geq b_n + \int_{t_n}^{t_{n+1}} R(t) dt - B$$

其中：

b_n 是刚巧在时刻 t_n 之后的缓存占有量；

t_n 是第 n 编码图像帧从HRD缓存器去除的时刻；

$R(t)$ 是 t 时刻的视频比特速率。



注 — 时间 $(t_{n+1}-t_n)$ 是CIF图像的整数个周期 (1/29.97, 2/29.97, 3/29.97,...).

图 B.1/H.263—HRD缓存器占有量

附件 C

多点考虑

提供以下功能支持切换型多点操作。

C.1 冻结图像请求

促使译码器冻结其显示的图像，直到收到冻结图像释放信号或达到至少6 s的暂停周期。这个信号的传输通过外部手段进行（例如ITU-T H.245建议书）。注意类似指令也可以使用视频比特流的图像帧头内增补的增强信息发送（见L.4）。

C.2 快速更新请求

促使编码器对其下一帧图像采用适当的编码参数以INTRA模式编码，以防止缓存器溢出。这个信号的传输方法通过外部手段进行（例如ITU-T H.245建议书）。

C.3 冻结图像释放

该信号来自已经响应快速更新请求的编码器，并允许译码器脱离其冻结图像模式且以正常方式显示译码图像。为响应快速更新请求在编码的第一帧图像的帧头中通过PTYPE（见5.1.3）传输此信号。

C.4 连续出现多点和视频多路复用（CPM）

注1 — 不用于ITU-T H.324建议书。

本建议书中提供一个可协商的连续出现多点和视频多路复用模式，在该模式中通过PSBI、GSBI、SSBI和ESBI场的使用，在一个新的视频比特流中最多4个独立的H.263比特流能够多路复用成单独的“子比特流”。此模式的能力交换通过外部手段完成（例如ITU-T H.242建议书）。使用BCPM和BSBI场支持响应单独子比特流的参考图像选择反向信道数据。

在CPM模式时，每个单独的H.263比特流中CPM场应置于“1”。子比特流通过流标识符编号标识，使用每个H.263比特流的图像和GOB或截面以及EOSBS帧头中的子比特流指示符（SBI）。SBI指示H.263比特流的编号，该H.263比特流中包含该图像头以及直至合成视频比特流中的下一个图像或GOB头或截面头为止的所有后随信息。

每个子比特流被认为是正常的H.263比特流并因此它应遵照通过外部手段所交换的能力。对于不同的H.263比特流，不以任何特别的预先规定的次序传输信息，一个SBI能够有与先前SBI无关的任何值并且不同的H.263比特流的图像速率可以是不同的。每个单独比特流中的信息也完全与其他比特流中的信息无关。例如，在一个子比特流中的GFID码字不受其他子比特流中的GFID或PTYPE码字的影响。类似的，当使用图像帧头中扩展的图像类型（PLUSPTPE）时模式状态推理准则以及视频比特流操作的所有其他方面应独立、分离对每个子比特流操作。

为方便起见，具有最低子比特流标识符编号（在SBI中发送）的子比特流被认为具有最高的优先地位，在那里某些资源要求的冲突或许被迫进行优先选择（除非不同的优先惯例已通过外部手段建立）。

为了标记CPM模式的每个子比特流的结束，提供如图C.1中所示的句法，提供发送该附加句法的能力首先通过外部手段协商(虽然CPM操作在本建议书的版本1中定义，但是子比特流结束的句法在版本2中增加，因此不考虑版本1 CPM操作部分)。子比特流结束句法 (ESTUF+EOSBS+ESBI) 标记每个子比特流的结束，而不是整个流的结束，如EOS代码所做的。

注2 — 这里所定义的ITU-T H.263建议书的协商CPM子比特流操作能力，在本建议书版本2产生之前未采纳到终端的任何ITU-T建议书中(诸如ITU-T H.324建议书)。这样采纳到未来H系列终端建议书中的任何外部协商的CPM操作应隐含支持子比特流的结束句法，除非另外在H系列终端建议书中指定。

对于子比特流结束句法存在三部分。跟随使用ESTUF强制字节定位，发送23比特的EOSBS码字（相应于具有GN=30的GOB帧头，在该句法中不另外使用别的GOB头，后随单个零赋值比特留待以后使用）。然后EOSBS码字后随两比特的ESBI码字指示那个子比特流受影响。该对码字表示相关的子比特流的数据发送已经结束并且同一子比特流发送的任何后续数据应完全独立于EOSBS以前到来的数据。特别地，EOSBS代码之后，该子比特流的下一个图像不应该是INTER图像或任何能够使用前向时间预测的其他的图像类型（允许I或EI图像，但不允许P图像、PB帧、改进PB帧、B图像或EP图像）。

EOSBS和ESBI的句法在以下子节中描述。ESTUF在5.1.26中描述。



图 C.1/H.263—子比特流结束指示符的句法框图

C.4.1 子比特流结束代码 (EOSBS) (23比特)

EOSBS代码为23比特的码字，它的值为0000 0000 0000 0000 1 11110 0。是否插入此码字由编码器决定。对于在随后的ESBI场中所指示的同一子比特流而言，除非先前至少一个图像头曾经发送，否则EOSBS将不发送，并且除非发送EOSBS的能力已经通过外部手段协商否则EOSBS亦不应发送。EOSBS码应字节定位。该定位通过EOSBS起始码之前插入ESTUF来实现以至EOSBS起始码的首比特是字节的第一（最高有效）比特。（见5.1.26）

EOSBS码指示该指定的子比特流的数据发送已经停止并且该子比特流已经宣告结束，直至对那个子比特流通过另一个图像起始码的发布重新起始时为止。具有相同子比特流标识符编号 (ESBI) 的后续图像应完全独立于且不应以任何方式依赖于该EOSBS码之前发送的图像。

同视频比特流有关的控制和其他信息一般无需具有这些代码所适用的子比特流的规格。（诸如，H.242中发送的冻结图像请求或快速更新请求），它将被预先假定仅适用于所有的激活的子比特流。子比特流被认为激活，只要该子比特流的至少一个图像起始码已经被接受并且可适用于哪个子比特流所发送的最后数据不是EOS或EOSBS+ESBI。

C.4.2 结束子比特流指示符 (ESBI) (2比特)

ESBI是2比特的固定长度码字紧随EOSBS之后。它指示结束子比特流的子比特流编号。它的值是子比特流编号的自然2比特的二进制表示。

附 件 D

非受限运动矢量模式

本附件描述本建议书任选的非受限运动矢量模式。H.263的本模式能力通过外部手段标示（例如ITU-T H.245建议书）。本模式的使用在PTYPE或PLUSPTYPE中指明。

对非受限运动矢量模式而言，运动矢量的范围和供编码运动矢量差分所使用的VLC表依赖于图像帧头的PLUSPTYPE场是否存在。当PLUSPTYPE存在时，运动矢量的范围也依赖于图像尺寸和图像帧头中UUI场的值。

D.1 图像边界上的运动矢量

在本建议书的缺省预测模式中，运动矢量适当受限以致它们所参照的所有像素均在编码图像区域内（见4.2.3）。然而，在非受限运动矢量模式中，此限制被撤销，因此允许运动矢量指向图像外部。当运动矢量所参照的像素处于编码图像区域外部时，使用边缘像素替代。限定运动矢量到编码图像区域内部的最外面的全像素位置就是要寻找的边缘像素。在像素基础上单独地对运动矢量的每个分量实施运动矢量限界。

例如，对QCIF图像，若使用非受限运动矢量模式，则亮度分量的参考像素值由以下公式给出：

$$R_{umv}(x, y) = R(x', y')$$

其中：

x, y, x', y' = 像素域中的空间坐标；

$R_{umv}(x, y)$ = 非受限运动矢量模式时，参考图像在 (x, y) 的像素值；

$R(x', y')$ = 参考图像在 (x', y') 的像素值；

$$x' = \begin{cases} = 0 & \text{若 } x < 0; \\ = 175 & \text{若 } x > 175; \\ = x & \text{其它;} \end{cases}$$

$$y' = \begin{cases} = 0 & \text{若 } y < 0; \\ = 143 & \text{若 } y > 143; \\ = y & \text{其它;} \end{cases}$$

$R(x', y')$ 的编码图像区域是 $0 \leq x' \leq 175$ ， $0 \leq y' \leq 143$ 。给出的边界是整数像素位置；然而，在该边界内 (x', y') 也能够是半像素位置。

D.1.1 运动矢量值的约束

在图像帧头中若PLUSPTYPE存在，则运动矢量值如此限制以致没有任何所选择的16*16或(8*8)区域单元具有水平或垂直距离大于编码图像区域外部15个像素。注意这是比PLUSPTYPE不存在时稍小的外推范围。

注1 — 若PLUSPTYPE缺席，当使用非受限运动矢量模式时，外推范围最大为编码图像区域外部31.5个像素，当使用先进的预测模式（见附件F），而不使用非受限运动矢量模式时，外推范围为编码图像区域外部16个像素。

注2 — 当先进的预测模式（见附件F）在使用时，由于交叠的块运动补偿，每个16*16或(8*8)区域的运动矢量影响一个较大的区域。对先进预测模式的“间接”运动矢量而言，这能够引起有效外推范围的增加，由于交叠的数量（4像素或8像素当简约分辨率更新模式也在使用时）增加了所要求的外推的数量（即使对每个运动矢量所允许的值的范围也应保持先进预测模式不使用时的同样的范围）。

D.2 运动矢量范围的外延

在缺省预测模式中，运动矢量的水平分量和垂直分量均限于区间 $[-16, 15.5]$ （对B图像的前向和后向运动矢量分量，这也是有效的。）然而，在非受限运动矢量模式中，矢量分量的最大范围被扩展。

在图像帧头中若PLUSPTYPE场不存在，则运动矢量范围外延为 $[-31.5, 31.5]$ ，具有的限制为：若预测值处于 $[-15.5, 16]$ 范围内，则仅在 $[-16, 15.5]$ 范围内围绕每个运动矢量分量预测值的唯一值可达到。若预测值处于 $[-15.5, 16]$ 范围之外，则在 $[-31.5, 31.5]$ 范围内具有同预测值相同符号的所有值加上零值可达到。因此，若 MV_c 是运动矢量分量， P_c 是其预测值，则：

$$\begin{array}{ll} -31.5 \leq MV_c \leq 0 & \text{若 } -31.5 \leq P_c \leq -16 \\ -16 + P_c \leq MV_c \leq 15.5 + P_c & \text{若 } -15.5 \leq P_c \leq 16 \\ 0 \leq MV_c \leq 31.5 & \text{若 } 16.5 \leq P_c \leq 31.5 \end{array}$$

非受限运动矢量模式中，表14对 MV_D 、 MVD_{2-4} 和 $MVDB$ 的解释如下：

- 若运动矢量分量的预测值处于 $[-15.5, 16]$ 范围以内，则仅矢量差的第一列适用。
- 若运动矢量分量的预测值处于 $[-15.5, 16]$ 范围以外，则应使用表14的矢量差生成 $[-31.5, 31.5]$ 范围以内具有同预测值相同的符号的矢量分量（包括零）。

MV_D 和 MVD_{2-4} 的预测值定义为6.1.1和F.2中所规定的 $MV1$ 、 $MV2$ 和 $MV3$ 矢量分量的中值。对于 $MVDB$ ，预测值 $P_c = (TR_B \times MV) / TR_D$ ，其中 MV 代表P图像中一个8*8亮度块的矢量分量（亦见G.4）。

若PLUSPTYPE存在，则运动矢量范围不依赖于运动矢量预测值。若UUI场置于“1”，则运动矢量范围依赖于图像格式。对同CIF相当的标准图像格式该范围为 $[-32, 31.5]$ ，对同4CIF相当的那些格式该范围为 $[-64, 63.5]$ ，对同16CIF相当的那些格式该范围为 $[-128, 127.5]$ ，以及对甚至更大的惯例图像格式，该范围为 $[-256, 255.5]$ 。对惯例图像格式而言，水平和垂直运动矢量范围可以不同。水平和垂直范围在表D.1和D.2中规定。

表 D.1/H.263—当PLUSPTYPE存在及UUI=1时的水平运动矢量范围

| 图像宽度 | 水平运动矢量范围 |
|-----------------|----------------|
| 4, ..., 352 | [- 32, 31.5] |
| 356, ..., 704 | [- 64, 63.5] |
| 708, ..., 1408 | [- 128, 127.5] |
| 1412, ..., 2048 | [- 256, 255.5] |

表 D.2/H.263—当PLUSPTYPE存在及UUI=1时的垂直运动矢量范围

| 图像高度 | 垂直运动矢量范围 |
|----------------|----------------|
| 4, ..., 288 | [- 32, 31.5] |
| 292, ..., 576 | [- 64, 63.5] |
| 580, ..., 1152 | [- 128, 127.5] |

在简约分辨率更新模式中，规定的范围适用于准运动矢量。这意味着放大生成的实际运动矢量范围至近似两倍的尺寸。（亦见附件Q）。

若UUI置于“01”，则运动矢量不被限于编码的区域边界如D.1.1中所阐述的，除由其距离限制外，在简约分辨率更新模式中同样的限制适用于实际的运动矢量（而非准运动矢量）。

当PLUSPTYPE存在时为编码运动矢量，表D.3用于编码运动矢量和运动矢量预测值之间的差。表D.3中每个条目都有单个值（和表14对照）。当PLUSPTYPE存在时，运动矢量范围以及表D.3对编码运动矢量数据的使用均适用于所有类型。

运动矢量差总被编码构成水平和垂直分量对。若一对量等于(0.5, 0.5)，则产生6个连续的零。为了防止起始码仿真，该事件应跟随一个置“1”的比特。此事相应于发送一个附加的零运动矢量分量。

表 D.3/H.263—PLUSPTYPE存在时所使用的运动矢量表

| 以半像素为单位的矢量差的绝对值 | 比特数 | 代 码 |
|--|-----|--|
| 0 | 1 | 1 |
| 2 | 3 | 0s0 |
| “x ₀ ” +2 (2:3) | 5 | 0x ₀ 1s0 |
| “x ₁ x ₀ ” +4 (4:7) | 7 | 0x ₁ 1x ₀ 1s0 |
| “x ₂ x ₁ x ₀ ” +8 (8:15) | 9 | 0x ₂ 1x ₁ 1x ₀ 1s0 |
| “x ₃ x ₂ x ₁ x ₀ ” +16 (16:31) | 11 | 0x ₃ 1x ₂ 1x ₁ 1x ₀ 1s0 |
| “x ₄ x ₃ x ₂ x ₁ x ₀ ” +32 (32:63) | 13 | 0x ₄ 1x ₃ 1x ₂ 1x ₁ 1x ₀ 1s0 |
| “x ₅ x ₄ x ₃ x ₂ x ₁ x ₀ ” +64 (64:127) | 15 | 0x ₅ 1x ₄ 1x ₃ 1x ₂ 1x ₁ 1x ₀ 1s0 |
| “x ₆ x ₅ x ₄ x ₃ x ₂ x ₁ x ₀ ” +128 (128:255) | 17 | 0x ₆ 1x ₅ 1x ₄ 1x ₃ 1x ₂ 1x ₁ 1x ₀ 1s0 |
| “x ₁₀ x ₉ x ₈ x ₇ x ₆ x ₅ x ₄ x ₃ x ₂ x ₁ x ₀ ” +256 (256:511) | 19 | 0x ₇ 1x ₆ 1x ₅ 1x ₄ 1x ₃ 1x ₂ 1x ₁ 1x ₀ 1s0 |
| “x ₁₀ x ₉ x ₈ x ₇ x ₆ x ₅ x ₄ x ₃ x ₂ x ₁ x ₀ ” +512 (512:1023) | 21 | 0x ₈ 1x ₇ 1x ₆ 1x ₅ 1x ₄ 1x ₃ 1x ₂ 1x ₁ 1x ₀ 1s0 |
| “x ₁₀ x ₉ x ₈ x ₇ x ₆ x ₅ x ₄ x ₃ x ₂ x ₁ x ₀ ” +1024 (1024:2047) | 23 | 0x ₉ 1x ₈ 1x ₇ 1x ₆ 1x ₅ 1x ₄ 1x ₃ 1x ₂ 1x ₁ 1x ₀ 1s0 |
| “x ₁₀ x ₉ x ₈ x ₇ x ₆ x ₅ x ₄ x ₃ x ₂ x ₁ x ₀ ” +2048 (2048:4095) | 25 | 0x ₁₀ 1x ₉ 1x ₈ 1x ₇ 1x ₆ 1x ₅ 1x ₄ 1x ₃ 1x ₂ 1x ₁ 1x ₀ 1s0 |

表D.3是规则构造的可逆表格。每行表示以半像素为单位的运动矢量差的间隔。比特“..X₁X₀”代表在运动矢量差的绝对值的二进制表示中跟随引导“1”的所有比特。比特“S”代表运动矢量差的符号，“0”正“1”负。运动矢量差的二进制表示夹在指示代码继续或结束的比特之间。例如运动矢量差 - 13有符号S=1且二进制表示1x₂x₁x₀=1101，它被编码为0x₂1x₁1x₀1s0=011011110。最后两比特组的第二位置中的0表示该代码结束。

附 件 E

基于句法的算术编码模式

E.1 引言

如5节所述在变长编/译码中（VLC/VLD），使用基于编码器句法的特定表格，符号被VLC编码。此表通常存储VLC码字的长度和数值。在表查找操作中符号被映射为表的一个条目，然后由此条目指定的二进制码字被常规地输送到接收方的缓存器中。VLD译码中，接收的比特流在基于编码器句法的特定表格中被逐条目地匹配。此表格必须与编码器中供编码当前符号所使用的表格相同。然后表中匹配的条目被映射回相应的符号，它是VLD译码器的最终结果并用于恢复视频图像。此VLC/VLD处理意味着每个符号必须编码为固定的整数比特。取消符号的固定整数比特限制将导致减少产生的比特速率，通过算术编码它能够实现。

本附件描述本建议书任选的基于句法的算术编码（SAC）模式。本模式中，所有相应的本建议书的变长编/译码操作均用算术编/译码操作替代。H.263本模式能力通过外部手段标示（例如ITU-T H.245建议书）。本模式的使用在PTYPE中指明。

E.2 SAC编码器规格

在SAC模式中，通过使用基于编码器句法的一个特定的整数（或者模型）排列并通过访问由C语言说明的以下程序，符号被编码。

```
#define q1    16384
#define q2    32768
#define q3    49152
#define top   65535

static long  low, high, opposite_bits, length;
void        encode_a_symbol(int index, int cumul_freq[ ])
{
    length = high - low + 1;
    high = low - 1 + (length * cumul_freq[index]) / cumul_freq[0];
    low += (length * cumul_freq[index+1]) / cumul_freq[0];
    for ( ; ; ) {
        if (high < q2) {
            send out a bit "0" to PSC_FIFO;
            while (opposite_bits > 0) {
                send out a bit "1" to PSC_FIFO;
                opposite_bits--;
            }
        }
        else if (low >= q2) {
            send out a bit "1" to PSC_FIFO;
            while (opposite_bits > 0) {
                send out a bit "0" to PSC_FIFO;
                opposite_bits--;
            }
            low -= q2;
            high -= q2;
        }
        else if (low >= q1 && high < q3) {
            opposite_bits += 1;
            low -= q1;
            high -= q1;
        }
        else break;

        low *= 2;
        high = 2 * high+1;
    }
}
```

低值、高值和反向的比特值分别初始化为0、为顶及为0。PSC_FIFO是作为缓冲来自算术编码器输出比特的FIFO。模型通过cumul_freq[]规定，使用模型中的符号索引规定符号。

E.3 SAC译码器规格

SAC译码器中，通过使用基于句法的特定模型并通过访问由C语言说明的以下程序，符号被译码。

```
static long          low, high, code_value, bit, length, index, cum;
int  decode_a_symbol(int cumul_freq[ ])
{
    length = high - low + 1;
    cum = (-1 + (code_value - low + 1) * cumul_freq[0]) / length;
    for (index = 1; cumul_freq[index] > cum; index++);
    high = low - 1 + (length * cumul_freq[index-1]) / cumul_freq[0];
    low += (length * cumul_freq[index]) / cumul_freq[0];
    for ( ; ; ) {
        if (high < q2);
        else if (low >= q2) {
            code_value -= q2;
            low -= q2;
            high -= q2;
        }
        else if (low >= q1 && high < q3) {
            code_value -= q1;
            low -= q1;
            high -= q1;
        }
        else break;

        low *= 2;
        high = 2 * high + 1;
        get bit from PSC_FIFO;
        code_value = 2 * code_value + bit;
    }
    return (index-1);
}
```

再次，模型通过cumul_freq[]规定。译码符号通过模型中的其索引恢复。PSC_FIFO是作为缓冲入网比特流的FIFO。为启动译码算术编码比特流，通过访问以下程序，译码器初始化。

```
void decoder_reset( )
{
    code_value = 0;
    low = 0;
    high = top;
    for (int i = 1; i <= 16; i++) {
        get bit from PSC_FIFO;
        code_value = 2 * code_value + bit;
    }
}
```

E.4 句法

作为本建议书的VLC表模式，符号的句法被分成为四层：图像、块组、宏块和块。顶上三层句法完全保持相同。块层的句法也保持相当类似，如图E.1中说明。

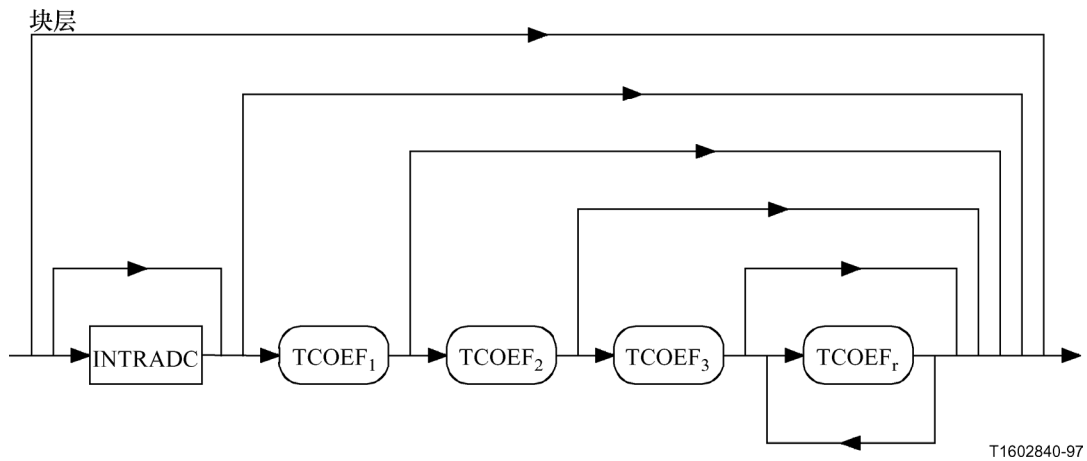


图 E.1/H.263—SAC块层结构

图E.1中，TCOEF₁、TCOEF₂、TCOEF₃和TCOEF_r是如5.4.2中规定的LAST-RUN-LEVEL符号，并且分别是可容许的第一、第二、第三和其余符号。在块层，仅当一个、两个、三个或更多个系数分别出现时，TCOEF₁、TCOEF₂、TCOEF₃和TCOEF_r才出现。

E.5 PSC_FIFO

编码器或译码器中的PSC_FIFO是大于17比特长度的FIFO。在编码器PSC_FIFO中，非法仿真的PSC和GBSC被找出并通过每连续出现14个“0”之后填充一个“1”来避免（它们不是PSC或GBSC的一部分）。在译码器PSC_FIFO中，每个14个“0”的串之后的第一个“1”被删除；若替代14个“0”的串之后跟“0”，它指示合法的PSC或GBSC被检测。PSC或GBSC的精确位置通过跟随“0”串的下一个“1”决定。

E.6 头部层符号

句法头部层被认为是块和宏块层以上的那些句法单元（见图6和正文中的句法说明）。基础版本1的头部级句法能够构成三种可能的串，(PSTUF)--PSC--TR--PTYPE--PQUANT--CPM--(PSBI)--(TRB--DBQUANT)--PEI--(PSUPP--PEI...), (GSTUF)--GBSC--GN--(GSBI)--GFID--GQUANT, 和(ESTUF)--EOS--(PSTUF)。在版本2的修订句法中，头部级句法可能有其他的结构（见图6和正文中的句法说明）。头部级句法串被直接发送到PSC_FIFO作为编码器一方本建议书的正规VLC表模式，并在合法的PSC、GBSC、SSC、EOS或EOSBS被检测后直接从译码器的PSC_FIFO中送出。

在视频会话中，若头部不是该句法串的第一个，则通过访问以下程序在发送头部之前，算术编码器需要重新设置。若(ESTUF)--EOS[或对所发送的最后头部帧子比特流的(ESTUF)--EOSBS]未发送，则在视频会话结束时也被访问此程序。

```
void encoder_flush( )
{
    opposite_bits++;
    if (low < q1) {
        send out a bit "0" to PSC_FIFO;
        while (opposite_bits > 0) {
            send out a bit "1" to PSC_FIFO;
            opposite_bits- -;
        }
    }
}
```

```

    }
}
else {
    send out a bit "1" to PSC_FIFO;
    while (opposite_bits > 0) {
        send out a bit "0" to PSC_FIFO;
        opposite_bits- -;
    }
}
low = 0;
high = top;
}

```

译码器中，每个固定长度符号串之后，程序译码器_重新设置被访问。

E.7 宏块层和块层符号

E.8中包括宏块层和块层符号模型。5节的VLC表中给出的索引供该模型中的整数索引所使用。

P图像中COD模型用cumf_COD命名。COD为0的索引是0，COD为1的索引是1。P图像中MCBPC模型用cumf_MCBPC_N04MVQ命名，除非在图像帧头中PLUSPTYPE存在以及或先进的预测模式(附件F)或分块滤波器模式(附件J)在使用，在此情形模型用cumf_MCBPC_4MVQ命名。MCBPC的索引，对I图像在表7中定义，对P图像在表8中定义。I图像中MCBPC模型用cumf_MCBPC_INTRA命名。

MODB模型，若使用附件G为cumf_MODB_G，若使用附件M为cumf_MODB_M。MODB的索引分别在表11或表M.1中定义。CBPB_N，N=1，2，…，4，模型是cumf_YCBPB，CBPB_N，N=5，6，模型是cumf_UVCBPB，对CBPB_N=0，索引为0；对CBPB_N=1，其索引为1。

CBPY，在INTER宏块其模型是cumf_CBPY，在INTRA宏块其模型是cumf_CBPY_INTRA。DQUANT模型是cumf_DQUANT。CBPY和DQUANT索引分别在表13和表12中规定。

MVD、MVD₂₋₄和MVDB模型是cumf_MVD，INTRADC模型是cumf_INTRADC，索引分别在表15和表14中规定。

非换码的TCOEF由TCOEF1/2/3/r的符号后随TCOEF符号的SIGN符号组成。对TCOEF1、TCOEF2、TCOEF3和TCOEFR，在INTER块模型是cumf_TCOEF1、cumf_TCOEF2、cumf_TCOEF3、cumf_TCOEFR。在INTRA块模型是cumf_TCOEF1_INTRA、cumf_TCOEF2_INTRA、cumf_TCOEF3_INTRA，cumf_TCOEFR_INTRA。对所有TCOEF，索引在表16规定。SIGN模型是cumf_SIGN。SIGN索引，正号为0，负号为1。

ESCAPE以后，对LAST、RUN、LEVEL，对INTER(INTRA)块模型是cumf_LAST(cumf_LAST_INTRA)、cumf_RUN(cumf_RUN_INTRA)、cumf_LEVEL(cumf_LEVEL_INTRA)。LAST索引，LAST=0为0，LAST=1为1，RUN和LEVEL索引在表17规定。

INTRA_MODE模型是cumf_INTRA_AC_DC，索引在表I.1中定义。

E.8 SAC模型

```

int cumf_COD[3]={16383, 6849, 0};
int cumf_MCBPC_no4MVQ[22]={16383, 4105, 3088, 2367, 1988, 1621, 1612, 1609, 1608, 496, 353, 195, 77, 22,
17, 12, 5, 4, 3, 2, 1, 0};

```

```

int cumf_MCBPC_4MVQ[26]={16383, 6880, 6092, 5178, 4916, 3965, 3880, 3795, 3768, 1491, 1190, 889, 655,
442, 416, 390, 360, 337, 334, 331, 327, 326, 88, 57, 26, 0};
int cumf_MCBPC_intra[10]={16383, 7410, 6549, 5188, 442, 182, 181, 141, 1, 0};
int cumf_MODB_G[4]={16383, 6062, 2130, 0};
int cumf_MODB_M[7] = {16383, 6717, 4568, 2784, 1370, 655, 0};
int cumf_YCBPB[3]={16383, 6062, 0};
int cumf_UVCBPB[3]={16383, 491, 0};
int cumf_CBPY[17]={16383, 14481, 13869, 13196, 12568, 11931, 11185, 10814, 9796, 9150, 8781, 7933, 6860,
6116, 4873, 3538, 0};
int cumf_CBPY_intra[17]={16383, 13619, 13211, 12933, 12562, 12395, 11913, 11783, 11004, 10782, 10689,
9928, 9353, 8945, 8407, 7795, 0};
int cumf_DQUANT[5]={16383, 12287, 8192, 4095, 0};
int cumf_MVD[65]={16383, 16380, 16369, 16365, 16361, 16357, 16350, 16343, 16339, 16333, 16326, 16318,
16311, 16306, 16298, 16291, 16283, 16272, 16261, 16249, 16235, 16222, 16207, 16175, 16141, 16094, 16044,
15936, 15764, 15463, 14956, 13924, 11491, 4621, 2264, 1315, 854, 583, 420, 326, 273, 229, 196, 166, 148, 137,
123, 114, 101, 91, 82, 76, 66, 59, 53, 46, 36, 30, 26, 24, 18, 14, 10, 5, 0};
int cumf_INTRADC[255]={16383, 16380, 16379, 16378, 16377, 16376, 16370, 16361, 16360, 16359, 16358,
16357, 16356, 16355, 16343, 16238, 16237, 16236, 16230, 16221, 16220, 16205, 16190, 16169, 16151, 16130,
16109, 16094, 16070, 16037, 16007, 15962, 15938, 15899, 15854, 15815, 15788, 15743, 15689, 15656, 15617,
15560, 15473, 15404, 15296, 15178, 15106, 14992, 14868, 14738, 14593, 14438, 14283, 14169, 14064, 14004,
13914, 13824, 13752, 13671, 13590, 13515, 13458, 13380, 13305, 13230, 13143, 13025, 12935, 12878, 12794,
12743, 12656, 12596, 12521, 12443, 12359, 12278, 12200, 12131, 12047, 12002, 11948, 11891, 11828, 11744,
11663, 11588, 11495, 11402, 11288, 11204, 11126, 11039, 10961, 10883, 10787, 10679, 10583, 10481, 10360,
10227, 10113, 9961, 9828, 9717, 9584, 9485, 9324, 9112, 9019, 8908, 8766, 8584, 8426, 8211, 7920, 7663, 7406,
7152, 6904, 6677, 6453, 6265, 6101, 5904, 5716, 5489, 5307, 5056, 4850, 4569, 4284, 3966, 3712, 3518, 3342,
3206, 3048, 2909, 2773, 2668, 2596, 2512, 2370, 2295, 2232, 2166, 2103, 2022, 1956, 1887, 1830, 1803, 1770,
1728, 1674, 1635, 1599, 1557, 1500, 1482, 1434, 1389, 1356, 1317, 1284, 1245, 1200, 1179, 1140, 1110, 1092,
1062, 1044, 1035, 1014, 1008, 993, 981, 954, 936, 912, 894, 876, 864, 849, 828, 816, 801, 792, 777, 756, 732,
690, 660, 642, 615, 597, 576, 555, 522, 489, 459, 435, 411, 405, 396, 387, 375, 360, 354, 345, 344, 329, 314, 293,
278, 251, 236, 230, 224, 215, 214, 208, 199, 193, 184, 178, 169, 154, 127, 100, 94, 73, 37, 36, 35, 34, 33, 32, 31,
30, 29, 28, 27, 26, 20, 19, 18, 17, 16, 15, 9, 0};
int cumf_TCOEF1[104]={16383, 13455, 12458, 12079, 11885, 11800, 11738, 11700, 11681, 11661, 11651,
11645, 11641, 10572, 10403, 10361, 10346, 10339, 10335, 9554, 9445, 9427, 9419, 9006, 8968, 8964, 8643,
8627, 8624, 8369, 8354, 8352, 8200, 8192, 8191, 8039, 8036, 7920, 7917, 7800, 7793, 7730, 7727, 7674, 7613,
7564, 7513, 7484, 7466, 7439, 7411, 7389, 7373, 7369, 7359, 7348, 7321, 7302, 7294, 5013, 4819, 4789, 4096,
4073, 3373, 3064, 2674, 2357, 2177, 1975, 1798, 1618, 1517, 1421, 1303, 1194, 1087, 1027, 960, 890, 819, 758,
707, 680, 656, 613, 566, 534, 505, 475, 465, 449, 430, 395, 358, 335, 324, 303, 295, 286, 272, 233, 215, 0};
int cumf_TCOEF2[104]={16383, 13582, 12709, 12402, 12262, 12188, 12150, 12131, 12125, 12117, 12113,
12108, 12104, 10567, 10180, 10070, 10019, 9998, 9987, 9158, 9037, 9010, 9005, 8404, 8323, 8312, 7813, 7743,
7726, 7394, 7366, 7364, 7076, 7062, 7060, 6810, 6797, 6614, 6602, 6459, 6454, 6304, 6303, 6200, 6121, 6059,
6012, 5973, 5928, 5893, 5871, 5847, 5823, 5809, 5796, 5781, 5771, 5763, 5752, 4754, 4654, 4631, 3934, 3873,
3477, 3095, 2758, 2502, 2257, 2054, 1869, 1715, 1599, 1431, 1305, 1174, 1059, 983, 901, 839, 777, 733, 683, 658,

```

```

606, 565, 526, 488, 456, 434, 408, 380, 361, 327, 310, 296, 267, 259, 249, 239, 230, 221, 214, 0};
int cumf_TCOEF3[104]={16383, 13532, 12677, 12342, 12195, 12112, 12059, 12034, 12020, 12008, 12003,
12002, 12001, 10586, 10297, 10224, 10202, 10195, 10191, 9223, 9046, 8999, 8987, 8275, 8148, 8113, 7552,
7483, 7468, 7066, 7003, 6989, 6671, 6642, 6631, 6359, 6327, 6114, 6103, 5929, 5918, 5792, 5785, 5672, 5580,
5507, 5461, 5414, 5382, 5354, 5330, 5312, 5288, 5273, 5261, 5247, 5235, 5227, 5219, 4357, 4277, 4272, 3847,
3819, 3455, 3119, 2829, 2550, 2313, 2104, 1881, 1711, 1565, 1366, 1219, 1068, 932, 866, 799, 750, 701, 662,
605, 559, 513, 471, 432, 403, 365, 336, 312, 290, 276, 266, 254, 240, 228, 223, 216, 206, 199, 192, 189, 0};
int cumf_TCOEFr[104]={16383, 13216, 12233, 11931, 11822, 11776, 11758, 11748, 11743, 11742, 11741,
11740, 11739, 10203, 9822, 9725, 9691, 9677, 9674, 8759, 8609, 8576, 8566, 7901, 7787, 7770, 7257, 7185,
7168, 6716, 6653, 6639, 6276, 6229, 6220, 5888, 5845, 5600, 5567, 5348, 5327, 5160, 5142, 5004, 4900, 4798,
4743, 4708, 4685, 4658, 4641, 4622, 4610, 4598, 4589, 4582, 4578, 4570, 4566, 3824, 3757, 3748, 3360, 3338,
3068, 2835, 2592, 2359, 2179, 1984, 1804, 1614, 1445, 1234, 1068, 870, 739, 668, 616, 566, 532, 489, 453, 426,
385, 357, 335, 316, 297, 283, 274, 266, 259, 251, 241, 233, 226, 222, 217, 214, 211, 209, 208, 0};
int cumf_TCOEF1_intra[104]={16383, 13383, 11498, 10201, 9207, 8528, 8099, 7768, 7546, 7368, 7167, 6994,
6869, 6005, 5474, 5220, 5084, 4964, 4862, 4672, 4591, 4570, 4543, 4397, 4337, 4326, 4272, 4240, 4239, 4212,
4196, 4185, 4158, 4157, 4156, 4140, 4139, 4138, 4137, 4136, 4125, 4124, 4123, 4112, 4111, 4110, 4109, 4108,
4107, 4106, 4105, 4104, 4103, 4102, 4101, 4100, 4099, 4098, 4097, 3043, 2897, 2843, 1974, 1790, 1677, 1552,
1416, 1379, 1331, 1288, 1251, 1250, 1249, 1248, 1247, 1236, 1225, 1224, 1223, 1212, 1201, 1200, 1199, 1198,
1197, 1196, 1195, 1194, 1193, 1192, 1191, 1190, 1189, 1188, 1187, 1186, 1185, 1184, 1183, 1182, 1181, 1180,
1179, 0};
int cumf_TCOEF2_intra[104]={16383, 13242, 11417, 10134, 9254, 8507, 8012, 7556, 7273, 7062, 6924, 6839,
6741, 6108, 5851, 5785, 5719, 5687, 5655, 5028, 4917, 4864, 4845, 4416, 4159, 4074, 3903, 3871, 3870, 3765,
3752, 3751, 3659, 3606, 3580, 3541, 3540, 3514, 3495, 3494, 3493, 3474, 3473, 3441, 3440, 3439, 3438, 3425,
3424, 3423, 3422, 3421, 3420, 3401, 3400, 3399, 3398, 3397, 3396, 2530, 2419, 2360, 2241, 2228, 2017, 1687,
1576, 1478, 1320, 1281, 1242, 1229, 1197, 1178, 1152, 1133, 1114, 1101, 1088, 1087, 1086, 1085, 1072, 1071,
1070, 1069, 1068, 1067, 1066, 1065, 1064, 1063, 1062, 1061, 1060, 1059, 1058, 1057, 1056, 1055, 1054, 1053,
1052, 0};
int cumf_TCOEF3_intra[104]={16383, 12741, 10950, 10071, 9493, 9008, 8685, 8516, 8385, 8239, 8209, 8179,
8141, 6628, 5980, 5634, 5503, 5396, 5327, 4857, 4642, 4550, 4481, 4235, 4166, 4151, 3967, 3922, 3907, 3676,
3500, 3324, 3247, 3246, 3245, 3183, 3168, 3084, 3069, 3031, 3030, 3029, 3014, 3013, 2990, 2975, 2974, 2973,
2958, 2943, 2928, 2927, 2926, 2925, 2924, 2923, 2922, 2921, 2920, 2397, 2298, 2283, 1891, 1799, 1591, 1445,
1338, 1145, 1068, 1006, 791, 768, 661, 631, 630, 615, 592, 577, 576, 561, 546, 523, 508, 493, 492, 491, 476, 475,
474, 473, 472, 471, 470, 469, 468, 453, 452, 451, 450, 449, 448, 447, 446, 0};
int cumf_TCOEFr_intra[104]={16383, 12514, 10776, 9969, 9579, 9306, 9168, 9082, 9032, 9000, 8981, 8962,
8952, 7630, 7212, 7053, 6992, 6961, 6940, 6195, 5988, 5948, 5923, 5370, 5244, 5210, 4854, 4762, 4740, 4384,
4300, 4288, 4020, 3968, 3964, 3752, 3668, 3511, 3483, 3354, 3322, 3205, 3183, 3108, 3046, 2999, 2981, 2974,
2968, 2961, 2955, 2949, 2943, 2942, 2939, 2935, 2934, 2933, 2929, 2270, 2178, 2162, 1959, 1946, 1780, 1651,
1524, 1400, 1289, 1133, 1037, 942, 849, 763, 711, 591, 521, 503, 496, 474, 461, 449, 442, 436, 426, 417, 407,
394, 387, 377, 373, 370, 367, 366, 365, 364, 363, 362, 358, 355, 352, 351, 350, 0};
int cumf_SIGN[3]={16383, 8416, 0};
int cumf_LAST[3]={16383, 9469, 0};
int cumf_LAST_intra[3]={16383, 2820, 0};

```



```

int cumf_RUN[65]={16383, 15310, 14702, 13022, 11883, 11234, 10612, 10192, 9516, 9016, 8623, 8366, 7595,
7068, 6730, 6487, 6379, 6285, 6177, 6150, 6083, 5989, 5949, 5922, 5895, 5828, 5774, 5773, 5394, 5164, 5016,
4569, 4366, 4136, 4015, 3867, 3773, 3692, 3611, 3476, 3341, 3301, 2787, 2503, 2219, 1989, 1515, 1095, 934,
799, 691, 583, 435, 300, 246, 206, 125, 124, 97, 57, 30, 3, 2, 1, 0};
int cumf_RUN_intra[65]={16383, 10884, 8242, 7124, 5173, 4745, 4246, 3984, 3034, 2749, 2607, 2298, 966, 681,
396, 349, 302, 255, 254, 253, 206, 159, 158, 157, 156, 155, 154, 153, 106, 35, 34, 33, 32, 31, 30, 29, 28, 27, 26,
25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0};
int cumf_LEVEL[255]={16383, 16382, 16381, 16380, 16379, 16378, 16377, 16376, 16375, 16374, 16373, 16372,
16371, 16370, 16369, 16368, 16367, 16366, 16365, 16364, 16363, 16362, 16361, 16360, 16359, 16358, 16357,
16356, 16355, 16354, 16353, 16352, 16351, 16350, 16349, 16348, 16347, 16346, 16345, 16344, 16343, 16342,
16341, 16340, 16339, 16338, 16337, 16336, 16335, 16334, 16333, 16332, 16331, 16330, 16329, 16328, 16327,
16326, 16325, 16324, 16323, 16322, 16321, 16320, 16319, 16318, 16317, 16316, 16315, 16314, 16313, 16312,
16311, 16310, 16309, 16308, 16307, 16306, 16305, 16304, 16303, 16302, 16301, 16300, 16299, 16298, 16297,
16296, 16295, 16294, 16293, 16292, 16291, 16290, 16289, 16288, 16287, 16286, 16285, 16284, 16283, 16282,
16281, 16280, 16279, 16278, 16277, 16250, 16223, 16222, 16195, 16154, 16153, 16071, 15989, 15880, 15879,
15878, 15824, 15756, 15674, 15606, 15538, 15184, 14572, 13960, 10718, 7994, 5379, 2123, 1537, 992, 693, 611,
516, 448, 421, 380, 353, 352, 284, 257, 230, 203, 162, 161, 160, 133, 132, 105, 104, 103, 102, 101, 100, 99, 98,
97, 96, 95, 94, 93, 92, 91, 90, 89, 88, 87, 86, 85, 84, 83, 82, 81, 80, 79, 78, 77, 76, 75, 74, 73, 72, 71, 70, 69, 68,
67, 66, 65, 64, 63, 62, 61, 60, 59, 58, 57, 56, 55, 54, 53, 52, 51, 50, 49, 48, 47, 46, 45, 44, 43, 42, 41, 40, 39, 38,
37, 36, 35, 34, 33, 32, 31, 30, 29, 28, 27, 26, 25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6,
5, 4, 3, 2, 1, 0};
int cumf_LEVEL_intra[255]={16383, 16379, 16378, 16377, 16376, 16375, 16374, 16373, 16372, 16371, 16370,
16369, 16368, 16367, 16366, 16365, 16364, 16363, 16362, 16361, 16360, 16359, 16358, 16357, 16356, 16355,
16354, 16353, 16352, 16351, 16350, 16349, 16348, 16347, 16346, 16345, 16344, 16343, 16342, 16341, 16340,
16339, 16338, 16337, 16336, 16335, 16334, 16333, 16332, 16331, 16330, 16329, 16328, 16327, 16326, 16325,
16324, 16323, 16322, 16321, 16320, 16319, 16318, 16317, 16316, 16315, 16314, 16313, 16312, 16311, 16268,
16267, 16224, 16223, 16180, 16179, 16136, 16135, 16134, 16133, 16132, 16131, 16130, 16129, 16128, 16127,
16126, 16061, 16018, 16017, 16016, 16015, 16014, 15971, 15970, 15969, 15968, 15925, 15837, 15794, 15751,
15750, 15749, 15661, 15618, 15508, 15376, 15288, 15045, 14913, 14781, 14384, 13965, 13502, 13083, 12509,
12289, 12135, 11892, 11738, 11429, 11010, 10812, 10371, 9664, 9113, 8117, 8116, 8028, 6855, 5883, 4710,
4401, 4203, 3740, 3453, 3343, 3189, 2946, 2881, 2661, 2352, 2132, 1867, 1558, 1382, 1250, 1162, 1097, 1032,
967, 835, 681, 549, 439, 351, 350, 307, 306, 305, 304, 303, 302, 301, 300, 299, 298, 255, 212, 211, 210, 167, 166,
165, 164, 163, 162, 161, 160, 159, 158, 115, 114, 113, 112, 111, 68, 67, 66, 65, 64, 63, 62, 61, 60, 59, 58, 57, 56,
55, 54, 53, 52, 51, 50, 49, 48, 47, 46, 45, 44, 43, 42, 41, 40, 39, 38, 37, 36, 35, 34, 33, 32, 31, 30, 29, 28, 27, 26,
25, 24, 23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0};
int cumf_INTRA_AC_DC[4]={16383, 9229, 5461, 0};

```

附件 F

先进的预测模式

F.1 引言

本附件描述本建议书任选的先进的预测模式，包括交叠块运动补偿及每宏块4个运动矢量的可能性。本模式能力通过外部手段标示（例如ITU-T H.245建议书）。本模式的使用在PTYPE中指明。先进的预测模式中，允许运动矢量跨越图像边界，如非受限运动矢量模式中的情形（此技术的描述参见D.1）。非受限运动矢量模式的扩展的运动矢量范围特性不自动包括在先进的预测模式中，只有选择非受限运动矢量模式时它才被激活。若同PB帧模式结合在一起使用先进的预测模式，则交叠的运动补偿仅供P图像预测使用，而不供B图像使用。

F.2 每宏块4个运动矢量

本建议书中，除先进的预测模式或分块滤波器模式外，每宏块使用一个运动矢量。本模式中，对每个宏块1个或4个矢量判决由MCBPC码字指示。对某确定宏块而言若仅传输一个运动矢量，那么它被规定为4个矢量具有同一矢量值。若MCBPC指示对当前宏块传输4个运动矢量，则第一运动矢量的信息作为码字MVD传输，其余3个另外的运动矢量信息作为码字MVD₂₋₄传输（亦见5.3.7和5.3.8）。

依照6.1.1中给出的判决准则，用类似于每宏块仅存在一个运动矢量时的方式，通过预测值和由MVD与MVD₂₋₄指示的矢量差相加可获得矢量。再次，对水平和垂直分量单独计算预测值。然而，候选预测值MV1、MV2、MV3重新定义，如图F.1所示。若每宏块仅存在一个矢量，那么MV1、MV2和MV3被定义为如图5中编号1的8×8块的候选预测值（这个定义在图F.1的4个小图的左上方给出）。

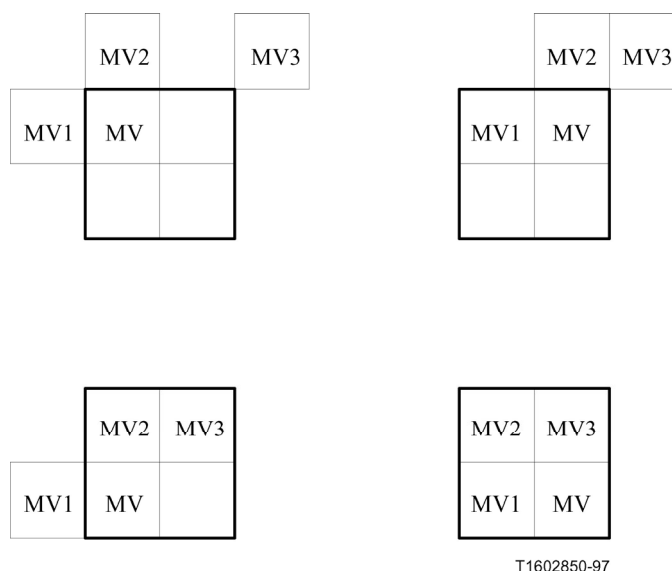


图 F.1/H.263—宏块中对于每个亮度块候选预测值MV1、MV2和MV3的再定义

若使用4个矢量，则每个运动矢量用于宏块中4个亮度块之一的所有像素。运动矢量的编号同图5中给出的4个亮度块编号相同。两个色度块的运动矢量 MVD_{CHR} 通过计算4个亮度矢量之和并将该和除以8获得；修正产生的1/16像素分辨率的矢量分量值朝向最接近的半像素位置，如表F.1所示。

表 F.1/H.263—1/16像素分辨率色度矢量分量的修正

| | | | | | | | | | | | | | | | | | |
|----------|---|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|-----|
| 1/16像素位置 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | /16 |
| 生成的位置 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | /2 |

使用6.1.2中所述的双线性内插可出现半像素值。先进的预测模式中，亮度预测可通过F.3节中所述的交叠的运动补偿获得。色度预测通过应用运动矢量 MVD_{CHR} 到两个色度块中的所有像素来获得（如在缺省预测模式中所做的）。

F.3 亮度的交叠运动补偿

在 8×8 亮度预测块中，每个像素为三个预测值的加权和除以8（使用四舍五入）。为了获取3个预测值，使用3个运动矢量：当前亮度块的运动矢量，来自4个“间接”矢量中的两个运动矢量：

- 当前亮度块的左方或右方块的运动矢量；
- 当前亮度块的上方或下方块的运动矢量。

对于每个像素，使用两个最靠近块边界的块的间接运动矢量。这意味着对上半部分块，使用相应于当前块上面的块运动矢量，而对下半部分块使用相应于当前块下面的块运动矢量（见图F.3）。类似的，对左半部分块，使用相应于当前块左边的块运动矢量；而对右半部分块，使用相应于当前块右边的块运动矢量（见图F.4）。

令 (x, y) 为以整数像素为单位度量的图像中的位置。

令 (m, n) 为图像中的整数块索引，如以下给出的：

$$m = x / 8 \quad \text{和} \quad n = y / 8$$

其中“/”代表舍位除法。

令 (i, j) 为 8×8 块中整数像素位置，如以下给出的：

$$i = x - m \cdot 8 \quad \text{和} \quad j = y - n \cdot 8$$

导出：

$$(x, y) = (m \cdot 8 + i, n \cdot 8 + j)$$

令 (MV_x^k, MV_y^k) 为运动矢量，其中可以包含全像素或半像素补偿，具有 $k = 0, 1$ 或 2 。例如， (MV_x^k, MV_y^k) 可以等于 $(-7.0, 13.5)$ 。这里 (MV_x^0, MV_y^0) 代表当前块 (m, n) 的运动矢量， (MV_x^1, MV_y^1) 代表当前块 (m, n) 的或上面或下面块的运动矢量，且如以上定义的， (MV_x^2, MV_y^2) 代表当前块 (m, n) 的或左或右块的运动矢量。

具有块索引 (m, n) 的 8×8 亮度预测块中，每个像素 $P(x, y)$ 的生成可由以下公式决定：

$$P(x, y) = (q(x, y) \cdot H_0(i, j) + r(x, y) \cdot H_1(i, j) + s(x, y) \cdot H_2(i, j) + 4) / 8$$

其中 $q(x, y)$ 、 $r(x, y)$ 和 $s(x, y)$ 是取自参考图像的预测值，定义为：

$$\begin{aligned} q(x, y) &= p(x + MV_x^0, y + MV_y^0), \\ r(x, y) &= p(x + MV_x^1, y + MV_y^1), \\ s(x, y) &= p(x + MV_x^2, y + MV_y^2), \end{aligned}$$

其中 $P(x + MV_x^k, y + MV_y^k)$ 是参考图像中在位置 $(x + MV_x^k, y + MV_y^k)$ 上的预测值。注意 $(x + MV_x^k, y + MV_y^k)$ 可以在该图像的外部并可以是全像素或半像素位置。在使用半像素运动矢量的情形中 $P(x + MV_x^k, y + MV_y^k)$ 涉及使用6.1.2中描述的内差处理之后获得的值。

矩阵 $H_0(i, j)$ 、 $H_1(i, j)$ 和 $H_2(i, j)$ 在图F₂、F₃和F₄中定义，其中 (i, j) 分别代表矩阵的列和行。

当既不使用截面结构模式（见附件K），又不使用独立分段译码模式时，来自其他视频图像分段的间接运动矢量以作为当前GOB内部间接运动矢量的相同方式使用。若或截面结构模式或独立分段译码模式在使用，则相应于来自其他视频图像分段的块间接运动矢量被设置为当前块的运动矢量，而不必考虑下一段落中所描述的其他情况（视频图像分段的定义见附件R）。

若围绕宏块的一个块未被编码，则相应的间接运动矢量置“0”。若围绕块的一个块被INTRA编码，则相应的间接运动矢量用当前块的运动矢量替代，PB帧模式时除外。在这种情况下（PB帧模式中INTRA块），使用INTRA块的运动矢量（亦见附件G）。若当前块处于图像的边界，因此围绕块不存在，则相应的间接运动矢量用当前运动矢量替代。在所有情形，若当前块处于宏块的底部（块编号3或4，见图5），则相应于当前宏块下面的宏块中的一个 8×8 亮度块的间接运动矢量由当前块的运动矢量替代。

预测加权值在图F2、F3和F4中给出。

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 4 | 5 | 5 | 5 | 5 | 5 | 5 | 4 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 6 | 6 | 6 | 6 | 5 | 5 |
| 5 | 5 | 6 | 6 | 6 | 6 | 5 | 5 |
| 5 | 5 | 6 | 6 | 6 | 6 | 5 | 5 |
| 5 | 5 | 6 | 6 | 6 | 6 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 4 | 5 | 5 | 5 | 5 | 5 | 5 | 4 |

图 F.2/H.263—具有当前亮度块运动矢量预测的加权值 H_0

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| 1 | 1 | 2 | 2 | 2 | 2 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 2 | 2 | 2 | 2 | 1 | 1 |
| 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

图 F.3/H.263—具有当前亮度块的顶或底部亮度块运动矢量预测的加权值 H_1

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 2 | 1 | 1 | 1 | 1 | 1 | 1 | 2 |
| 2 | 2 | 1 | 1 | 1 | 1 | 2 | 2 |
| 2 | 2 | 1 | 1 | 1 | 1 | 2 | 2 |
| 2 | 2 | 1 | 1 | 1 | 1 | 2 | 2 |
| 2 | 2 | 1 | 1 | 1 | 1 | 2 | 2 |
| 2 | 2 | 1 | 1 | 1 | 1 | 2 | 2 |
| 2 | 2 | 1 | 1 | 1 | 1 | 2 | 2 |
| 2 | 1 | 1 | 1 | 1 | 1 | 1 | 2 |

图 F.4/H.263—具有当前亮度块的左或右方亮度块运动矢量预测的加权值 H_2

附件 G

PB帧模式

G.1 引言

本附件描述本建议书任选的PB帧模式。本模式能力通过外部手段标示（例如ITU-T H.245建议书）。本模式的使用在PTYPE中指明。

一个PB帧由两种编码图像作为一个单元组成。名称PB源于ITU-T H.262建议书中图像种类的名称，该建议书中存在P图像和B图像。这样PB帧由一个P图像和一个B图像组成，P图像源于先前译码的P图像预测，B图像源于先前译码的P图像与当前译码的P图像的两个图像预测。选择名称B图像是因为B图像部分可能源于过去与未来图像的双向预测。预测处理在图G.1中说明。

一个改进版本的PB帧模式，称为“改进的PB帧模式”，在附件M中描述。这里保留本附件中所描述的PB帧模式，仅为了与采用改进的PB帧模式之前设计的系统兼容。由于这一理由，本附件中所描述的PB帧模式不能够同要求使用PLUSPTYPE句法的附加特性一起使用。

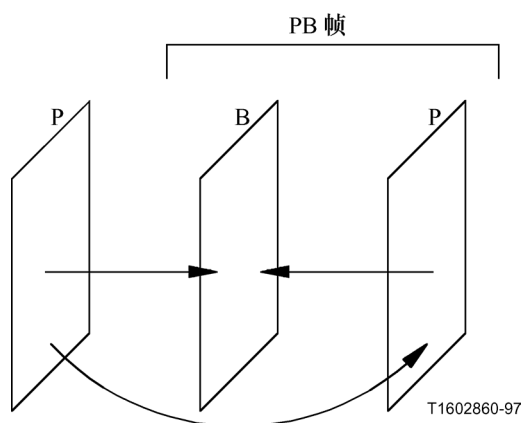


图 G.1/H.263—PB帧模式预测

G.2 PB帧和INTRA块

当使用PB帧时，编码模式INTRA有以下含义（亦见5.3.2）：

- P块为INTRA编码。
- 作为INTER块而言B块为具有预测的INTER编码。

若使用PB帧，对PTYPE指示“INTER”图像中的INTRA宏块而言也应包括运动矢量数据（MVD）。在此情形仅对B块使用该矢量。对INTRA从不使用码字MVD₂₋₄（亦见表10）。若在先进预测模式和PB帧模式中，围绕块的一个块采用INTRA模式编码，则相应的间接运动矢量不用当前块的运动矢量替代。相反，使用间接“INTRA”运动矢量。

G.3 块层

PB帧中，宏块包含12个块。首先传输作为缺省H.263模式中的6个P块数据，然后传输6个B块数据（亦见5.4）。块层结构显示在图11。若MCBPC指示宏块类型3或4，则对于宏块的任何P块，INTRADC均存在

(见表7和表8)。对B块，INTRADC不存在。若由MCBPC或CBPY指示，对P块TCOEF存在；若由CBPB指示，对B块TCOEF存在。

G.4 PB帧中B图像的矢量计算

B图像的矢量如下计算（亦见6.1.1）。假设我们有以半像素为单位的矢量分量MV准备在P图像中使用（MV代表8×8亮度块的矢量分量；若每宏块仅传输一个矢量，则对4个8×8亮度块的每一个MV都有相同值）。对B图像的预测，我们需要前向和后向两个矢量分量MV_F和MV_B。这些前向和后向矢量分量从MV推出并最后通过MVDB给出的Δ（增量）矢量增强。

- TR_D：来自上一个图像头的时域参考TR（或者当使用惯例图像时钟频率时，在改进的PB帧中扩展的时域参考ETR与时域参数TR的结合）的增量（见5.1.2）。若TR_D为负，则TR_D=TR_D+d，其中对CIF图像频率d=256，对任何惯例图像时钟频率d=1024。
- TR_B：见5.1.22。

假设MV_D是由MVDB给出的Δ矢量分量且和矢量分量MV对应。若MVDB不存在，则MV_D置零。若MVDB存在，则对宏块内的4个亮度B块的每一个使用MVDB给出的相同的MV_D。

现在使用以下公式以半像素为单位给出MV_F和MV_B：

$$\begin{aligned} MV_F &= (TR_B \times MV) / TR_D + MV_D \\ MV_B &= ((TR_B - TR_D) \times MV) / TR_D && \text{若 } MV_D = 0 \\ MV_B &= MV_F - MV && \text{若 } MV_D \neq 0 \end{aligned}$$

其中“/”代表舍位除法。假设换算恰好影响P图像和B图像的实际位置。利用MV_F值的范围受限的这一事实。MVDB的每个VLC字代表一对差值。对MV_F而言，仅该对中的一个值将产生落入到可允许范围内的值（缺省为[-16, 15.5]；在非受限运动矢量模式为[-31.5, 31.5]）。MV_F和MV_B的公式亦用于INTRA块的情形，在那里矢量数据仅用于预测B块。

对色度块，MV_F通过计算4个相应亮度MV_F矢量和并用该和除以8推出；修正产生的1/16像素分辨率的矢量分量朝向最近的半像素位置，如表F.1中所示。色度块的MV_B通过计算4个相应亮度MV_B矢量和并用该和除以8推出；修正产生的1/16像素分辨率的矢量分量朝向最近的半像素位置，如表F.1中所示。

运动矢量的水平或垂直分量的正值表明预测是由来自空间位置在被预测像素的右方或下方的参考图像中的像素组成的。

G.5 PB帧中B块预测

在本节，块代表一个8×8块。以下规程适用于亮度块以及色度块。首先，计算前向和后向矢量。假设P宏块（亮度与色度）被首先译码、重构和限幅（见6.3.2）。此宏块称为P_{REC}。基于P_{REC}及P_{REC}的预测，可计算B块预测。

B块预测有两种方式适用于块的不同部分：

- 对于后向矢量 MV_B 指向 P_{REC} 内部的像素，使用双向预测。作为使用同先前译码图像有关的 MV_F 前向预测与使用同 P_{REC} 有关的 MV_B 后向预测的一个平均可获得B块预测。计算平均通过两个预测之和除以2获得（舍位除法）。
- 对于所有其他像素，使用同先前译码图像有关的 MV_F 前向预测。

图G.2表明块的哪部分双向预测（B块的阴影部分），哪部分仅前向预测（B块的其余部分）。

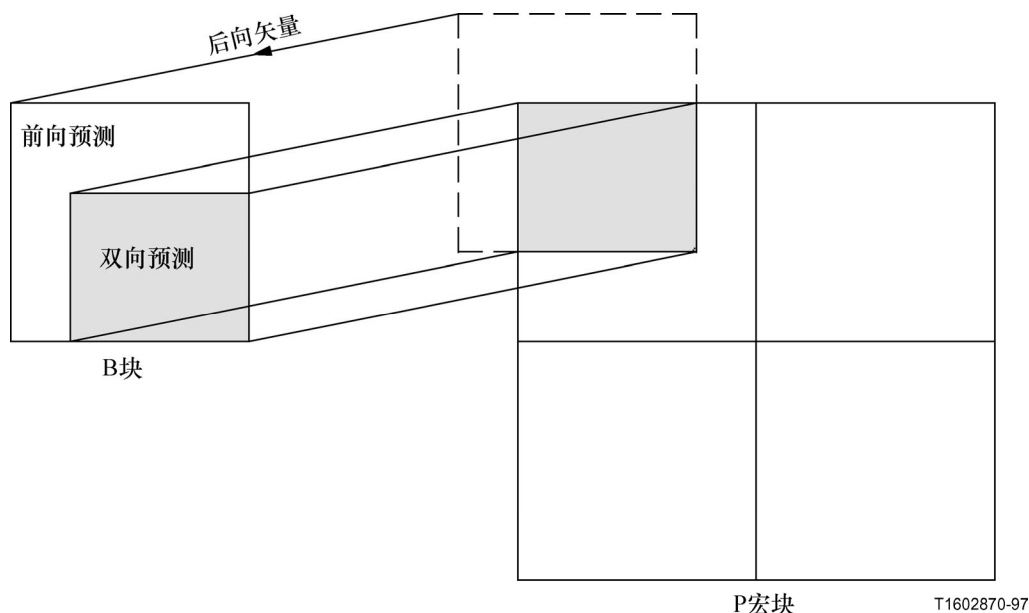


图 G.2/H.263—对B块的前向和双向预测

双向预测供后向矢量 MV_B 指向 P_{REC} 内部的像素使用。这些像素由以C语言说明的以下程序确定。

定义：

- nh 宏块内块的水平位置（0或1）。
- nv 宏块内块的垂直位置（0或1）。
- $mh(nh, nv)$ 以半像素为单位，块 (nh, nv) 的水平矢量分量。
- $mv(nh, nv)$ 以半像素为单位，块 (nh, nv) 的垂直矢量分量。
- mhc 水平色度矢量分量。
- mvc 垂直色度矢量分量。

亮度程序

```
for (nh = 0; nh <= 1; nh++) {
    for (nv = 0; nv <= 1; nv++) {
        for (i = nh * 8 + max(0, (-mh(nh,nv)+1)/2 - nh * 8);
            i <= nh * 8 + min(7,15-(mh(nh,nv)+1)/2 - nh * 8); i++) {
            for (j = nv * 8 + max(0, (-mv(nh,nv)+1)/2 - nv * 8);
                j <= nv * 8 + min(7,15-(mv(nh,nv)+1)/2 - nv * 8); j++) {
                predict pixel (i,j) bidirectionally
            }
        }
    }
}
```

色度程序

```
for (i = max(0, (-mhc+1)/2); i <= min(7,7-(mhc+1)/2); i++) {
    for (j = max(0, (-mvc+1)/2); j <= min(7,7-(mvc+1)/2); j++) {
        predict pixel (i,j) bidirectionally;
    }
}
```

预测未双向预测的像素仅采用前向预测。

附 件 H

视频编码信号的前向纠错

H.1 引言

本附件描述H.263编码视频数据传输的一个任选的前向纠错方法（代码和定帧）。此前向纠错可用于没有任何外部手段提供前向纠错的环境，例如在多路复用级或系统级。它不供ITU-T H.324建议书使用。定帧和前向纠错代码均与ITU-T H.261建议书中的相同。

H.2 纠错定帧

为允许视频数据和纠错校验信息可以被译码器标识，加入一个纠错定帧模式。此模式由8帧的复帧组成，每帧包含1比特定帧，1比特填充指示符（Fi），492比特的编码数据（或填充全1）和18比特校验（见图H.1）。对每个复帧，由8个单独帧的定帧比特构成的帧定位模式是：

$$(S_1S_2S_3S_4S_5S_6S_7S_8) = (00011011)$$

编码器可将填充指示符（Fi）置0。在此种情况下，492个连续填充比特（填充全1）用于替代492比特的编码数据。这可以供填充数据使用（见3.6）。

H.3 纠错码

纠错码是一个BCH（511，493）前向纠错码。译码器是否使用是任选的。校验计算面向包含1比特填充指示符（Fi）和492比特视频编码数据的493比特的代码进行。

生成多项式为：

$$g(x) = (x^9 + x^4 + 1)(x^9 + x^6 + x^4 + x^3 + 1)$$

例如：对011111...11（493比特）的输入数据，产生的纠错校验比特是011011010100011011（18比特）。

H.4 纠错定帧的重新锁定时间

在确认帧锁定完成之前应收到三个连续纠错帧定位码型（24比特）。译码器应当设计成纠错定帧状态改变之后34 000比特内帧锁定能够重新建立。

注一 假定重新锁定周期内，视频数据不包含三次正确状态仿真的纠错定帧序列。

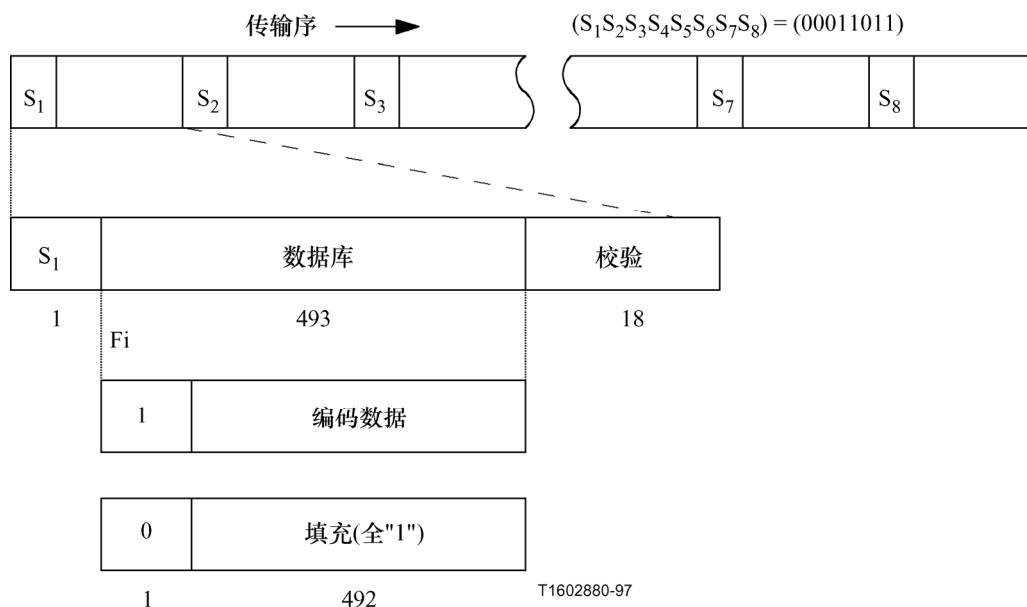


图 H.1/H.263—纠错帧

附件 I

先进的INTRA编码模式

本附件描述本建议书任选的先进的INTRA编码模式。本H.263模式能力通过外部手段标示（例如，ITU-T H.245建议书）。本模式的使用在图像帧头的PLUSPTYPE场中指示。

I.1 引言

此任选模式改变INTRA类型宏块的译码（其他类型宏块不受影响）。INTRA宏块的编码效率通过使用以下手段得以改善：

- 1) INTRA块预测使用相同分量（Y、C_B或C_R）的毗邻INTRA块；
- 2) 修正INTRA系数反量化；且
- 3) INTRA系数的单独的VLC。

特殊的INTRA编码块可以从正在译码的当前块的上方块，从正在译码的当前块的左方块，或从以上两种块来预测。存在特定的位置情况，在那里毗邻块未被INTRA编码或者不在同一视频图像分段内。块预测总是使用来自与译码块相同的亮度或色差分量（Y，C_B或C_R）数据。预测中，DC系数总以其他方式预测。AC系数的第一行可以从上方块中的那些系数预测，或者AC系数的第一列可以从左边块中的那些系数预测，或者仅DC系数可以作为来自上方块和左方块的平均来预测，如在逐个宏块基础上所标示的。剩

余AC系数从不预测。修正INTRADC系数的反量化允许变化的量化步长尺寸，而不象本建议书正文中提及的对INTRADC系数使用固定的步长尺寸。实施所有INTRA系数的反量化，无须在量化器重构空间中存在死区。

I.2 句法

当使用先进的INTRA编码模式时，变动宏块层句法如图I.1所指定的。图I.1所示的句法同5.3中定义的句法一样，除插入INTRA宏块的附加INTRA_MODE场之外。仅当MCBPC指示INTRA类型宏块（宏块类型3或4）时，INTRA_MODE才存在。使用表I.1中所示的变长码编码预测模式。每个INTRA宏块仅传输一个预测模式。

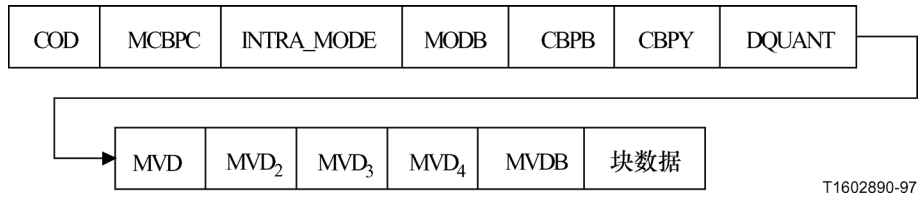


图 I.1/H.263—宏块层结构

表 I.1/H.263—INTRA—MODE的VLC

| 索引 | 预测模式 | VLC |
|----|--------------|-----|
| 0 | 0 (仅 DC) | 0 |
| 1 | 1 (垂直 DC&AC) | 10 |
| 2 | 2 (水平 DC&AC) | 11 |

I.3 译码处理

除之字形扫描外使用另外两种扫描方式。两种附加的扫描在图I.2的a)和b)部分中显示，且之字形扫描在图14显示。

| | | | | | | | |
|----|----|----|----|----|----|----|----|
| 1 | 2 | 3 | 4 | 11 | 12 | 13 | 14 |
| 5 | 6 | 9 | 10 | 18 | 17 | 16 | 15 |
| 7 | 8 | 20 | 19 | 27 | 28 | 29 | 30 |
| 21 | 22 | 25 | 26 | 31 | 32 | 33 | 34 |
| 23 | 24 | 35 | 36 | 43 | 44 | 45 | 46 |
| 37 | 38 | 41 | 42 | 47 | 48 | 49 | 50 |
| 39 | 40 | 51 | 52 | 57 | 58 | 59 | 60 |
| 53 | 54 | 55 | 56 | 61 | 62 | 63 | 64 |

a) 可选择的水平扫描

| | | | | | | | |
|----|----|----|----|----|----|----|----|
| 1 | 5 | 7 | 21 | 23 | 37 | 39 | 53 |
| 2 | 6 | 8 | 22 | 24 | 38 | 40 | 54 |
| 3 | 9 | 20 | 25 | 35 | 41 | 51 | 55 |
| 4 | 10 | 19 | 26 | 36 | 42 | 52 | 56 |
| 11 | 18 | 27 | 31 | 43 | 47 | 57 | 61 |
| 12 | 17 | 28 | 32 | 44 | 48 | 58 | 62 |
| 13 | 16 | 29 | 33 | 45 | 49 | 59 | 63 |
| 14 | 15 | 30 | 34 | 46 | 50 | 60 | 64 |

b) 可选择的垂直扫描
(如ITU-T H.262建议书中规定)

图 I.2/H.263—先进的INTRA编码的可选择的DCT扫描模型

对INTRA编码块，若预测模式=0，则对宏块中的所有块选择图14中所示的之字形扫描，否则，预测方向通常选择对宏块的扫描。

预测模式=1使用垂直毗邻块构成预测。该预测模式针对由较强水平频率内容所支配的INTRA块设计，因此垂直的毗邻块用于预测当前块的水平频率内容，对表示垂直AC内容的所有系数采用0预测。然后选择扫描模型，使用可选择的水平扫描在垂直频率之前扫描较强的水平频率。

预测模式=2使用水平毗邻块构成预测。该预测模式针对由较强垂直频率内容所支配的INTRA块设计，因此水平相邻块用于预测当前块的垂直频率内容，对表示水平AC内容的所有系数采用0预测。然后选择扫描模型使用可选择的垂直扫描在水平频率之前扫描较强的垂直频率。

对非INTRA块，8×8块的变换系数使用图14中所示的“之字形”扫描模型扫描。

对所有INTRADC和INTRAAC系数，使用单独的VLC表。该表在表I.2中指明。注意只要不使用先进的INTRA编码，则表I.2中使用的VLC码字条目同所使用的正规的TCOEF表（表16）中使用的那些VLC码字条目相同，但具有不同的LEVEL和RUN解释（无需改动的LAST）。

表 I.2/H.263—INTRA TCOEF的VLC

| 索引 | LAST | RUN | LEVEL | 比特数目 | VLC代码 |
|----|------|-----|-------|------|------------|
| 0 | 0 | 0 | 1 | 3 | 10s |
| 1 | 0 | 1 | 1 | 5 | 1111s |
| 2 | 0 | 3 | 1 | 7 | 0101 01s |
| 3 | 0 | 5 | 1 | 8 | 0010 111s |
| 4 | 0 | 7 | 1 | 9 | 0001 1111s |

表 I.2/H.263—INTRA TCOEF的VLC (续)

| 索引 | LAST | RUN | LEVEL | 比特数目 | VLC代码 |
|----|------|-----|-------|------|-----------------|
| 5 | 0 | 8 | 1 | 10 | 0001 0010 1s |
| 6 | 0 | 9 | 1 | 10 | 0001 0010 0s |
| 7 | 0 | 10 | 1 | 11 | 0000 1000 01s |
| 8 | 0 | 11 | 1 | 11 | 0000 1000 00s |
| 9 | 0 | 4 | 3 | 12 | 0000 0000 111s |
| 10 | 0 | 9 | 2 | 12 | 0000 0000 110s |
| 11 | 0 | 13 | 1 | 12 | 0000 0100 000s |
| 12 | 0 | 0 | 2 | 4 | 110s |
| 13 | 0 | 1 | 2 | 7 | 0101 00s |
| 14 | 0 | 1 | 4 | 9 | 0001 1110s |
| 15 | 0 | 1 | 5 | 11 | 0000 0011 11s |
| 16 | 0 | 1 | 6 | 12 | 0000 0100 001s |
| 17 | 0 | 1 | 7 | 13 | 0000 0101 0000s |
| 18 | 0 | 0 | 3 | 5 | 1110s |
| 19 | 0 | 3 | 2 | 9 | 0001 1101s |
| 20 | 0 | 2 | 3 | 11 | 0000 0011 10s |
| 21 | 0 | 3 | 4 | 13 | 0000 0101 0001s |
| 22 | 0 | 0 | 5 | 6 | 0110 1s |
| 23 | 0 | 4 | 2 | 10 | 0001 0001 1s |
| 24 | 0 | 3 | 3 | 11 | 0000 0011 01s |
| 25 | 0 | 0 | 4 | 6 | 0110 0s |
| 26 | 0 | 5 | 2 | 10 | 0001 0001 0s |
| 27 | 0 | 5 | 3 | 13 | 0000 0101 0010s |
| 28 | 0 | 2 | 1 | 6 | 0101 1s |
| 29 | 0 | 6 | 2 | 11 | 0000 0011 00s |
| 30 | 0 | 0 | 25 | 13 | 0000 0101 0011s |
| 31 | 0 | 4 | 1 | 7 | 0100 11s |
| 32 | 0 | 7 | 2 | 11 | 0000 0010 11s |
| 33 | 0 | 0 | 24 | 13 | 0000 0101 0100s |
| 34 | 0 | 0 | 8 | 7 | 0100 10s |
| 35 | 0 | 8 | 2 | 11 | 0000 0010 10s |
| 36 | 0 | 0 | 7 | 7 | 0100 01s |
| 37 | 0 | 2 | 4 | 11 | 0000 0010 01s |
| 38 | 0 | 0 | 6 | 7 | 0100 00s |
| 39 | 0 | 12 | 1 | 11 | 0000 0010 00s |
| 40 | 0 | 0 | 9 | 8 | 0010 110s |
| 41 | 0 | 0 | 23 | 13 | 0000 0101 0101s |
| 42 | 0 | 2 | 2 | 8 | 0010 101s |

表 I.2/H.263—INTRA TCOEF的VLC (续)

| 索引 | LAST | RUN | LEVEL | 比特数目 | VLC代码 |
|----|------|-----|-------|------|-----------------|
| 43 | 0 | 1 | 3 | 8 | 0010 100s |
| 44 | 0 | 6 | 1 | 9 | 0001 1100s |
| 45 | 0 | 0 | 10 | 9 | 0001 1011s |
| 46 | 0 | 0 | 12 | 10 | 0001 0000 1s |
| 47 | 0 | 0 | 11 | 10 | 0001 0000 0s |
| 48 | 0 | 0 | 18 | 10 | 0000 1111 1s |
| 49 | 0 | 0 | 17 | 10 | 0000 1111 0s |
| 50 | 0 | 0 | 16 | 10 | 0000 1110 1s |
| 51 | 0 | 0 | 15 | 10 | 0000 1110 0s |
| 52 | 0 | 0 | 14 | 10 | 0000 1101 1s |
| 53 | 0 | 0 | 13 | 10 | 0000 1101 0s |
| 54 | 0 | 0 | 20 | 12 | 0000 0100 010s |
| 55 | 0 | 0 | 19 | 12 | 0000 0100 011s |
| 56 | 0 | 0 | 22 | 13 | 0000 0101 0110s |
| 57 | 0 | 0 | 21 | 13 | 0000 0101 0111s |
| 58 | 1 | 0 | 1 | 5 | 0111s |
| 59 | 1 | 14 | 1 | 10 | 0000 1100 1s |
| 60 | 1 | 20 | 1 | 12 | 0000 0000 101s |
| 61 | 1 | 1 | 1 | 7 | 0011 11s |
| 62 | 1 | 19 | 1 | 12 | 0000 0000 100s |
| 63 | 1 | 2 | 1 | 7 | 0011 10s |
| 64 | 1 | 3 | 1 | 7 | 0011 01s |
| 65 | 1 | 0 | 2 | 7 | 0011 00s |
| 66 | 1 | 5 | 1 | 8 | 0010 011s |
| 67 | 1 | 6 | 1 | 8 | 0010 010s |
| 68 | 1 | 4 | 1 | 8 | 0010 001s |
| 69 | 1 | 0 | 3 | 8 | 0010 000s |
| 70 | 1 | 9 | 1 | 9 | 0001 1010s |
| 71 | 1 | 10 | 1 | 9 | 0001 1001s |
| 72 | 1 | 11 | 1 | 9 | 0001 1000s |
| 73 | 1 | 12 | 1 | 9 | 0001 0111s |
| 74 | 1 | 13 | 1 | 9 | 0001 0110s |
| 75 | 1 | 8 | 1 | 9 | 0001 0101s |
| 76 | 1 | 7 | 1 | 9 | 0001 0100s |
| 77 | 1 | 0 | 4 | 9 | 0001 0011s |
| 78 | 1 | 17 | 1 | 10 | 0000 1100 0s |
| 79 | 1 | 18 | 1 | 10 | 0000 1011 1s |
| 80 | 1 | 16 | 1 | 10 | 0000 1011 0s |

表 I.2/H.263—INTRA TCOEF的VLC (续完)

| 索引 | LAST | RUN | LEVEL | 比特数目 | VLC代码 |
|-----|--------|-----|-------|------|-----------------|
| 81 | 1 | 15 | 1 | 10 | 0000 1010 1s |
| 82 | 1 | 2 | 2 | 10 | 0000 1010 0s |
| 83 | 1 | 1 | 2 | 10 | 0000 1001 1s |
| 84 | 1 | 0 | 6 | 10 | 0000 1001 0s |
| 85 | 1 | 0 | 5 | 10 | 0000 1000 1s |
| 86 | 1 | 4 | 2 | 11 | 0000 0001 11s |
| 87 | 1 | 3 | 2 | 11 | 0000 0001 10s |
| 88 | 1 | 1 | 3 | 11 | 0000 0001 01s |
| 89 | 1 | 0 | 7 | 11 | 0000 0001 00s |
| 90 | 1 | 2 | 3 | 12 | 0000 0100 100s |
| 91 | 1 | 1 | 4 | 12 | 0000 0100 101s |
| 92 | 1 | 0 | 9 | 12 | 0000 0100 110s |
| 93 | 1 | 0 | 8 | 12 | 0000 0100 111s |
| 94 | 1 | 21 | 1 | 13 | 0000 0101 1000s |
| 95 | 1 | 22 | 1 | 13 | 0000 0101 1001s |
| 96 | 1 | 23 | 1 | 13 | 0000 0101 1010s |
| 97 | 1 | 7 | 2 | 13 | 0000 0101 1011s |
| 98 | 1 | 6 | 2 | 13 | 0000 0101 1100s |
| 99 | 1 | 5 | 2 | 13 | 0000 0101 1101s |
| 100 | 1 | 3 | 3 | 13 | 0000 0101 1110s |
| 101 | 1 | 0 | 10 | 13 | 0000 0101 1111s |
| 102 | ESCAPE | | | 7 | 0000 011 |

取决于INTRA_MODE的值，或一个或8个系数为预测残差，如下所述必须加到预测值上。图I.3显示三个 8×8 块相同分量（Y， C_B 或 C_R ）的最终重构DCT电平，称之为 $RecA'(u,v)$ 、 $RecB'(u,v)$ 及 $RecC'(u,v)$ ，其中u和v分别是列（水平）和行（垂直）的索引号。重构处理不同于6.2.1中描述的处理过程。INTRADC残差通过使用可变步长尺寸不同地重构，与使用表15相对照，然后预测值加上该残差值获得最终的系数重构值。非INTRADC的INTRA系数也采用与6.2.1中不同的方式重构使用无“死区”的重构空间并在某些情形中添加预测值获得最终的系数重构值。该块可以包含DC和AC两种预测残差。

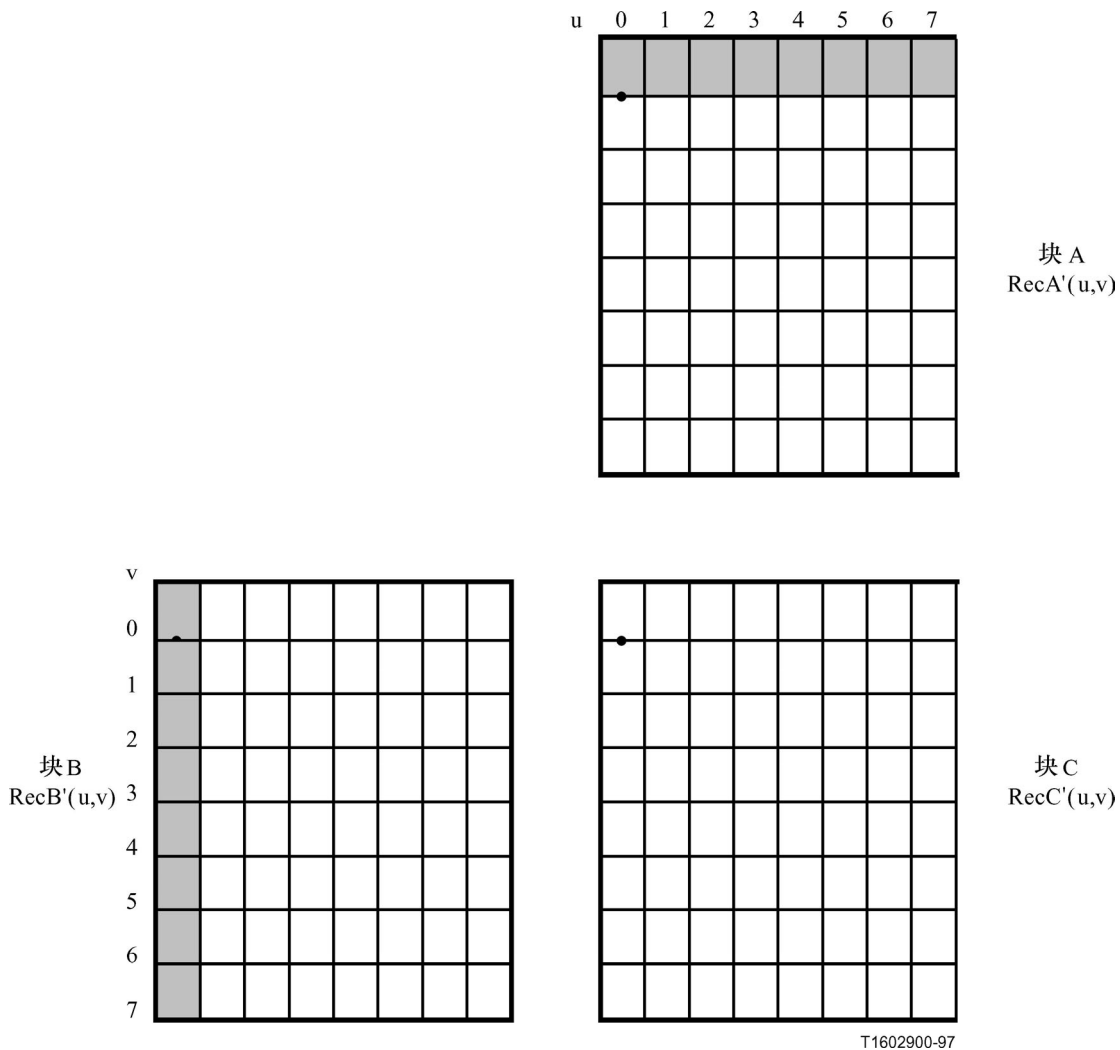


图 I.3/H.263—DCT域中三个相邻块

当使用先进的INTRA编码时，MCBPC和CBPY的定义被改变。就MCBPC和CBPY而论，当先进的INTRA编码使用时，INTRADC变换系数不再作为单独的情形来对待，而采用与AC系数相同的方式来替代处理。这意味着零INTRADC将不被作为LEVEL编码，而将简单的增加跟随AC系数的游程。

对于改进的PB帧的B图像部分（见附件M）而言，反量化过程通过使用先进的INTRA编码模式不改变。

定义RecC(u,v)为当前块的重构系数残差。对所有的INTRA系数，重构残差值通过下式获得：

$$\text{RecC}(u,v) = 2 * \text{QUANT} * \text{LEVEL}(u,v) \quad u = 0, \dots, 7, v = 0, \dots, 7.$$

注一 在以上公式中，LEVEL(u,v)代表具有幅度和符号的量。

定义RecC'(u,v)为当前块的最终的重构系数值（预测，如下所述的奇数化及限幅调整后）。最终重构的系数值RecC'(u,v)通过把RecC(u,v)添加到适当的预测上如INTRA-MODE场中所标示的，若DC系数奇数化必须改变最低有效比特及限幅来恢复。

$RecA'(u,v)$ 代表当前块直接上方块的最终重构的系数值。 $RecB'(u,v)$ 代表当前块直接左方块的最后重构的系数值。

在块C的系数值预测中使用块A和块B的重构系数值的能力依赖于块A和块B是否与块C在同一视频图像分段中。定义A块与另一个块“在相同视频图像分段中”仅当以下条件被满足：

- 1) 相关块在该图像的边界内，且
- 2) 若不在截面结构模式中（附件K），则相关块或是在相同的GOB之内或者对当前GOB而言无任何GOB头存在，且
- 3) 若在截面结构模式中，相关块在相同截面内。

即将译码的块C仅从与块C在同一视频图像分段内的INTRA块中预测，如下所示。

若使用预测模式=0（仅DC预测）并且块A和块B均为与块C在同一视频图像分段内的，INTRA块则块C的DC系数预测来自块A和块B的DC系数的平均（具有舍位截断）。若仅仅两个块A和块B中的一个是与块C在同一视频图像分段内的INTRA块，那么仅使用两个块中的这一个块的DC系数作为预测模式=0的预测值。若两个块A和块B都不是与块C在同一视频图像分段内的INTRA块，那么该预测使用值1024作为DC系数的预测值。

若预测模式=1或2（垂直DC&AC或水平DC&AC预测）且参考块（块A或块B）不是与块C在同一视频图像分段内的INTRA块，那么该预测使用值1024作为DC系数的预测值并且值0作为C块的AC系数的预测值。

为了极小化IDCT失配误差的影响“使奇数化”的处理适用于DC系数。系数的某些值能够引起不同的IDCT运算之间的舍入误差失配，特别是(0,0)、(0,4)、(4,0)及(4,4)系数的某些值。例如，某些整数k的 $8k+4$ 的DC系数导致反变换块具有常量 $k+0.5$ ，对此该轻微误差在不同运算上能够引起不同方向的舍入。

定义ClipAC()函数指示限幅-2048到2047的取值范围。定义ClipDC()函数指示限幅0到2047的取值范围。定义OddifyClipDC(X) 函数为

```
If (x is even) {
    result = clipDC(x+1)
} else {
    result = clipDC(x)
}
```

每个INTRA预测模式的重构如下指定，在那里运算符“/”定义为舍位除法。

模式0：仅DC预测

```
RecC'(u,v) = clipAC(Rec((u,v)) (u,v) ≠ (0,0), u=0,...,7, v=0,...,7
若 (块A和块B均为INTRA编码且都与块C在同一视频图像分段内)
    tempDC = RecC(0,0) + ( RecA'(0,0) + RecB'(0,0) ) / 2
} else {
若 (块A为INTRA编码且与块C在同一视频图像分段内)
    tempDC = RecC(0,0) + RecA'(0,0)
```

```

    } else {
若 (块B为INTRA编码且与块C在同一视频图像分段内)
        tempDC = RecC(0,0) + RecB'(0,0)
    } else {
        tempDC = RecC(0,0) + 1024
    }
}
}
RecC'(0,0) = oddifyclipDC( tempDC )

```

模式1: 来自上方块的DC和AC预测

```

若 (块A为INTRA编码且与块C在同一视频图像分段内)
    tempDC = RecC(0,0) + RecA'(0,0)
    RecC'(u,0) = clipAC( RecC(u,0) + RecA'(u,0) )    u = 1, ..., 7,
    RecC'(u,v) = clipAC( RecC(u,v) )                u = 0, ..., 7, v = 1, ..., 7.
} else {
    tempDC = RecC(0,0) + 1024
    RecC'(u,v) = clipAC( RecC(u,v) )                (u,v)≠(0,0), u=0,...,7, v = 0,...,7
}
RecC'(0,0) = oddifyclipDC( tempDC )

```

模式2: 来自左方块的DC和AC预测

```

若 (块B为INTRA编码且块C在同一视频图像分段内)
    tempDC = RecC(0,0) + RecB'(0,0)
    RecC'(0,v) = clipAC( RecC(0,v) + RecB'(0,v) )    v = 1, ..., 7,
    RecC'(u,v) = clipAC( RecC(u,v) )                u = 1, ..., 7, v = 0, ..., 7.
} else {
    tempDC = RecC(0,0) + 1024
    RecC'(u,v) = clipAC( RecC(u,v) )                (u,v)≠(0,0), u = 0, ..., 7,
                                                    v = 0, ..., 7
}
RecC'(0,0) = oddifyclipDC( tempDC )

```

附件 J

分块滤波器模式

J.1 引言

本附件描述编码环路内任选的块边缘滤波器的使用。块边缘滤波器的主要作用是减少分块效应。在 8×8 块边缘上实施滤波。运动矢量可以有 8×8 或 16×16 分辨率（见J.2）。本附件描述的处理仅适用于P-、I-、EP-或EI图像或改进的PB帧的P图像部分（对标准化而言B图像或改进的PB帧的B图像部分的可能滤波不是问题；然而，为改进图像质量推荐某种类型的滤波）。本模式能力经由外部手段标示（例如ITU-T H.245建议书）。本模式的使用在图像帧头的PLUSPTYPE场中指示。

注 — 当在编码器和译码器中采用附件J的小量化步长尺寸而不采用附件W IDCT时，可加重IDCT的失配问题。舍入误差失配量甚至被附件J滤波处理放大。编码器应避免此问题，例如通过在量化步长尺寸非常小时不采用附件J的方式。

J.2 同UMV和AP模式的关系（附件D和F）

分块滤波器模式的使用在图像质量上有类似附件F中定义的对运动补偿（OBMC）的影响，当单独使用时。当两种技术一起使用时，在图像质量上可以获得进一步改善。先进的预测模式（亦见附件F）由三种单元组成：

- 1) 每宏块4个运动矢量如F.2中定义的；
- 2) 亮度的交叠运动补偿如F.3中定义的；
- 3) 图像边界上的运动矢量如D.1中定义的；

为使分块滤波器模式能够提供最大的性能，出于复杂度考虑可以阻止使用先进预测模式的OBMC部分，则分块滤波器模式包括使用每宏块4个运动矢量及图像边界上运动矢量的能力。

概括地讲，附件D、F和J中定义的三种选择包括以下5种编码单元：

- 1) 图像边界上的运动矢量（D.1）；
- 2) 运动矢量范围的外延（D.2）；
- 3) 每宏块4个运动矢量（F.2）；
- 4) 亮度的交叠运动补偿（F.3）；
- 5) 分块边缘滤波器（J.3）；

表J.1指明五种单元中那些单元的启动取决于附件D、F和J中定义的三种选择的开启。

表 J.1/H.263—UMV、AP和DF模式的特征单元

| 非受限运动矢量模式 | 先进的预测模式 | 分块滤波器模式 | 图像边界上的运动矢量 | 运动矢量范围的外延 | 每宏块4个运动矢量 | 亮度的交叠运动补偿 | 分块边缘滤波器 |
|-----------|---------|---------|------------|-----------|-----------|-----------|---------|
| OFF | OFF | OFF | OFF | OFF | OFF | OFF | OFF |
| OFF | OFF | ON | ON | OFF | ON | OFF | ON |
| OFF | ON | OFF | ON | OFF | ON | ON | OFF |
| OFF | ON | ON | ON | OFF | ON | ON | ON |
| ON | OFF | OFF | ON | ON | OFF | OFF | OFF |
| ON | OFF | ON | ON | ON | ON | OFF | ON |
| ON | ON | OFF | ON | ON | ON | ON | OFF |
| ON | ON | ON | ON | ON | ON | ON | ON |

J.3 分块边缘滤波器定义

在编码器方同样亦在译码器方滤波器操作跨越8×8块边缘实施。重构的图像数据（预测与重构的预测误差之和）被限幅到0到255的范围如6.3.2中所述。然后使用滤波，改变为未来预测即将存储到图像缓冲器中的图像。滤波运算包括附加的限幅以确保产生的像素值驻留在0到255的范围内。当独立分段译码模式使用时不实施任何跨越图像边缘的滤波，当截面结构模式使用时不实施任何跨越截面边缘的滤波（见附件K和R），或者当截面结构模式不使用时不实施任何跨越具有GOB头部存在的GOB顶部边界的滤波（见附件R）。色度同亮度数据一样滤波。

当本附件描述的模式同附件M的改进的PB帧模式一起使用时，限幅操作之后但在分块边缘滤波器操作之前，B宏块的后向预测根据重构的P宏块（G.5中称为P_{REC}）。B宏块的前向预测根据先前译码图像的滤波修正(供P宏块预测使用的同一图像数据)。

分块滤波器操作使用重构图像的水平或垂直线上的4个(限幅的)像素值的集,表示为A、B、C和D，其中A和B属于一个块称之为块1，C和D属于相邻块称之为块2，块2在块1的右方或下方。图J.1显示这些像素位置的实例。

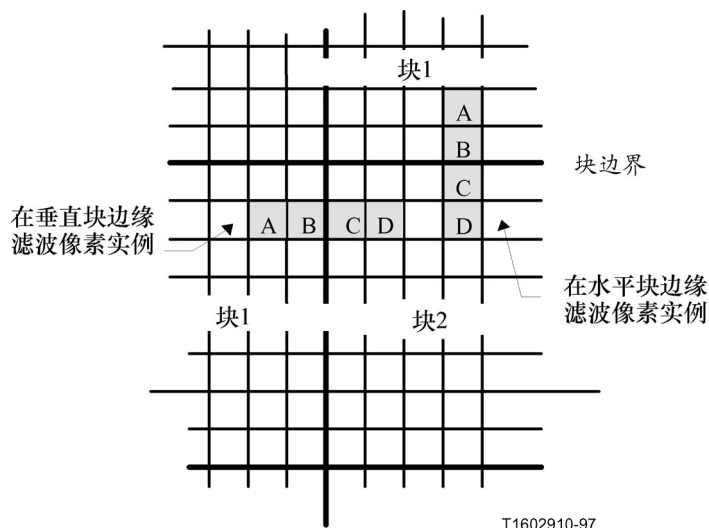


图 J.1/H.263—滤波像素的位置实例（略）

为了采用跨越特殊边缘的滤波器，以下条件的一个或两个必须满足：

- 条件1：块1属于编码的宏块（COD=0 || MB类型=INTRA）；或
- 条件2：块2属于编码的宏块（COD=0 || MB类型=INTRA）。

若滤波预期跨越边缘使用，则A、B、C和D须用A₁、B₁、C₁、D₁替代，其中：

$$B_1 = \text{clip}(B + d_1)$$

$$C_1 = \text{clip}(C - d_1)$$

$$A_1 = A - d_2$$

$$D_1 = D + d_2$$

$$d = (A - 4B + 4C - D) / 8$$

$$d_1 = \text{UpDownRamp}(d, \text{STRENGTH})$$

$$d_2 = \text{clipd1}((A - D) / 4, d_1/2)$$

$$\text{UpDownRamp}(x, \text{STRENGTH}) = \text{SIGN}(x) * (\text{MAX}(0, \text{abs}(x) - \text{MAX}(0, 2 * (\text{abs}(x) - \text{STRENGTH}))))$$

STRENGTH依赖于QUANT并决定滤波的数量。STRENGTH和QUANT间的关系在表J.2中给出。

若块2属于编码的宏块则QUANT=供块2使用的量化参数，或

若块2不属于编码的宏块（而块1属于）则QUANT=供块1使用的量化参数。

表 J.2/H.263—滤波器的QUANT和STRENGTH间的关系

| QUANT | STRENGTH | QUANT | STRENGTH |
|-------|----------|-------|----------|
| 1 | 1 | 17 | 8 |
| 2 | 1 | 18 | 8 |
| 3 | 2 | 19 | 8 |
| 4 | 2 | 20 | 9 |
| 5 | 3 | 21 | 9 |
| 6 | 3 | 22 | 9 |
| 7 | 4 | 23 | 10 |
| 8 | 4 | 24 | 10 |
| 9 | 4 | 25 | 10 |
| 10 | 5 | 26 | 11 |
| 11 | 5 | 27 | 11 |
| 12 | 6 | 28 | 11 |
| 13 | 6 | 29 | 12 |
| 14 | 7 | 30 | 12 |
| 15 | 7 | 31 | 12 |
| 16 | 7 | | |

函数Clip (x) 依照6.3.2定义，函数Clip d1 (x, lim) 限幅x到±abs (lim) 的范围。符号“/”代表舍位除法指向0。

图J.2显示作为d的函数d1的值如何变化。作为结果因此仅当d小于2*STRENGTH（且不等于0）时滤波器才有效率。这是防止图像内容中健全真实边缘的滤波。然而，若简约分辨率更新模式使用，则STRENGTH置于无限，作为结果因此d1的值总是等于d的值（见Q.7.2）。

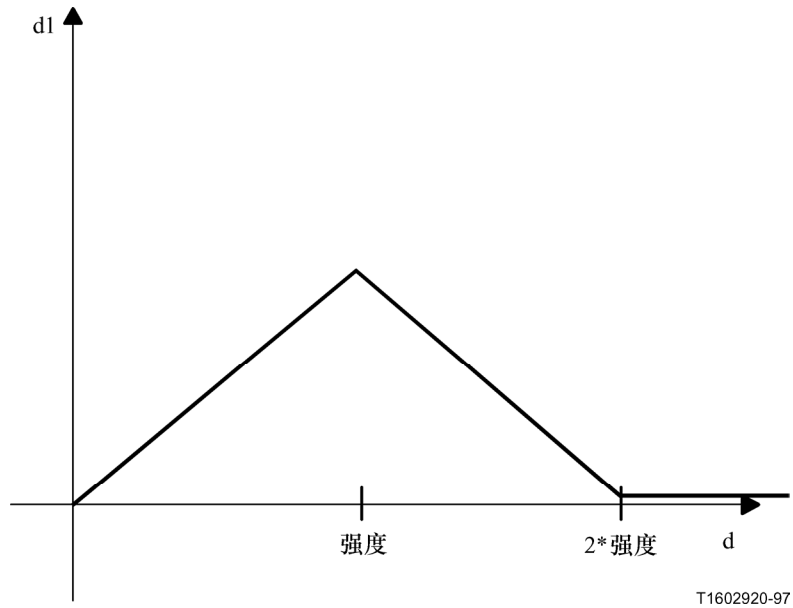


图 J.2/H.263—分块滤波器模式的参数d与参数d1的函数关系

参数d1定义的设计确保编码器与译码器之间任何小的失配将保持很小并且将不建立在视频图像序列的多幅图像上。对于采用简单切换滤波器通或断的情况，这将是一个问题，因为d的仅±1的失配就能够引起滤波器在编码器方切换到通而在译码器方切换到断，或刚好相反。

由于舍入影响，实施滤波的边缘顺序必须指明。

跨越水平边缘的滤波：

基本上假设该处理首先发生。进一步明确，在跨越水平边缘滤波中所使用的。像素 $\begin{bmatrix} A \\ B \\ C \\ D \end{bmatrix}$ 应未曾受到先

前的跨越垂直边缘的滤波影响。

跨越垂直边缘的滤波：

使用像素（A，B，C，D）跨越垂直边缘滤波之前，由跨越水平边缘滤波所引起的像素（A，B，C，D）的所有修正应已经发生。

注意：若参与滤波处理的一个或多个像素（A，B，C，D）在图像外部，则无任何滤波发生。同样，若独立分段译码模式在使用（见附件R）且参与滤波处理的一个或多个像素（A，B，C，D）在不同的视频图像分段内（见I.3有关块被认为是在同一视频图像分段内的定义），则无任何滤波实施。

附 件 K

截面结构模式

K.1 引言

本附件描述本建议书任选的截面结构模式。本ITU-T H.245建议书模式能力通过外部手段标示（例如ITU-T H.245建议书）。本模式的使用在图像帧头的PLUSPTYPE场中指明。为在众多环境中方便理想的使用，本模式包括两个子模式，它们所具有的能力也可以通过外部手段标示（例如，ITU-T H.245建议书）。这两个子模式用于指示矩形截面是否使用，和/或截面是否将按顺序传输或以任意次序发送。

截面定义为截面头部后随按扫描顺序的连续的宏块。一个例外是图像的比特流中即时后随图像起始码的截面，（该截面未必是用宏块0起始的截面）。在此情形仅截面头的部分被传输如K.2中所述。截面层定义视频图像分段，而且在此任选模式中使用截面层代替GOB层。截面视频图像分段在图像中宏块边界上起始并且包含许多宏块。同一图像内不同截面应不互相交叠，并且每个宏块应属于且仅属于一个截面。

该模式包含两个子模式，在图像帧头的SSS场中标示：

- 1) 矩形截面子模式（RS）：当RS使用时，截面应占据以宏块为单位由截面头中SWI参数指示宽度的矩形区域，并在该矩形区域内包括按扫描次序取样的许多宏块。当矩形截面子模式不使用时，SWI场在截面头中不存在并且截面包含作为一个整体的按扫描次序排列的图像内的许多宏块。
- 2) 任意截面排序子模式（ASO）：当ASO使用时，比特流内截面可以按任何次序出现。当ASO不使用时，截面必须按（唯一的）排序发送，对截面排序而言截面头中的MBA场伴随着图像中从每个截面到每个后续截面的顺序单调递增。

比特流内为了允许截面头部位置充当比特误差与分组丢失恢复的再同步位置点以及为了允许图像内的失序截面译码，截面边界采用与简单宏块边界不同的处理。这样，无任何数据从属性能够跨越当前图像内的截面边界，除分块滤波器模式外，当该模式使用而独立分段译码模式不使用时，滤波器跨越图像中的块边界。然而，截面内运动矢量能够引起跨越为预测起见所使用的参考图像中的截面边界的数据从属性，除非任选的独立分段译码模式在使用。

采纳以下规则确保截面边界位置能够充当再同步位置点以及确保截面能够失序发送而不引起附加的译码延迟：

- 1) 运动矢量的预测值同似乎GOB头存在时的预测值一样（见6.1.1），对截面内运动矢量的预测值而言防止使用当前截面外部块的运动矢量。
- 2) 先进的INTRA编码模式（见附件I）把截面边界当做它是有关INTRA块DCT系数的预测值的使用图像边界来处理。

- 3) 在先进预测模式内的交叠块运动补偿使用的间接运动矢量的分配亦禁止把当前截面外部块的运动矢量作为间接运动矢量使用。（见F.3）

K.2 截面层结构

截面层结构的句法显示在图K.1，对所有截面除图像比特流中即时跟随图像起始码的截面外。对于跟随图像起始码的截面，仅包含仿真预防比特（SEPB1，SEPB3和有条件的SEPB2如以下所示）、MBA场、以及在RS子模式时也包括SWI。

| | | | | | | | | | | |
|-------|-----|-------|------|-----|-------|--------|-----|-------|------|---------------|
| SSTUF | SSC | SEPB1 | SSBI | MBA | SEPB2 | SQUANT | SWI | SEPB3 | GFID | Macroblock 数据 |
|-------|-----|-------|------|-----|-------|--------|-----|-------|------|---------------|

图 K.1/H.263—截面层结构

对GFID参见5.2.5，对宏块层的描述参见5.3。

K.2.1 填充（SSTUF）（可变长度）

可变长度码字由少于8个比特组成。无论何时只需要SSC码字之前编码器应直接插入该码字以确保SSC的字节定位。若SSTUF存在，SSTUF的最后比特应是字节的最后（最低有效）比特，以使SSC码字的起始被字节定位。应设计译码器丢弃SSTUF。注意到SSTUF内0供填充使用。

K.2.2 截面起始码（SSC）（17比特）

17比特字。它的值为0000 0000 0000 0000 1。截面起始码应字节定位。通过起始码前插入SSTUF可实现字节定位以致起始码的首比特称为字节的第一（最高有效）比特。对于跟随图像起始码的截面，截面起始码不存在。

K.2.3 截面仿真预防比特1（SEPB1）（1比特）

单个比特总有值“1”，它被包括防止起始码仿真。

K.2.4 截面子一比特流指示符（SSBI）（4比特）

仅当图像帧头中CPM=1时4比特长度的码字才存在。对连续出现多点和视频多路复用操作的截面，SSBI指示子比特流编号，如附件C中所述。表K.1中显示从SSBI值到子比特流编号的映射。对跟随图像起始码的截面，SSBI不存在。

表 K.1/H.263—SSBI值和相关的子比特流编号

| 子比特流编号 | SSBI场值 | 仿真的GN值 |
|--------|--------|--------|
| 0 | 1001 | 25 |
| 1 | 1010 | 26 |
| 2 | 1011 | 27 |
| 3 | 1101 | 29 |

K.2.5 宏块地址 (MBA) (5/6/7/9/11/12/13/14比特)

具有长度依赖当前图像尺寸及简约分辨率更新模式是否生效 (见附件Q) 的码字。比特为当前截面中第一个宏块的宏块编号的二进制表示, 宏块编号按照扫描顺序从图像的起始点计数, 起始于具有宏块编号0的图像最左上角。MBA唯一地标识当前截面起始于图像中的哪一个宏块。该码字的码字长度在表K.2中提供。对于惯例图像尺寸而言, 场宽度由表中的第一条目给出, 具有大于或等于的宏块编号, 并且最大值为当前图像中宏块编号减一。当在简约分辨率更新模式时, 有关的图像尺寸为较低的分辨率更新图像尺寸而不是在图像帧头中指示的图像尺寸 (见附件Q)。

表 K.2/H.263—MBA参数的规格

| 图像格式 | 缺省 | | RRU模式 | |
|-----------|------|-----|-------|-----|
| | 最大值 | 场宽度 | 最大值 | 场宽度 |
| 子 QCIF | 47 | 6 | 11 | 5 |
| QCIF | 98 | 7 | 29 | 6 |
| CIF | 395 | 9 | 98 | 7 |
| 4CIF | 1583 | 11 | 395 | 9 |
| 16CIF | 6335 | 13 | 1583 | 11 |
| 2048×1152 | 9215 | 14 | 2303 | 12 |

K.2.6 截面仿真预防比特2 (SEPB2) (1比特)

单个比特总有值“1”。某种条件下被包括以防止起始码仿真。对于不是跟随图像起始码的截面, 仅当MBA场宽大于11比特并且在图像帧头中CPM=“0”时, 或者MBA场宽大于9比特并且图像帧头中CPM=“1”时, SEPB2才包括。对跟随图像起始码的截面, 仅当矩形截面子模式使用时, SEPB2才包括。

K.2.7 量化器信息 (SQUANT) (5比特)

5比特的固定长度码字指示供截面使用的量化器QUANT直至由任何随后的DQUANT更新时为止。码字为QUANT值的自然二进制表示, 具有半步长尺寸, 范围从1到31。对于跟随图像起始码的截面, SQUANT不存在。

K.2.8 以宏块表示的截面宽度指示 (SWI) (3/4/5/6/7比特)

仅当矩形截面子模式激活时码字才存在, 并且具有依赖于当前图像尺寸以及简约分辨率更新模式是否激活的长度, 如表K.3中所示。对惯例图像尺寸而言, 场宽由宽度上大于或等于该格式的下一个标准格式尺寸给出 (QCIF、CIF …), 以及最大值为跨越图像的宏块总数减一。该表的最后一行指示图像尺寸宽度大于16CIF的场宽。SWI涉及具有由MBA指示其首宏块 (左上) 的当前矩形截面的宽度。实际截面宽度的计算由下式给出:

$$\text{实际截面宽度} = \text{SWI} + 1$$

在简约分辨率更新模式中，有关的图像尺寸为更新信息的较低的分辨率图像尺寸，而不是图像帧头中指示的图像尺寸。

表 K.3/H.263—SWI参数规格

| 图像格式 | 缺省 | | RRU模式 | |
|------------------|-----|-----|-------|-----|
| | 最大值 | 场宽度 | 最大值 | 场宽度 |
| 子 QCIF | 7 | 4 | 3 | 3 |
| QCIF | 10 | 4 | 5 | 3 |
| CIF | 21 | 5 | 10 | 4 |
| 4CIF | 43 | 6 | 21 | 5 |
| 16CIF | 87 | 7 | 43 | 6 |
| 1412…2048 像素宽 | 127 | 7 | 63 | 6 |

K.2.9 截面仿真预防比特3 (SEPB3) (1比特)

为预防起始码仿真，总有值“1”的单个比特。

附 件 L

增补的增强信息规范

L.1 引言

本附件描述本建议书图像层的PSUPP场中发送的增补的增强信息格式。提供本附件中描述的任何或所有增强能力的译码器能力可以通过外部手段标示（例如，ITU-T H.245建议书）。不提供增强能力的译码器可以简单地丢弃比特流中出现的任何PSUPP信息比特。该增补的增强信息的存在在PEI中指示，并且一个附加的PEI比特插入PSUPP数据的每个字节之间，如5.1.24和5.1.25中所描述的。

本附件中，“译码的图像”和“显示的图像”之间存在差别。出于本附件的考虑，“显示的图像”是指具有与当前图像所指定的图像格式相同的图像，其中当前的图像格式通过视频比特流的图像层句法指定。如本附件所描述的，构造“显示的图像”来自译码的图像、先前显示的图像、这里所描述的增补的增强信息、以及在某些情形部分地来自外部控制的背景图像。

L.2 PSUPP格式

PSUPP数据由4比特功能类型指示FTYPE、后随4比特参数数据尺寸规格DSIZE，后随DSIZE字节的功参数数据、任选地后随另一个功能类型指示，如此继续。一个功能类型指示值定义为一个转义代码提供对未来的可扩展能力以允许多于15种不同功能的定义。接收到它不支持的功能类型指示的译码器能够丢弃那个功能的功能参数数据然后对其可以支持的后续的功能类型指示校核。定义的FTYPE值在表L.1中显示。

表 L.1/H.263—FTYPE 功能类型值

| | |
|----|--------------|
| 0 | 保留的 |
| 1 | 无事可做 |
| 2 | 全图像冻结请求 |
| 3 | 部分图像冻结请求 |
| 4 | 尺寸再现部分图像冻结请求 |
| 5 | 部分图像冻结释放请求 |
| 6 | 全图像快拍标记 |
| 7 | 部分图像快拍标记 |
| 8 | 视频时间分段起始标记 |
| 9 | 视频时间分段结束标记 |
| 10 | 逐行精细分段起始标记 |
| 11 | 逐行精细分段结束标记 |
| 12 | 色度键控信息 |
| 13 | 保留的 |
| 14 | 保留的 |
| 15 | 扩展的功能类型 |

L.3 无事可做

无事可做功能不要求任何行动。该功能用于防止起始码仿真，每当先前的PSUPP字节的最后字节的最后5个或更多的比特全为零，以及无任何另外的PSUPP功能请求发送时，无事可做功能应插入到PSUPP中以防止可能的起始码仿真。当仿真功能不被先前句子中表示的规则所需要时，无事可做功能也可以发送。无事可做的DSIZE应为零。

L.4 全图像冻结请求

全图像冻结请求功能指示全部的先前显示的视频图像的内容应保持不变，无需使用当前译码的图像内容更新显示的图像。显示的图像应保持不变直到当前的PTYPE或后续的PTYPE中的冻结图像释放比特置于1时为止，或者直到暂停发生为止，无论那种情况应该首先发生的为准。预定暂停5秒或5幅图像之后该请求将失效，无论那种情况应以较长的时间周期为准。通过在该暂停周期期满时限或之前发布另一个全图像冻结请求，暂停可以被阻止（例如：自发布以来在具有时间参考指示时间间隔大于或等于5秒的第一图像的头部中重复该请求，或者发布之后在第5幅图像的头部中重复该请求）。全图像冻结请求功能的DSIZE应为0。

L.5 部分图像冻结请求

部分图像冻结请求功能指示先前显示的视频图像的特有的矩形区域的内容将保持不变，无须使用当前译码的图像内容更新该显示图像的特定区域。显示图像的特定区域应保持不变直到当前的PTYPE或后续的PTYPE中的冻结图像释放比特置于1时为止，直到接收到影响该特定区域的部分图像冻结释放请求时为止，直到图像帧头中指定的源格式不同于先前图像帧头的源格式时为止，或直到暂停发生时为止，无论那种情况应以首先发生的为准。图像源格式中的任何变化应充当所有有效部分图像冻结请求的冻结释放角色。预定暂停5秒或5幅图像之后该请求将失效，无论那种情况应以较长的时间周期。通过在该暂停周期期满时限或者之前发布同一的部分图像冻结请求，暂停可以被阻止（例如：自发布以来，在具有时间参考指示时间间隔大于或等于5秒的第一图像的头部中重复该请求，或者发布之后在第5幅图像的头部重复该请求）。部分图像冻结请求的DSIZE应等于4。后随的4字节的PSUPP分别包含冻结图像矩形的左上角的水平

和垂直位置，以及该矩形的宽度和高度，使用每个分量8比特并以8像素为单位表示。例如，左上角中24像素宽和16像素高的视频显示区域通过4个参数（0，0，3，2）来指示。

L.6 尺寸再现部分图像冻结请求

尺寸再现部分图像冻结请求功能指示先前显示的视频图像的特定的矩形区域的内容将被尺度再现以适应较小部分的显示视频图像，然后该显示图像保持不变，无须使用当前译码的图像内容更新该显示图像的特定区域。显示图像的特定区域应保持不变直到当前的PTYPE或后续PTYPE中的冻结释放比特置于1时为止，直到接收影响该特定区域的部分图像冻结释放请求时为止，直到图像帧头中指定的源格式不同于先前图像帧头的源格式时为止，或者直到暂停发生时为止，无论那种情况应以首先发生的为准。图像源格式中的任何变化应充当所有有效尺寸再现部分图像冻结请求的冻结释放角色。预定暂停5秒或5幅图像之后，该请求将失效，无论那种情况应以较长的时间周期为准。通过在该暂停周期满时限或者之前发布受影响区域的显示图像的部分图像冻结请求，暂停可以被阻止（例如，自发布以来在具有时间参考指示时间间隔大于或等于5秒的第一图像的头部中发布部分图像冻结请求，或发布之后在第五幅图像的头部中发布部分图像冻结请求）。尺寸再现部分图像冻结请求的DSIZE应等于8。后随的8字节的DSUPP数据包含32比特用于指定显示图像的受影响地区的矩形区域，32比特用于指定相应的译码图像的受影响地区的矩形区域。译码图像中矩形区域的宽度和高度均应等于显示图像中矩形区域所指定的宽度和高度的 2^i 倍，其中i为1到8范围内的整数。这两个矩形区域的每一个矩形区域的位置和尺寸使用与在部分图像冻结请求功能中指定这些区域所使用的同样的格式指定。

L.7 部分图像冻结释放请求

部分图像冻结释放请求功能指示显示视频图像的指定的矩形区域的内容应被当前的和后读的译码图像更新。部分图像冻结释放请求功能的DSIZE应等于4。后随的4字节PSUPP数据采用与在部分图像冻结请求功能中指定这些区域所使用的同样的格式指定该显示图像的矩形区域。

L.8 全图像快拍标记

全图像快拍标记功能指示当前的图像作为视频内容的静止图像快拍签标供外部使用。全图像快拍标记功能的DSIZE应等于4。后随的4字节的PSUPP数据指定供外部使用的快拍标识编号。

L.9 部分图像快拍标记

部分图像快拍标记功能指示当前图像的特定的矩形区域作为视频内容的静止图像快拍签标供外部使用。部分图像快拍标记功能的DSIZE应等于8。后随的前4个字节的PUSPP数据指定供外部使用的快拍标识编号，并且后随的剩余4个字节的PSUPP数据采用与在部分图像冻结请求功能中指定这些区域所使用的同样的格式指定该译码图像的矩形区域。

L.10 视频时间分段起始标记

视频时间分段起始标记功能指示特定的视频数据子序列的起始作为视频内容的有用途的分段签标供外部使用，同当前图像一道起始。标记的视频数据的子序列应继续直至由接收的匹配的视频时间分段结束标记功能终止时为止，或者直至暂停发生时为止，无论那种情况应以首先发生的为准预定暂停5秒或5幅图像之后，标记的子序列将结束，无论那种情况应以较长的时间周期为准。通过在该暂停周期期满时限或者之前发布同一视频时间分段起始标记功能，暂停可以被阻止（例如，自发布以来，在具有时间参考指示时间间隔大于或等于5秒的第一图像的头部中重复视频时间分段起始标记功能，或发布之后在第五幅图像的头部中重复视频时间分段起始标记功能）。视频时间分段起始标记功能的DSIZE应等于4。后随的4字节的PSUPP数据指定供外部使用的视频时间分段标识编号。

L.11 视频时间分段结束标记

视频时间分段结束标记功能指示特定的视频数据的子序列的结束作为视频内容的有用途的分段签标供外部使用同先前的图像一道结束。视频时间分段起始标记功能的DSIZE应等于4。后随的4字节的PSUPP数据指定供外部使用的视频时间分段标识编号。

L.12 逐行精细分段起始标记

逐行精细分段起始标记功能指示特定的视频数据子序列的开始，该序列签标为当前图像后随一系列的零或者当前图像精细质量的多个图像，而不是作为连续活动场景的表示。该标记的视频数据子序列应继续直至由接收的匹配的逐行精细分段结束标记功能终止时为止，或者直到暂停发生时为止，无论那种情况应以首先发生的为准。预定暂停5秒或5幅图像之后，该标记的子序列将结束，无论那种情况应以较长的时间周期为准。通过在该暂停周期期满时限或者之前发布同一逐行精细分段起始标记功能，暂停可以被阻止（例如，自发布以来，在具有时间参考指示时间间隔大于或等于5秒的第一图像的头部中重复逐行精细起始标记功能，或发布之后在第五幅图像的头部中重复逐行精细起始标记功能）。视频时间分段起始标记功能的DSIZE应等于4。后随的4字节的PSUPP数据指定供外部使用的逐行精细分段标识编号。

L.13 逐行精细分段结束标记

逐行精细分段结束标记指示特定的视频数据的子序列的结束，该序列被标签为初始图像后随一系列的零或者初始图像精细质量的多个图像，并且同先前的图像一道结束。视频时间分段起始标记功能的DSIZE应等于4。后随的4字节的PSUPP数据指定供外部使用的逐行精细分段标识编号。

L.14 色度键控信息

色度键控信息功能（CKIF）指示使用“色度键控”技术表示译码视频图像中的“透明”和“半透明”像素。当作为显示呈现时，“透明”像素不显示，作为替代，背景图像或为先导参考图像或为外部控制的图像呈现。半透明像素通过当前图像中的像素值与背景图像中相对应的像素值混合来显示。使用一个字节指示供色度键控所使用的每个分量（Y、C_B和C_R）的键控彩色值，为表示准备作为“半透明”的像素，使用两个阈值，用T₁和T₂代表。令 α 代表像素的透明度； $\alpha = 255$ 指示像素不透明， $\alpha = 0$ 指示像素透明。对其他的 α 值而言，像素的生成值将是当前图像中的像素值和来自背景图像（由外部指定）像素值的加权组合。 α 的值可以用于构成称之为“字母地图”的图像。这样，每个分量的生成值可以为：

$$[\alpha \cdot X + (255 - \alpha) \cdot Z] / 255$$

其中X为译码的像素分量值（对Y、C_B或C_R），Z为来自背景图像的相应的像素分量值。

α 值可以如下计算。首先计算像素色与基色值的距离：

$$d = A_Y (X_Y - K_Y)^2 + A_B (X_B - K_B)^2 + (A_R - K_R)^2$$

其中X_Y、X_B、X_R为译码的像素色的Y、C_B、C_R值，K_Y、K_B、K_R为相应的基色参数，且A_Y、A_B和A_R为键控标记比特指示哪种彩色分量作为键控使用。一旦距离D计算后， α 值可以如以下准码中指定的那样计算。

对于每个像素，

如果 $(d < T_1)$ 则 $\alpha = 0$;

或如果 $(d > T_2)$ 则 $\alpha = 255$;

其他情况 $\alpha = [255 \cdot (d - T_1)] / (T_2 - T_1)$

然而，译码器中实施色度键控操作的精确算法未在此指定，由于对交互操作能力而言不需要该方法的标准规范。为了传送数据参数的预期解释，出于说明的考虑，在此提供描述的一个过程。

由于推导的 α 值是X_Y、X_B和X_R的简单函数，可以构造查找表（LUT）实现以上操作。这样的LUT具有 $2^{8 \times N}$ 个条目相应于所有的像素值，其中N为用做键控的彩色分量的数目。在LUT中每个条目将包含相应的 α 值。

同CKIF一起依照发送数据量，色度键控信息的DSIZE应在1到9的取值范围内（包括端点）。同图像一起发送不超过1个CKIF。

跟随DSIZE字节的首字节应包含当前图像的表示级 — 假设具有低表示级的流构成具有高表示级的流的背景图像。

若DSIZE大于1，表示级字节之后的下一个字节将用于发送6个标记比特定义为：

比特1: A_Y : 指示亮度Y的 K_Y 键控参数值存在的标记比特

比特2: A_B : 指示色度 C_B 的 K_B 键控参数值存在的标记比特

比特3: A_R : 指示色度 C_R 的 K_R 键控参数值存在的标记比特

比特4: A_1 : 指示透明度的 T_1 阈值参数存在的标记比特

比特5: A_2 : 指示不透明度的 T_2 阈值参数存在的标记比特

比特6: RPB: 指示使用参数图像作为背景图像的标记比特

比特7: 保留的

比特8: 保留的

DSIZE应等于1或应等于2加上 A_Y 、 A_B 及 A_R 之间置“1”的标记比特数目，加上2倍的 A_1 和 A_2 之间置“1”的比特数目。若DSIZE大于1，那么对每个置于1的标记比特 A_Y 、 A_B 及 A_R 应发送一个另外的字节指定每个色彩分量的值及两个另外的字节指定 T_1 和 T_2 中间每个标记的阈值数值。这些字节应遵从标记比特相同的次序。

若DSIZE等于1，或者若所有三个键控彩色标记比特 A_Y 、 A_B 和 A_R 均为0，那么供先前键控图像所使用的键控标记比特 A_Y 、 A_B 和 A_R 以及基色 K_Y 、 K_B 和 K_R 也将供当前图像使用。若视频序列未曾发送任何先前的值，则缺省的键控标记比特 $A_Y = 1$ ， $A_B = 1$ 和 $A_R = 1$ 以及缺省的基色 $K_Y = 50$ ， $K_B = 220$ 和 $K_R = 100$ 将作为先前的值使用。

若DSIZE等于1或者键控阈值标记比特 A_1 和 A_2 均为0，那么供先前键控图像所使用的阈值 T_1 和 T_2 也将供当前图像使用。若视频序列未曾发送任何先前的值，则缺省的阈值 $T_1 = 48$ 和 $T_2 = 75$ 将作为先前的值使用。

在像素为“半透明的图像部分中（即 $T_1 < d < T_2$ 时），在使用色度键控的分量中，译码的像素典型地包含色度基色。这可以导致某些彩色仿真。为了解决此类问题，这些像素同背景色混合之前，其像素值可以调整。这样的校正过程可以适用于作为由标记比特指示的色度键控操作中准备使用的彩色分量。该过程如下：

$$X' = K + (T_2 / d)(X - K)$$

其中X是原始译码的像素分量值，X'是校正值。

由于调整的像素值 X'_Y 、 X'_B 和 X'_R 是 X_Y 、 X_B 和 X_R 的函数，彩色校正能够通过使用LUT实现。该LUT将有 2^{8N} 个条目相应于所有的像素值，其中N作为键控所使用的彩色分量的数目。每个条目将包含相应的校正值。

若参考图像背景（RPB）标记比特置“1”，则它指示时域先前参数图像（对当前图像实施任何附件P在采样之前）将被认作为当前图像及所有后续的色度键控图像的（不透明）背景直至由另一个具有RPB标记置“1”的图像到达替代时为止。若当前的图像无任何时域先前的参数图像（例如若当前图像是INTRA或EI的图像），则RPB标记比特所涉及的图像为曾经正式作为参数图像的图像，只要当前图像是适当的INTER或EP类型的图像。若RPB标记比特置“0”，则它指示背景将保持如先前的控制（或在外部控制下或一旦收到具有RPB置“1”的先导图像时使用先前存储的参数图像）。

通过色度键控信息功能的发布要求色度键控的使用应同当前图像一道起始并应继续直至到后续的INTRA或EI类型图像出现时为止或直至到暂停周期计时期满时为止，无论哪种情况应以首先发生的为准。预定暂停5秒或者5帧图像之后，色度键控的使用将结束，无论哪种情况应以较长的时间周期为准。通过在该暂停周期计时期满时限或之前发布同一色度键控信息功能，暂停可以被阻止（例如自发布以来在具有时间参考指示时间间隔大于或等于5秒的第一图像的帧头中重复色度键控信息功能，或发布之后在第五幅图像的帧头中重复色度键控信息功能）。对于同INTRA或EI类型的每个图像一起以及每个暂停间隔内出现的完整的再同步而言，编码器应发送足够的具有色度键控信息功能的信息（它应不依赖使用存储的或缺省的基色值或阈值）。

L.15 扩展的功能类型

扩展的功能类型指示用于标示以下的PSUPP字节包含扩展的功能。扩展的功能用法保留给ITU以便对规定的大量的后向兼容的PSUPP数据功能具有最新的能力，扩展的功能类型指示的DSIZE应等于0，为了允许后向兼容未来使用的扩展的功能类型指示，译码器应把后随该扩展功能类型指示的字节中的第二个4比特集当做DSIZE值来处理，对于扩展的功能参数数据而言，该值指示准备跳过的后续的PSUPP字节的数目，它可以由另一个FTYPE指示来跟随。

附 件 M

改进的PB帧模式

M.1 引言

本附件描述本建议书任选的改进的PB帧模式。使用现存的改进的PB帧模式替代附件G中定义的PB帧模式被认为将是有益的。本模式的能力通过外部手段标示（例如，ITU-T H.245建议书）本模式的使用在图像帧头的PLUSPTYPE场中指示。

本选择的大部分同附件G中定义的PB帧选择相类似。

为避免同附件O中定义的B图像冲突，术语B图像、B宏块及B块将不在本附件中使用。作为替代，我们将使用记号 B_{PB} 表示改进的PB帧的“B部分”，当提到附件G时，B图像和B块应读做 B_{PB} 图像和 B_{PB} 块。

PB帧模式和改进的PB帧模式之间的主要区别是在改进的PB帧模式中，除双向预测模式外， B_{PB} 宏块具有可用的前向和后向预测模式，本附件中，MVDB（若存在）涉及为前向运动矢量。（注意，附件G中MVDB用于涉及双向预测的下标度前向和后向矢量的增强，而不是独特的前向运动矢量）。

同附件G的所有区别均在本附件中标识。当不指示任何事情时，它意味着使用附件G中所描述的统一规程。

M.2 B_{PB}宏块预测模式

存在三种不同方式的编码B_{PB}宏块。不同的编码模式由参数MODB标示，B_{PB}宏块编码模式为：

M.2.1 双向预测

在双向预测模式中，预测使用B_{PB}图像之前和之后的参数图像（在改进的PB帧序列的情形中，它意味着时间先前的改进的PB帧的P图像部分和当前的改进的PB帧的P图像部分）。该预测等效于附件G中规定的预测其中MV_D=0。注意在此模式中（且仅在此模式中），即使P宏块以INTRA编码，PB宏块的运动矢量数据（MVD）也必须包括。（注意到MVD—运动矢量数据和在附件G中定义的MV_D—增量矢量之间的区别。）

M.2.2 前向预测

在前向预测模式中，包含在MVDB中的矢量数据供来自先前参数图像的前向预测使用（INTRA或INTER图像，或PB或改进的PB帧的P图像部分）。它意味着在此预测模式中对B_{PB}宏块而言总是仅有一个16*16矢量存在。

简单的预测值供前向运动矢量的编码使用。本预测值的规则为若当前宏块不在图像或截面的最左边缘以及该宏块向左有前向运动矢量，则当前宏块的前向运动矢量的预测值设置为该块向左的前向运动矢量值，否则该预测值置0。预测值和理想的运动矢量之间的差采用与P图像所使用的矢量数据一样的方式进行VLC编码（MVD）。

有关D.1中定义的图像边界上的运动矢量，所描述的技术也适用于前向B_{PB}矢量，只要该特征在使用（它适用于双向预测模式）。

M.2.3 后向预测

在后向预测模式中，前向同样也适用于B_{PB}宏块的预测和P_{REC}（预测）相同（G.5中定义）。对后向预测而言，不使用任何运动矢量数据。

M.3 B宏块双向预测的矢量计算

在使用双向预测的情形中计算标度的前向和后向矢量如附件G中描述，其中MV_D=0。

M.4 MODB表

新定义的MODB（替换表11）显示在表M.1中，它指示B_{PB}块的可能的编码模式。

表 M.1/H.263—改进的PB帧模式的MODB表

| 索引 | CBPB | MVDB | 比特数目 | 代码 | 编码模式 |
|----|------|------|------|-------|------|
| 0 | | | 1 | 0 | 双向预测 |
| 1 | X | | 2 | 10 | 双向预测 |
| 2 | | X | 3 | 110 | 前向预测 |
| 3 | X | X | 4 | 1110 | 前向预测 |
| 4 | | | 5 | 11110 | 后向预测 |
| 5 | X | | 5 | 11111 | 后向预测 |

注—符号“X”表示该相关的句法单元存在。

附 件 N

参考图像选择模式

N.1 引言

本附件描述本建议书的任意的参数图像选择模式，使用称之为NEWPRED的修正的帧间预测方法操作。本H.263模式的能力通过外部手段标示（例如ITU-T H.245建议书）。译码器中容纳的附加图像存储的数量也可以通过外部手段标示以有助于编码器方的存储管理。该模式能够使用从译码器发送到编码器的反向信道消息以通告该编码器哪些图像部分已经在译码器方译码。此模式的使用在图像帧头的PLUSPTYPE场中指示。本模式有两个反向信道模式切换规定反向信道是否使用以及什么种类的消息在那个反向信道上从译码器返还，并且对反向信道消息而言具有按照信道方式规定的另一个子模式。

本模式的两个反向信道模式切换确定在反向信道上发送的消息种类，指定ACK（确认消息）或NACK（非确认消息）是否发送。两个切换一起定义四种基本的操作方法：

- 1) NEITHER，在那里无任何反向信道数据从译码器返还给编码器；
- 2) ACK，在那里译码器仅返还确认消息；
- 3) NACK，在那里译码器仅返还非确认消息；并且
- 4) ACK+NACK，在那里译码器返还确认和非确认两种消息。

作为以上概括准备发送的特定种类的消息在图像帧头中指示。

对反向信道消息而言，也存在按照信道表示的两种操作方法：

- 1) 分离的逻辑信道模式：该操作方法通过系统多路复用层中的分离的逻辑信道传递反向信道数据，并且
- 2) 视频混合模式：该操作方法在前向视频数据的视频编码数据流内。交付所接收的视频反向信道数据。

本附件指定反向信道消息和反向信道数据的句法。

N.2 视频源编码算法

本模式的源编码器概略框图在图N.1中显示。该图显示使用许多图像存储器的构造。源编码器可以选择其中的一个图像存储器来控制由于帧间编码所带来的时域误差传播。当本模式用于比图像稍小的单元时，诸如GOB或截面，独立分段译模式（见附件R），把具有非空头的GOB或者截面的边界当作图像边界来处理，能够用来避免由于运动补偿跨越GOB或截面边界所带来的误差传播。编码比特流中包括标示选择哪些图像供预测使用的信息。由编码器所使用的选择供预测所使用的图像策略超出本建议书的范围。

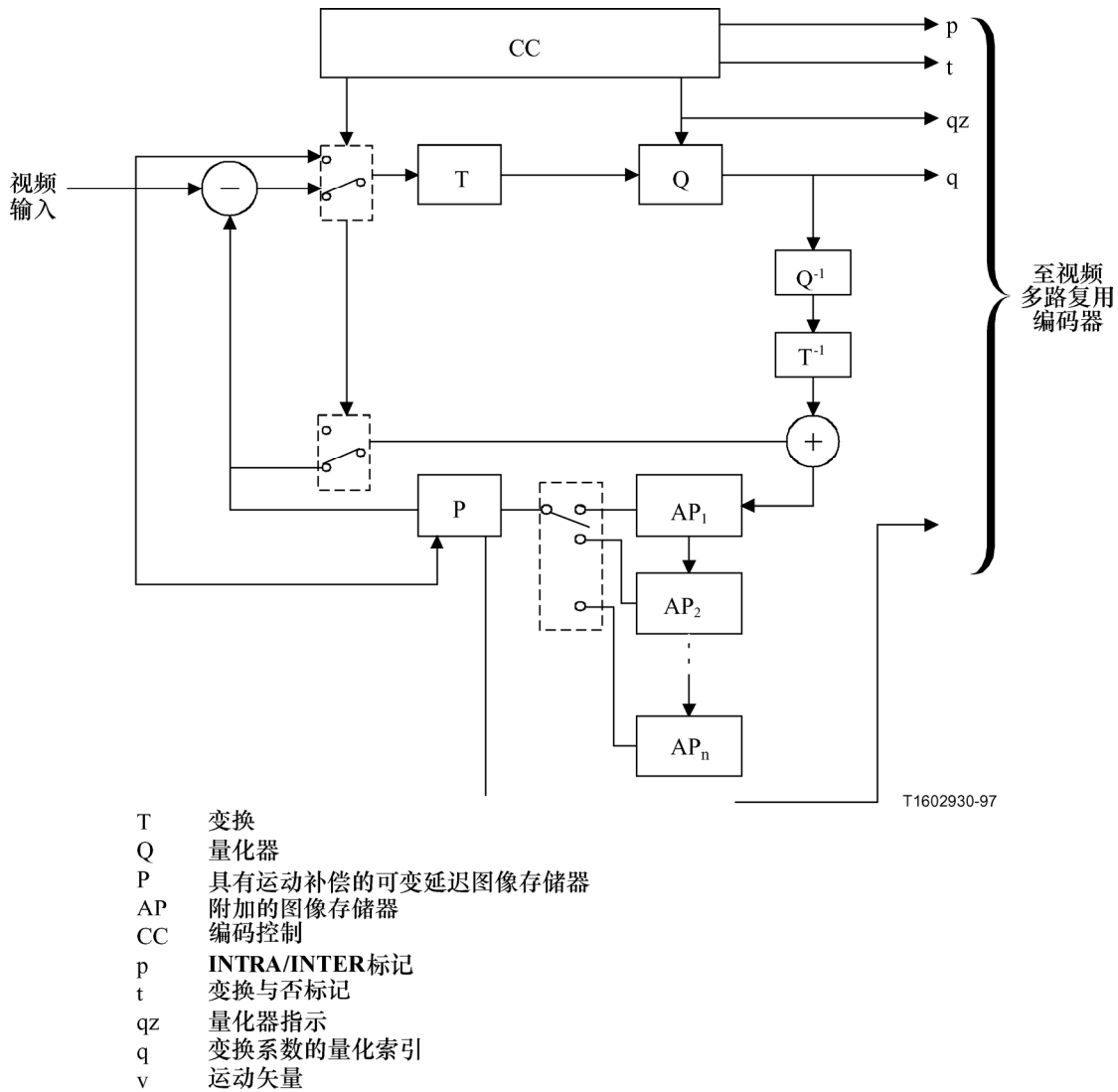


图 N.1/H.263—NEWPRED的源编码器

N.3 反向信道消息信道

按照反向信道消息的信道类型，本模式有两种操作方法。一种是可分离逻辑信道模式，另一种是视频混合模式。分离的逻辑信道模式是一个更佳的选择模式并通过专用的逻辑信道传递N.4.2中定义的反向信道消息。视频混合模式为那些由于信道的组合数目限制不能建立反向信道消息的可分离额外信道的系统所配置。频混合模式通过具有前向视频数据的同一逻辑信道以相反的方向传递反向信道消息。

N.3.1 分离的逻辑信道模式

分离的逻辑信道模式通过仅对反向信道消息开放的专用的逻辑信道传递反向信道消息。同传递视频数据的前向信道有关的机制通过外部手段提供（例如，ITU-T H.245建议书）。当这里所定义的反向信道句法不包含任何同步标记字时，为同步该反向信道内的消息，分离的逻辑信道操作要求一个外部的成帧机制。

N.3.2 视频混合模式

视频混合模式通过具有前向视频数据的同一逻辑信道以相反的方向传递反向信道消息。多路复用的比特流句法在N.4.1中描述。反向信道消息可以通过使用GOB或截面头中的反向信道消息指示（BCI）来插入。

N.4 句法

N.4.1 前向信道

传递压缩视频信号的前向信道数据的句法仅在块组（GOB）层或截面层中改变。

GOB层的句法在图N.2中说明。添加TRI、TR、TRPI、TRP、BCI及BCM场到图9。

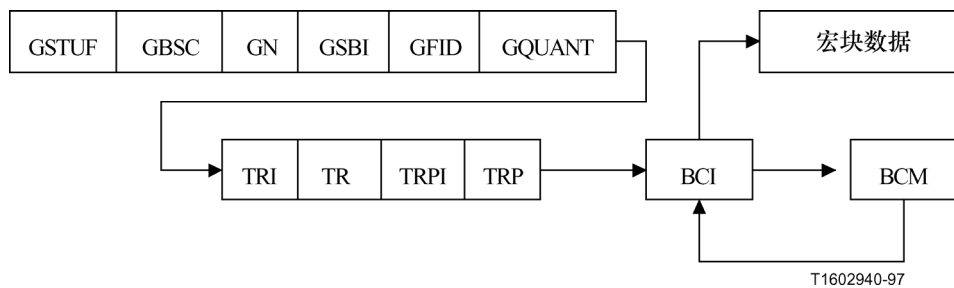


图 N.2/H.263—NEWPRED的GOB层结构

当任选的截面结构模式（见附件K）使用时，截面层的句法采用与GOB层同样的方式修改。句法说明见图N.3。

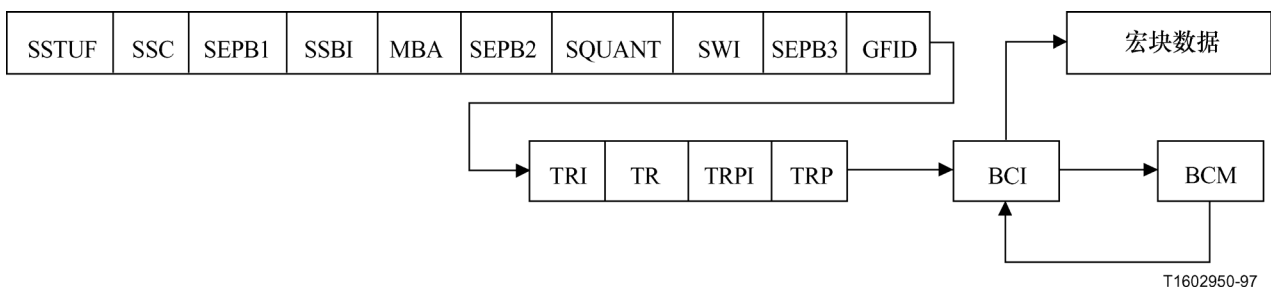


图 N.3/H.263—NEWPRED的截面层结构

N.4.1.1 时域参数指示符 (TRI) (1比特)

TRI指示以下TR场是否存在。

0: TR场不存在。

1: TR场存在。

N.4.1.2 时域参数 (TR) (8/10比特)

只要存在, TR为8比特数, 惯例图像时钟频率使用时除外; 在那种情况下, TR为10比特数, 由图像帧头的TR和连在一起的ETR组成。

N.4.1.3 预测的时域参考指示符 (TRPI) (1比特)

TRPI指示以下TRP场是否存在。

0: TRP场不存在。

1: TRP场存在。

每当图像为I或EI图像时, TRPI应等于0。

N.4.1.4 预测的时域参考 (TRP) (10比特)

若存在 (如TRPI中指示的), TRP指示供编码预测所使用的时域参考, 在B图像和改进的PB帧的B图像部分的情形中除外。对B图像或改进的PB帧的B图像部分而言, 具有时域参考TRP的图像供前向预测使用。(在逆时间方向中预测总是使用即时时间后续图像。) TRP为10比特数。对参考图像而言, 若惯例图像时钟频率未使用, 则TRP的两个MSB为0并且LSB包含在参考图像帧头中所出现的8比特TR。对参考图像而言, 若惯例图像时钟频率在使用, 则TRP为10比特数, 由来自参考图像头部的TR和连在一起的ETR组成。

TRP不存在时, 最近的时间先前的参照图像应供预测使用, 如同不在参考图像选择模式中。下一个PSC、GSC或SSC之前TRP有效。

N.4.1.5 反向信道消息指示 (BCI) (可变长度)

该场包含一或两个比特; 当置于“1”时, 它标示以下视频反向信道消息 (BCM) 场的存在。否则, 该场值为“01”, 指示视频反向信道信息场缺席或结束。BCM和BCI的组合可以不存在, 只要存在就可以重复。若视频混合模式未使用, 则BCI应总设置为“01”。

N.4.1.6 反向信道消息 (BCM) (可变长度)

反向信道消息具有N.4.2中定义的句法, 仅当前导BCI场置于“1”时存在。

N.4.2 反向信道消息 (BCM) 句法

传送确认 / 非确认消息的反向信道句法在图N.4中说明。该消息从译码器返还到编码器为了通告前向信道数据是否被正确译码。

| | | | | | | | | | | | |
|----|-----|----|--------|-------|------|------|-------|--------|-------|-----|-------|
| BT | URF | TR | ELNUMI | ELNUM | BCPM | BSBI | BEPB1 | GN/MBA | BEPB2 | RTR | BSTUF |
|----|-----|----|--------|-------|------|------|-------|--------|-------|-----|-------|

图 N.4/H.263—NEW PRED的反向信道消息 (BCM) 句法结构

N.4.2.1 反向信道消息类型（BT）（2比特）

反向信道消息类型指示相应部分的编码消息是否被正确译码。对编码器而言，要求那种类型的消息在前向信道的图像帧头中指示。

00：为未来使用保留。

01：为未来使用保留。

10：NACK。它指示相应部分的前向信道数据的错误译码。

11：ACK。它指示相应部分的前向信道数据正确的译码。

N.4.2.2 非可靠标记（URF）（1比特）

当TR或GN/MBA的可靠值对译码器不可用时，非可靠标记置“1”。（当BT为NACK时，译码器方可靠的TR可以无效）。

0：可靠。

1：非可靠。

N.4.2.3 时域参考（TR）（10比特）

时域参考包含视频图像分段的TR信息，对于该视频图像分段在反向信道消息中指示ACK/NACK。

注 — 这里所使用的名词“视频图像分段”的含义在附件R中定义，对参考图像而言，若惯例图像时钟频率未使用，则TR的两个MSB为0并且LSB包含在参考图像帧头中所出现的8比特TR。对参考图像而言，若惯例图像时钟频率在使用，则TR为10比特数由来自参考图像头中的TR和连在一起的ETR组成。

N.4.2.4 增强层编号指示（ELNUM1）（1比特）

增强层编号指示为“0”，除非任选时间，SNR和空间的可量测性模式（附件O）在前向信道数据中使用以及某些前向信道的增强层被组合在一个逻辑信道中以及反向信道消息涉及增强层（而不是基础层），在那些情形中增强层编号指示应为“1”。

N.4.2.5 增强层编号（ELNUM）（4比特）

当且仅当ELNUM1为“1”时，增强层编号才存在，在那些情形中它包含在反向信道消息中所涉及的该增强层的层编号。

N.4.2.6 BCPM（1比特）

BCPM为“0”除非在前向信道数据中使用CPM模式，在那些情形中BCP为“1”，若BCPM为“1”，它指示BSBI存在。

N.4.2.7 反向信道子比特流指示符（BSBI）（2比特）

仅当BCPM为“1”时，2比特的固定长度码字才存在。BSBI为前向信道数据中在反向信息中指示ACK/NACK消息的适当的子比特流编号的自然二进制表示，如5.2.4和附件C中所描述的。

N.4.2.8 反向信道仿真预防比特1（BEPB1）（1比特）

当且仅当视频混合模式使用时，该场存在。该场总是置“1”以防止起始码仿真。

N.4.2.9 GOB编号/宏块地址 (GN/MBA) (5/6/7/9/11/12/13/14比特)

GOB编号或宏块地址在该场中存在。若任选的截面结构模式（见附件K）未使用，则该场包含在反向信道消息中指示NACK/ACK消息视频图像分段起始的GOB编号，若任选的截面结构模式在使用，则该场包含截面起始的宏块地址，对该截面而言在反向信道消息中指NACK/ACK消息。该场的长度为本建议书中别处对GN或MBA所指示的长度。

注 — 当在视频最大模式中收到本场时，采用任选的截面结构模式指采用的方式在BCM适用的视频比特流中，而不是传送BCM数据的视频比特流中。

N.4.2.10 反向信道仿真预防比特2 (BEPB2) (1比特)

当且仅当视频混合模式使用时，该场存在。该场总是置“1”以防止起始码仿真。

N.4.2.11 请求的时域参考 (RTR) (10比特)

仅当BT为NACK时请求的时域参考存在。RTR指示同NACK有关的GOB或截面的请求时域参考。典型的，它是译码器相应位置的最后正确译码的视频图像分段的TR。对请求的参考图像而言，若惯例图像时钟频率未使用则RTR的两个MSB为零并且LSB包含该请求的参考图像的图像帧头所出现的8比特TR。对请求的参考图像而言，若惯例图像始时钟频率在使用，则RTR为10比特数由来自请求的参考图像头部的TR和连在一起的ETR组成。

N.4.2.12 填充 (BSTUF) (可变长度)

当且仅当分离的逻辑信道模式在使用且外部帧中反向信道消息处于最后，则该场存在。BSTUF由零或多个比特的“0”值的变长码字组成该场仅存在于外部帧的结尾。

N.5 译码器处理

本模式的译码器或许需要附加数目的图像存储器以存储正确译码的视频信号及其时域参考 (TR) 信息。译码器使用TR是TRP的存储的图像作为帧间译码的参考图像以代替最后的译码图像，只要在前向信道数据中TRP场存在。当TR是TRP的图像在译码器不再有效时，译码器可以通过外部手段（例如，ITU-T H.245建议书）发送强制的INTRA更新信号到编码器。除非不同的帧存储策略由外部手段协商，否则正确译码的视频图像分段应存储到存储器中，在首进首出基础上作为以后的参考图像使用，如图N.1中所示（对B图像除外，它不用作为参考图像），并且被检测为曾经非正确译码的视频图像分段将不替代在该存储区段中正确译码的视频图像分段。

确认消息 (ACK) 和非确认消息 (NACK) 作为后向信道消息定义。当译码器成功地译码视频图像分段时可以返回ACK消息。当译码器未能译码视频图像分段时可以返回NACK消息并且在译码器得到包括请求的TRP或者INTRA更新的期望的前向信道数据之前，该译码器可以继续返还消息。哪种类型的消息应该发送在前向信道数据图像帧头的RPSMF场中指示。

在使用通称为“视频冗余度编码”方案中，参考图像选择模式可以由某些编码器使用采用一种对同一时间瞬间描绘的场景发送多种表示的方式（通常使用不同的参考图像）。在这样的情形中，参考图像选择模式在使用并且比特流中毗邻的图像具有相同的时域参考，译码器应把此事当作近似同一描绘场景内容的冗余复制曾经发送的指示，并将译码和使用最先接收的图像而丢弃后续的冗余图像。

附件 O

时间的、SNR及空间的可量测性模式

本附件描述本建议书支持时间的、SNR及空间的可量测性的任选模式。本模式也可以与误差控制方案连同使用。本模式的能力以及支持其特征的程度通过外部手段标示(例如, ITU-T H.245建议书)。本模式的使用在PLUSPTYPE中指示。

O.1 概述

可量测性考虑译码的序列在多个质量等级上。该模式通过使用图像与增强图像的分级分割进入一个或多个层来实现。存在三种类型的图像供可量测性使用：B、EI及EP图像，如以下说明。这些类型中的每一类均有增强层编号ELNUM指示它隶属于哪一层，以及参考层编号RLNUM指示哪一层可供其预测使用。最低层称之为基础层，且有层编号1。

可量测性通过三种基本方法来实现：时间的、SNR及空间的增强。

O.1.1 时间的可量测性

时间的可量测性使用双向预测的图像或B图像来实现。B图像允许预测来自参考层中先前的和后续的任何两个重构图像。和P图像的压缩效率相比较该性质一般可导致改进的压缩效率。这些B图像不同于PB（或改进的PB）帧的B图像部分（见附件G和M），在那里它们是比特流中单独的实体：它们不同后续的P（或EP）图像句法的交互混合。

B图像（以及PB或改进的PB帧的B部分）不作为参考图像供任何其他图像的预测使用。若有必要此性质将考虑丢弃B图像而无需对任何后续图像产生不利的影响，如此提供时间的可量测性。图O.1说明P和B图像的预测结构。

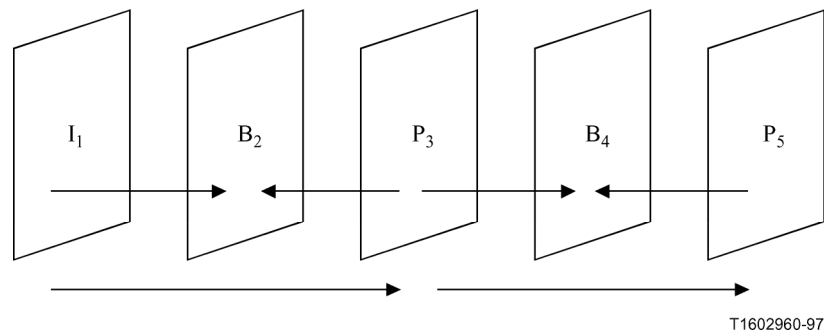


图 O.1/H.263—B图像预测相关性说明

比特流中B图像的位置是按数据相关性排序而不是按严格的时间排序。（该规则与比特流中其他图像的排序是相一致的，但对于不是B图像的所有其他图像类型而言，数据相关性排序与时间排序之间从无任何此类冲突产生。）例如，若视频序列的图像被编号为1、2、3…，那么编码图像的比特流排序将为I₁、P₃、B₂、P₅、B₄…其中下标涉及原始的图像编号（如图O.1中所说明的）。

对于可以插入到参考层中参考图像对之间的B图像数目不存在任何限制（为防止出自图像帧头中的时域参考场溢出的时间多义性有所不同是必要的）然而，此类图像的最大值可以通过外部手段标示（例如，ITU-T H.245建议书）。

B图像的图像高度、宽度以及像素宽高比应总是等于其时间后续的参考层图像的那些参数。允许运动矢量扩展超出B图像的图像边界。

O.1.2 SNR可量测性

实现可量测性的其他基本方法是通过空间/SNR增强。简而言之空间可量测性与SNR可量测性是等效的，除供内插使用外。由于压缩引入假象和失真，因此编码器中重构图像和其原始图像之间差值图像（几乎总）为非零赋值图像，包含所谓的编码误差。通常，该编码误差在编码器方丢失并不恢复。采用SNR可量测性，这些编码误差图像也能够编码并发送到译码器方，产生对译码图像的增强。该额外数据充当增加视频图像的信噪比，因此，称之为SNR可量测性。图O.2说明SNR可量测性的数据流。从较低层出发的垂直箭头说明增强层中的图像被来自参考（较低）层中的该图像重构近似所预测。

若预测组成仅来自较低层，那么增强层图像被称之为EI图像。然而，使用以前的增强层图像和时域同时的较低层参考图像产生修正的双向预测图像也是可能的。这种类型的图像称之为EP图像或者“增强”P图像。EI和EP图像的预测流显示在图O.2中。（虽然没有特别在图O.2中显示，但增强层中EI图像可以有P图像作为其较低层参考图像，并且EP图像可以有I图像作为其较低层增强图像）。

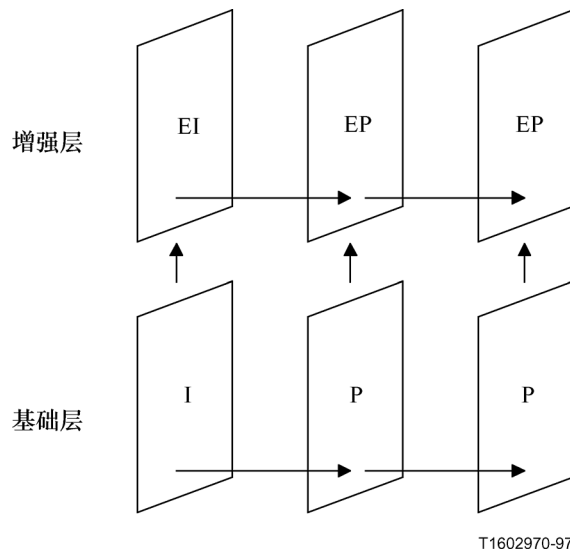


图 O.2/H.263—SNR可量测性说明

对EI和EP两种图像而言，来自参考层的预测不使用任何运动矢量。然而，作为具有正规的P图像，当预测来自同一层中其时间居前的参考图像时EP图像使用运动矢量。

O.1.3 空间的可量测性

在时间、SNR及空间的可量测性模式中第三即最后的可量测性方法是空间的可量测性，它同SNR可量测性有密切的关系。唯一的差别是参考层中的图像被用于预测空间增强层中的图像之前，它被两个系数之一或水平或垂直内插（1维空间可量测性），或者被水平和垂直两个系数内插（2维空间可量测性）。此操作的内插滤波器在O.6中定义。对于具有某种形式的空间可量测性能力的译码器而言，它也可以需要具有惯例图像格式的能力。例如，若基础层为子-QCIF（128×96），2维空间增强层图像将为256×192，它不符合标准图像格式。另一个例子将是若基础层为QCIF（176×144），具有12:11标准的像素宽高比。一维水平空间增强层将相应于352×144的图像格式并具有6:11的像素宽高比。这样，在这些情形中惯例图像格式将不得不供增强层所使用。不要求惯例图像格式的例子为QCIF基础层同CIF的2维空间增强层的使用。空间可量测性在图O.3中说明。

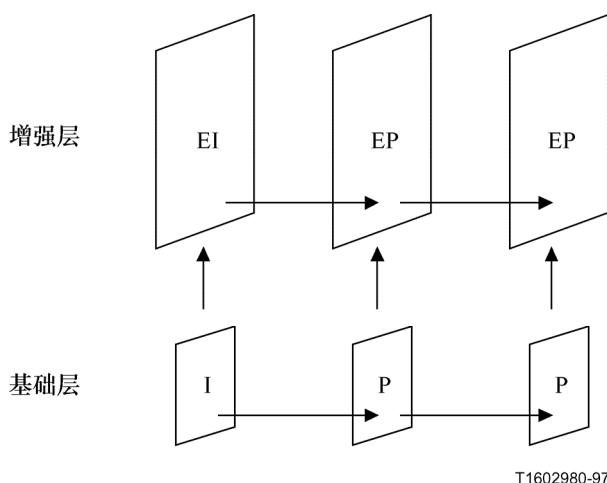


图 O.3/H.263—空间可量测性说明

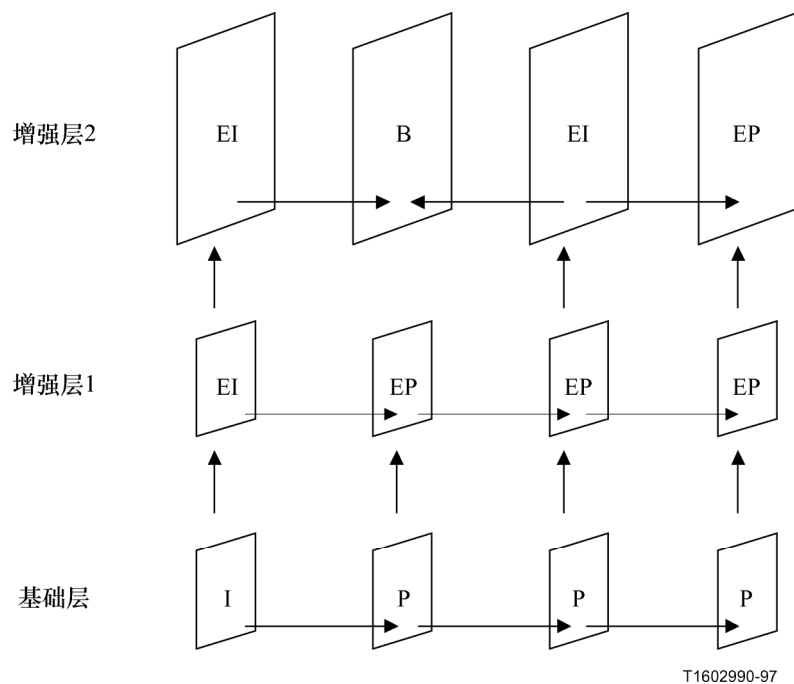
在参考图像用于作为编码处理参考之前，若要求上采样处理来增加参考层图像的尺寸，则空间可量测性图像的处理与句法基本上和SNR可量测性图像的处理与句法相同。

由于使用SNR可量测性图像与使用空间可量测性图像之间存在很小的句法上的区别，因此对于任何一种方式所使用的图像均称之为EI和EP图像。

在EI和EP图像中供向上预测所使用的基础层中的图像可以是I图像、P图像或者PB帧或改进的PB帧的P图像部分（但不应是B图像或者PB帧或改进的PB帧的B图像部分）。

O.1.4 多层可量测性

对B图像而言不仅准备时间插入到I、P、PB和改进的PB类型图像之间而且也准备插入到EI和EP类型的图像之间，不管这些图像是由SNR还是由空间增强图像组成这都是可能的。与基础层连带的SNR或空间增强层也可能不止一个。这样多层可标度比特流可以是SNR层、空间层及B图像的组合。然而，图像的尺寸不能随着层数增加而减少。它仅能够停留在相同或在一维通过两系数之一增加或在二维通过两个系数增加。图O.4说明多层可标度比特流。



T1602990-97

图 O.4/H.263—多层可量测性说明

在多层可量测性的情形中，EI或EP图像中供向上预测所使用的参考层中的图像可以是I、P、EI或EP图像，或者是基础层中PB或改进的PB帧的P图像部分（但不应是B图像或是PB帧的B图像部分）。

作为具有两层的情形，B图像可以在任何层中出现。然而，增强层中与其参考层中的B图像时间上同时发生的任何图像必须是B图像或者是PB或PB帧的B图像部分。这是为了保护B图像的可自由使用的性质。然而，注意B图像可在较低层不具有任何相应图像的层中出现。这允许编码器采用比低层更高的图像速率发送增强视频。

每个增强图像（B、EI或EP）的增强层编号和参考层编号分别在图像帧头中的ELNUM和RLNUM场中（若存在）指示。对于这些场不存在时的情况见5.1.4.4中描述的推理规则。若B图像在增强层中出现，在该层时域周围SNR或空间的可量测性图像也出现，则B图像的参考层编号（RLNUM）应与增强层编号（ELNUM）相同。

B图像的图像亮度、图像宽度及像素宽高比总是等于其时间后续的参考层图像的那些参数。

O.2 图像的传输序

取决于其他图像的图像在比特流中应定位在它们所依赖的那些图像之后。

对参考图像而言（即具有类型I、P、EI或者EP或者PB或改进的PB帧的P部分图像），适当指定的比特流句法排序应遵从以下两个规则：

- 1) 所有具有相同时域参考的参考图像在比特流中应以递增的增强层排序出现（由于为译码下一个较高层参考图像需要每个较低层参考图像）。
- 2) 如以上条款1)中讨论的所有时域同时发生的参考图像在比特流中应在任意的B图像前出现，对于这些参考图像的任意一个图像而言均为该B图像的参考层中第一时间后续参考图像（为了减少译码作为B图像参考可能需要的所有参考图像的延迟）。

然后具有较早时域参考的B图像应跟随（按每个增强层内的时域排序）。

每个B图像的比特流位置应遵守以下规则：

- 1) 其比特流位置应在参考层中其第一时间后续参考图像的比特流位置之后（由于B图像的译码一般依赖于那个参考图像的前导译码）。
- 2) 其比特流位置应在参考层中同第一时间后续参考图像一道时域同时发生的所有参考图像的比特流位置之后（为了减少译码作为B图像参考可能需要的所有参考图像的延迟）。
- 3) 其比特流位置应先于任何另外的时间后续的在其参考层中的非B图像的比特流位置（因为否则将增加参考层图像的图像存储缓冲器需求）。
- 4) 其比特流位置应在同所有第一时间后续参考图像一道时域同时发生的EI和EP图像的比特流位置之后。
- 5) 其比特流位置应先于其同一增强层内的所有时间后续图像的比特流位置（因为否则将引入不必要的延迟并增加增强层图像存储缓冲器需求）。

对在此所显示的分层结构，图O.5说明由以上规则给出的两种可允许的图像传输序（采用打点框中的数字指示比特流序，对两种可供选择的传输序用逗号隔开）。

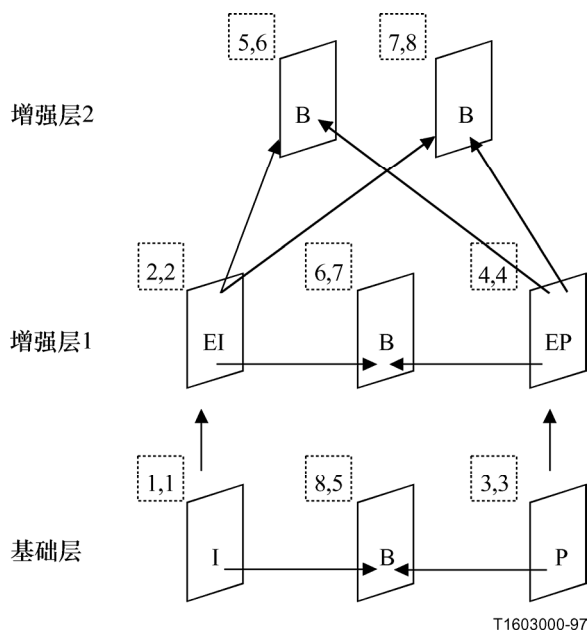


图 O.5/H.263—图像传输序实例

O.3 图像层句法

在任何增强（B、EI或EP）图像中增强层编号（ELNUM）（见5.1.11）总存在且在I或P图像或PB帧或改进的PB帧中增强层编号应不存在。对某些增强图像，参考层编号（RLNUM）（见5.1.12）存在而对其他的增强图像应推断，如5.1.12中所述。

存在明确的一个基础层，它有ELNUM和RLNUM等于1；RLNUM给出B图像的前向和后向参考图像的增强层编号及EI和EP图像的向上参考图像的增强层编号。基础层的参考图像可以由I、PB、改进的PB和P图像组成，其中没有一个在其图像头部中包括ELNUM或RLNUM（它们的隐含值为1）。

对B图像而言，RLNUM必须是小于或等于ELNUM，而对EI和EP图像而言，RLNUM必须比ELNUM小。

ELNUM可以不同于系统级所使用的层编号。由于B图像没有任何其他图像依赖于它们自身，因此它们甚至可以由本建议书外部系统单元放置在单独的增强层中（例如，ITU-T H.245和H.223建议书）。更进一步，至于增强图像是否在单独的视频信道中发送还是保持同基础层图像一起多路复用的发送，这是由操作决定的。

如5.1.4.5中所阐述的，分块滤波器模式（见附件J）不在B图像内使用。这是因为B图像不供任何其他图像的预测使用，并且因此分块滤波器对这些图像的应用完全是后处理技术，它超出本建议书的范围。然而鼓励某些类型的分块滤波器供B图像使用，并且事实上附件J中描述的滤波器作为此类使用或许是一个很好的设计。

对于I和P图像时域参考（TR）（见5.1.2）被精确定义。

B、EI或EP图像的图像帧头中将不存在 TR_B （见5.1.22）或DBQUANT（见5.1.23）场。

O.4 宏块层句法

B和EP图像的宏块层句法是相同的，由于每种图像均采用类似方式使用两个参考图像。然而，依赖于图像类型理解稍微变化。图O.6说明B和EP句法。MBTYPE场指示是否存在直接模式预测、前向预测、后向/向上预测或者双向预测。对B和EP图像MBTYPE的定义不同，如下所述。

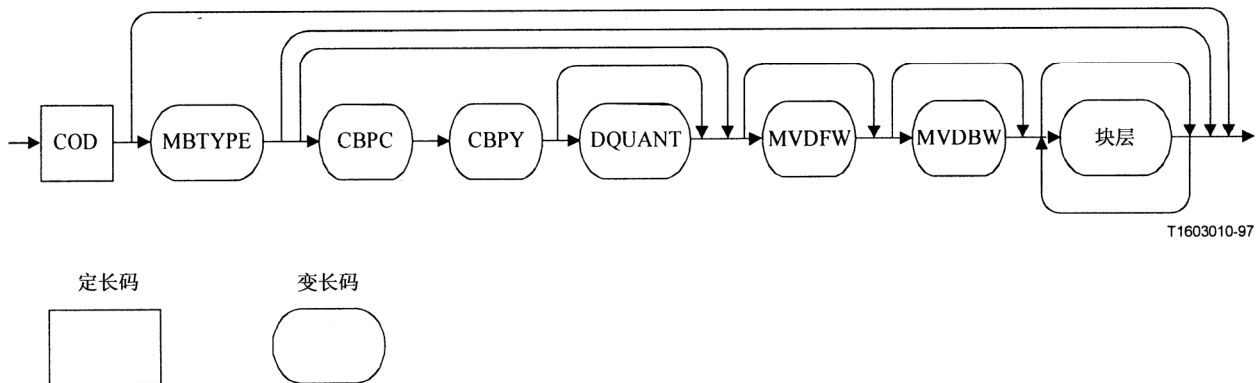


图 O.6/H.263—EP和B图像的宏块句法

直接预测模式仅对B图像有效。它为双向预测模式类似于改进的PB帧中的双向模式（附件M）。唯一区别是由于译码器方知道完整的后向预测图像因此在哪些像素可以后向预测上不存在任何限制。双向模式使用各自的前向和后向预测运动矢量。在直接和双向两种模式中，预测像素值通过平均前向和后向预测像素来计算。平均由两个预测之和除以2（舍位除法）实现。直接模式中，当参考宏块中存在4个运动矢量时，则当作改进的PB帧模式（附件M）使用所有4个运动矢量。

对B图像而言，前向预测意味着预测来自参考层中先前的参考图像。后向预测意味着预测来自参考层中时间后续参考图像。

对EP图像而言，前向预测意味着预测来自同一层中先前的EI或EP图像，而同时使用向上预测意味着预测来自时间同时发生的参考层中的（也许内插的）参考图像。虽然对前向预测可使用运动矢量，但对于向上预测（句法上处在同B图像后向预测一样的地位）不使用任何运动矢量。

EI图像的宏块句法稍有不同。如图O.7中所示，MBTYPE和CBPC组合成MCBPC场。不使用任何前向预测，仅使用来自参考层中时间上同时发生的参考图像的向上预测。不使用任何运动矢量。

在B或EP图像中，图像边界上的运动矢量可以使用如图D.1中所述（虽然如D.2中所述运动矢量范围的外围仅当非受限运动矢量模式也在使用时才激活）。

编码器应保证所有宏块与D.1.1一致，包括采用直接预测方式的所有预测；即直接预测方式不由编码器选择，除非直接方式预测过程推断的运动矢量值不会产生 16×16 （或 8×8 ）预测区的元素，该区在编码图像区外水平或垂直距离多于15像素。

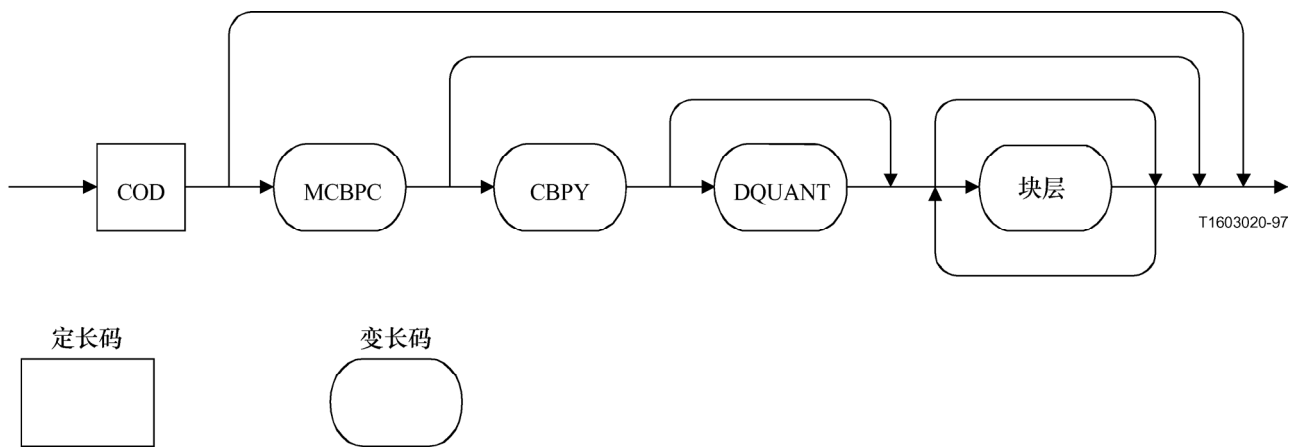


图 O.7/H.263—EI图像的宏块句法

0.4.1 编码宏块指示 (COD) (1比特)

1比特，置“0”时，标示该宏块编码。若置“1”，无此宏块的任何进一步信息被传输并且该宏块作为“跳过”处理如下述。

0.4.2 MBTYPE/MCBPC (VLC)

对B图像和EP图像存在不同的MBTYPE表。对EI图像而言，它被替代为MCBPC表。表O.1是B图像的MBTYPE表。表O.2是EP图像的MBTYPE表。表O.3是EI图像的MCBPC表。

对B图像而言，“直接（跳过）”预测类型指示在宏块中既无MBTYPE，也无任何数据被传输并且译码器导出前向和后向运动矢量及相应的双向预测。它通过COD比特标示。B图像的“前向（无纹理）”、“后向（无纹理）”及“双向（无纹理）”预测类型指示不具有任何系数以及具有一个前向和后向预测传输运动矢量，以及具有两个双向预测传输运动矢量的前向、后向及双向预测。

对EP图像而言，“前向（跳过）”预测类型指示无任何附加的宏块数据被发送，以至译码器将使用具有零运动矢量和无任何系数的前向预测。EP图像的“向上（无纹理）”和“双向（无纹理）”预测类型指示无任何系数和具有零运动矢量的向上和双向预测。

对EI图像而言“向上（跳过）”预测类型指示无任何附加的宏块数据被发送，以至译码器将使用具有零运动矢量和无任何系数的向上预测。

表 O.1/H.263—B图像的MBTYPE VLC码

| 索引 | 预测类型 | MVDFW | MVDBW | CBPC+CBPY | DQUANT | MBTYPE | 比特数 |
|----|---------|-------|-------|-----------|--------|-----------|-----|
| — | 直接(跳过的) | | | | | (COD=1) | 0 |
| 0 | 直接 | | | X | | 11 | 2 |
| 1 | 直接+Q | | | X | X | 0001 | 4 |
| 2 | 前向(无纹理) | X | | | | 100 | 3 |
| 3 | 前向 | X | | X | | 101 | 3 |
| 4 | 前向+Q | X | | X | X | 00110 | 5 |
| 5 | 后向(无纹理) | | X | | | 010 | 3 |
| 6 | 后向 | | X | X | | 011 | 3 |
| 7 | 后向+Q | | X | X | X | 00111 | 5 |
| 8 | 双向(无纹理) | X | X | | | 00100 | 5 |
| 9 | 双向 | X | X | X | | 00101 | 5 |
| 10 | 双向+Q | X | X | X | X | 00001 | 5 |
| 11 | INTRA | | | X | | 000001 | 6 |
| 12 | INTRA+Q | | | X | X | 0000001 | 7 |
| 13 | 填充 | | | | | 000000001 | 9 |

表 O.2/H.263—EP图像的MBTYPE VLC码

| 索引 | 预测类型 | MVDFW | MVDBW | CBPC+CBPY | DQUANT | MBTYPE | 比特数 |
|----|---------|-------|-------|-----------|--------|-----------|-----|
| — | 前向(跳过的) | X | | | | (COD=1) | 0 |
| 0 | 前向 | X | | X | | 1 | 1 |
| 1 | 前向+Q | X | | X | X | 001 | 3 |
| 2 | 向上(无纹理) | | | | | 010 | 3 |
| 3 | 向上 | | | X | | 011 | 3 |
| 4 | 向上+Q | | | X | X | 00001 | 5 |
| 5 | 双向(无纹理) | | | | | 00010 | 5 |
| 6 | 双向 | X | | X | | 00011 | 5 |
| 7 | 双向+Q | X | | X | X | 000001 | 6 |
| 8 | INTRA | | | X | | 0000001 | 7 |
| 9 | INTRA+Q | | | X | X | 00000001 | 8 |
| 10 | 填充 | | | | | 000000001 | 9 |

表 O.3/H.263—EI图像的MCBPC VLC码

| 索引 | 预测类型 | 编码块模型 (5, 6) | CBPY | DQUANT | MCBPC | 比特数 |
|----|----------|--------------|------|--------|-----------|-----|
| — | 向上 (跳过的) | | | | (COD=1) | 0 |
| 0 | 向上 | 00 | X | | 1 | 1 |
| 1 | 向上 | 01 | X | | 001 | 3 |
| 2 | 向上 | 10 | X | | 010 | 3 |
| 3 | 向上 | 11 | X | | 011 | 3 |
| 4 | 向上+Q | 00 | X | X | 0001 | 4 |
| 5 | 向上+Q | 01 | X | X | 000001 | 7 |
| 6 | 向上+Q | 10 | X | X | 0000010 | 7 |
| 7 | 向上+Q | 11 | X | X | 0000011 | 7 |
| 8 | INTRA | 00 | X | | 00000001 | 8 |
| 9 | INTRA | 01 | X | | 00001001 | 8 |
| 10 | INTRA | 10 | X | | 00001010 | 8 |
| 11 | INTRA | 11 | X | | 00001011 | 8 |
| 12 | INTRA+Q | 00 | X | X | 00001100 | 8 |
| 13 | INTRA+Q | 01 | X | X | 00001101 | 8 |
| 14 | INTRA+Q | 10 | X | X | 00001110 | 8 |
| 15 | INTRA+Q | 11 | X | X | 00001111 | 8 |
| 16 | 填充 | | | | 000000001 | 9 |

O.4.3 色度的编码块模型 (CBPC) (可变长度)

若存在, CBPC指示色度块的编码块模型如表O.4中所述。CBPC仅存在于EP和B图像中, 若其存在由MBTYPE指明 (见表O.1和O.2)。

表 O.4/H.263—CBPC VLC码

| 索引 | 编码块模型 (5,6) | CBPC | 比特 |
|----|-------------|------|----|
| 0 | 00 | 0 | 1 |
| 1 | 01 | 10 | 2 |
| 2 | 10 | 111 | 3 |
| 3 | 11 | 110 | 3 |

O.4.4 亮度的编码块模型 (CBPY) (可变长度)

若存在, CBPY指示在宏块的亮度部分中哪些块存在。仅当其存在由MBTYPE指明时CBPY才存在 (见表O.1、O.2和O.3)。CBPY编码如5.3.5及表12中所述。在EI和EP图像中向上预测宏块、在EP图像中双向预测宏块以及在EI、EP和B图像中INTRA宏块均使用INTRA宏块的CBPY定义, 并且EI、EP和B图像中的其他宏块类型使用INTER宏块的CBPY定义。

O.4.5 量化器信息 (DQUANT) (2比特/可变长度)

当在其他图像宏块类型时使用DQUANT。见5.3.6和附件T。

0.4.6 运动矢量数据 (MVDFW, MVDBW) (可变长度)

只要存在MVDFW是前向矢量的运动矢量数据。只要存在（仅在B图像中允许）MVDBW是后向矢量的运动矢量数据。可变长度码字在表14中给出，或表D.3中给出，若非受限运动矢量模式在使用（见附件D）。

0.5 运动矢量译码

0.5.1 差分运动矢量

前向、后向或双向预测块的运动矢量被差分编码。为恢复该宏块运动矢量，预测添加到运动矢量差上。预测以类似于6.1.1中描述的方式构成，除预测前向运动矢量仅来自宏块周围的前向运动矢量，预测后向运动矢量仅来自宏块周围的后向运动矢量的情况外。同样的判决规则适用于图像、GOB或截面边界的特定情况如6.1.1中所述。若毗邻宏块没有相同类型的（前向或后向）运动矢量，则该宏块的候选预测值对该运动矢量类型而言为零。

0.5.2 直接模式运动矢量

对采用直接模式编码的宏块而言，不传输任何矢量差。替代的，前向和后向运动矢量直接从时间连续的P矢量计算如G.4中所描述的，具有限制为 MV_D 总为0。对于其他运动矢量的预测而言，这些推导的矢量不使用。如果暂时后续参考图像的相应区采用INTRA 模式编码，指配给该区用于直接模式预测过程的前向和后向的运动矢量应有0值。

0.6 内插滤波器

对二维空间可量测性而言，内插图像所使用的方法在图O.8和图O.9中显示。第一个图显示内部像素的内插，而第二个图显示在图像边界近旁的内插。它是附件Q中以及某些情况下在附件P中所使用的相同技术。对一维空间可量测性而言，内插图像所使用的方法在图O.10和图O.11中显示。图O.10显示在水平方向上内部像素的内插。垂直方向的内插类似。图O.11显示图像边界上像素的内插。同样，垂直方向的内插类似。同样，它是某些情况下附件P中所使用的相同技术。

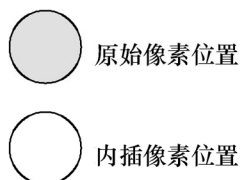
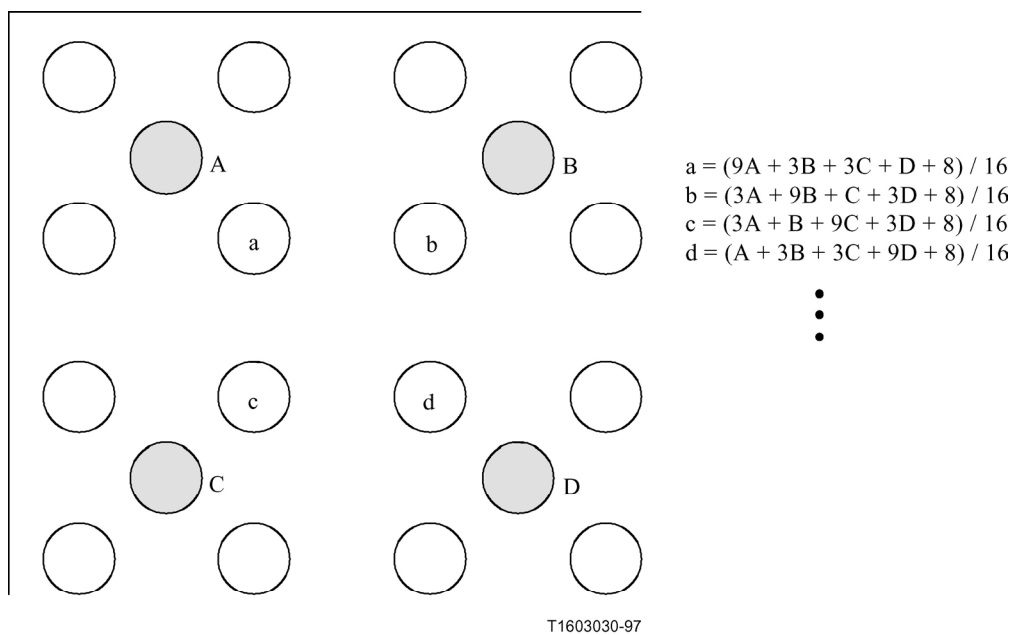


图 O.8/H.263—二维内插可量测性像素方法

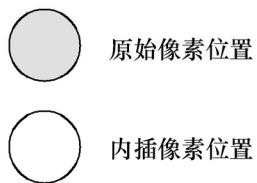
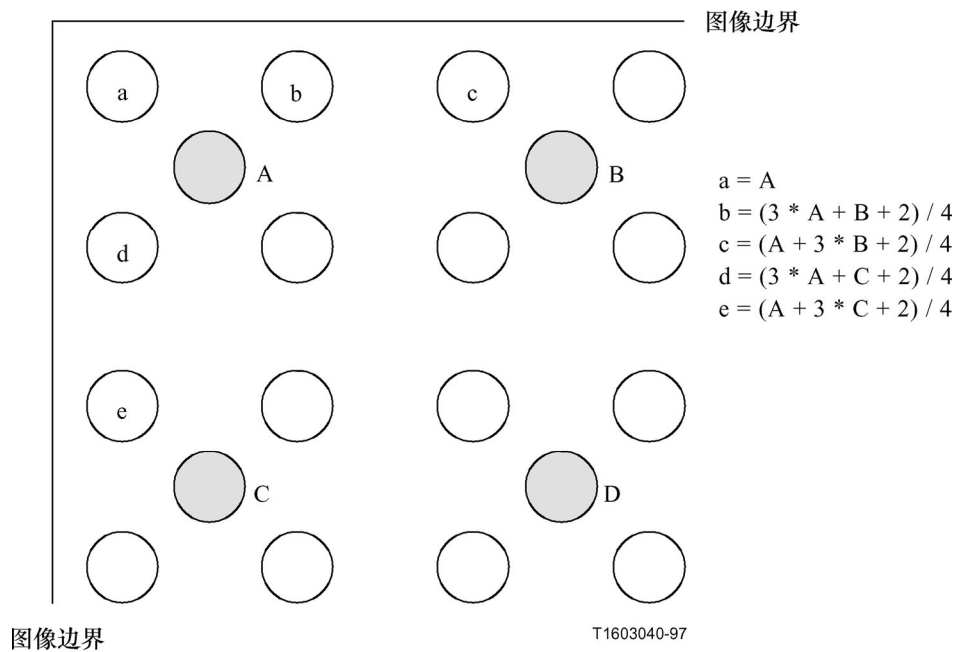
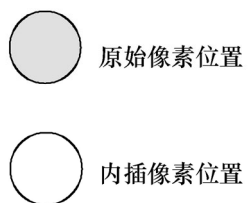
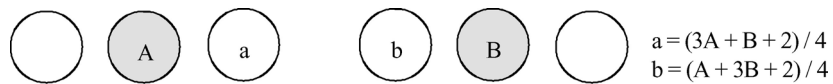


图 O.9/H.263—边界上的二维内插方法



T1603050-97

图 O.10/H.263—一维内插可量测性像素方法

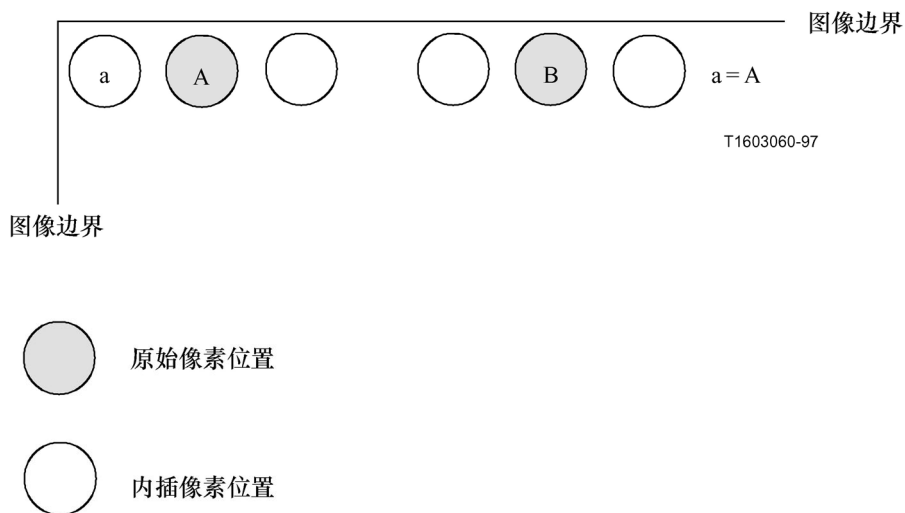


图 O.11/H.263—图像边界上的一维内插方法

附 件 P

参考图像再采样

P.1 引言

应预测当前图像的要求，为了产生“卷曲”的图像，本附件描述能够适用于先前译码参考图像的再采样处理的使用和句法。该再采样句法能够指定当前图像和具有不同源格式的先导图像间的关系，并且也能够指定相对于当前图像的先导图像的形状、大小及位置的“全局运动”卷曲变化。特别地，参考图像再采样模式能够用于适应编码期间图像的分辨率变化。使用快速算法生成双线性内插系数。使用该模式的能力以及支持其特性的扩展通过外部协商（例如，ITU-T H.245建议书）。能力协商期间，该模式可以在规定的受限环境内使用（例如，仅支持4因子图像尺寸再现、仅支持半像素分辨率图像卷曲，或支持任意图像尺寸的尺寸再现和置换）。

注一 规定不同分辨率图像之间的缺省图像变换以保持图像区域边缘的以及亮度和色度采样点相关位置的空间定位。对于供编码器生成不同分辨率图像以及译码之后供显示不同分辨率图像所使用的任何再采样操作设计而言，该规定或许是有意义的（特别考虑伴随再采样操作引入的相移所产生的空间位置变化）。此外，为了允许在编码图像尺寸之间切换而无须尺寸再现该图像显示，由于该模式能够用于动态的自适应图像分辨率的变化，因此具有该模式的操作可以受益于外部协商采用比其编码图像尺寸更高的分辨率显示译码图像。

若在PLUSPTYPE场中未设置参考图像再采样比特，但PLUSPTYPE存在并且图像为 INTER-、B-或者 EP图像或者改进的PB帧，以及该图像尺寸不同于时域先前的编码图像尺寸，那么该情况要求参考图像再采样具有卷曲参数（见P.2.2）置于0，填充模式（见P.2.3）置于clip，以及置换精度（见P.2.1）置于1/16像素精度。这使得再采样处理很容易充当预测图像分辨率上的编码变化。在简单的4因子分辨率变化情形中诸如CIF和4CIF之间的变换，再采样处理简化为与供空间可量测性（附件O）或简约分辨率更新（附件Q）所使用的滤波器同样简单的滤波器，对舍入控制的使用除外。

若图像为EP图像并且参考图像再采样比特在参考层图像帧头的PLUSPTYPE场中设置，那么该比特也应设置在增强层EP图像的图像帧头中。

若B图像使用参考图像再采样模式，则再采样处理应适用于时间先前的参照图像以及不适用于时间后续的参照图像。再采样处理所适用的时间先前参照图像应为译码的图像（即，只要对后续的参考图像也请求此模式，则该模式比由参考图像再采样所使用的任何再采样更重要）。时域后续的参照图像应有与B图像相同的图像尺寸。

若对改进的PB帧请求参考图像再采样模式，则发送一个卷曲参数集并且再采样的参考图像当做改进的PB帧的B图像和P图像两部分的参考使用。

当采用参考图像选择模式（见附件 N）时，不再请求参考图像再采样模式，除非所有图像中的TRPI 和 TRP值、当前图像的GOB和截面头规定采用相同的参考图像 — 在该情况下所指示的参考图像隐式确定是否将请求再采样过程及再采样过程将适用的图像。

参考图像再采样以当前图像区域的四个角的置换形式来定义。对于具有水平尺寸 H 和垂直尺寸 V 的当前图像的亮度场而言，分别对图像的左上、右上、左下和右下角定义4个概念性运动矢量 \underline{v}^{00} , \underline{v}^{H0} , \underline{v}^{0V} , \underline{v}^{HV} 。这些矢量描述如何移动当前图像的各个角映射它们成为相应于先前译码图像的各个角，如图P.1中所示。这些矢量的单位与参考图像坐标方格中的那些单位相同。为生成当前图像内部的某些实值位置 (x, y) 上的矢量 $\underline{v}(x, y)$ ，使用近似的双线性内插，如下：

$$\underline{v}(x, y) = \left(1 - \frac{y}{V}\right) \left[1 - \left(\frac{x}{H}\right) \underline{v}^{00} + \left(\frac{x}{H}\right) \underline{v}^{H0} \right] + \left(\frac{y}{V}\right) \left[\left(1 - \frac{x}{H}\right) \underline{v}^{0V} + \left(\frac{x}{H}\right) \underline{v}^{HV} \right]$$

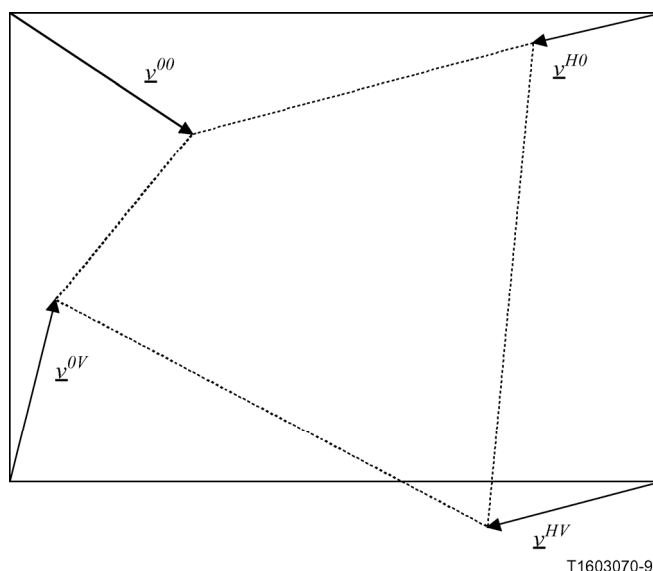


图 P.1/H.263—供卷曲使用的概念性运动矢量实例

当前图像的水平尺寸 H 及垂直尺寸 V 和参考图像的水平尺寸 H_R 及垂直尺寸 V_R 由该图像帧头所指示，不考虑这些值是否可被16除尽。若图像宽度或高度不可被16除尽，则使用再采样处理中所使用的同样的填充模式通过添加像素到再采样的图像上生成附加的区域。

作为简单的描述，再采样矢量 \underline{r}^0 , \underline{r}^x , \underline{r}^y 和 \underline{r}^{xy} 定义为：

$$\begin{aligned}\underline{r}^0 &= \underline{v}^{00} \\ \underline{r}^x &= \underline{v}^{H0} - \underline{v}^{00} \\ \underline{r}^y &= \underline{v}^{0V} - \underline{v}^{00} \\ \underline{r}^{xy} &= \underline{v}^{00} - \underline{v}^{H0} - \underline{v}^{0V} + \underline{v}^{HV}\end{aligned}$$

使用该定义，双线性内插公式改写为：

$$\underline{v}(x, y) = \underline{r}^0 + \left(\frac{x}{H}\right)\underline{r}^x + \left(\frac{y}{V}\right)\underline{r}^y + \left(\frac{x}{H}\right)\left(\frac{y}{V}\right)\underline{r}^{xy}$$

对卷曲而言，假设图像区域的左上角的坐标是 $(x, y) = (0, 0)$ 并且每个像素具有单位高度和宽度，以致该像素的中心位于点 $(x, y) = (i_L + \frac{1}{2}, j_L + \frac{1}{2})$ ，对于 $i_L = 0, \dots, H - 1$, $j_L = 0, \dots, V - 1$ ，其中 L 下标指示 i_L 和 j_L 从属于亮度场（作为像素宽高比通常为常量或者通过该再采样实施宽高比的变换，对于这些应用，不存在任何需要实际的像素宽高比）。使用该变换，在参考图像亮度场中最感兴趣的位置上 x 和 y 置换为：

$$v_x(i_L, j_L) = \frac{1}{HV} \left[HVr_x^0 + \left(i_L + \frac{1}{2}\right)Vr_x^x + \left(j_L + \frac{1}{2}\right)Hr_x^y + \left(i_L + \frac{1}{2}\right)\left(j_L + \frac{1}{2}\right)r_x^{xy} \right]$$

$$v_y(i_L, j_L) = \frac{1}{HV} \left[HVr_y^0 + \left(i_L + \frac{1}{2}\right) Vr_y^x + \left(j_L + \frac{1}{2}\right) Hr_y^y + \left(i_L + \frac{1}{2}\right) \left(j_L + \frac{1}{2}\right) r_y^{xy} \right]$$

由于必须要计算的所有位置和相位均同左上角像素的中心有关，该中心有坐标 $(x, y) = \left(\frac{1}{2}, \frac{1}{2}\right)$ ，主要感兴趣的量为：

$$\begin{aligned} x_R(i_L, j_L) - \frac{1}{2} &= \left(i_L + \frac{1}{2}\right) + v_x(i_L, j_L) - \frac{1}{2} \\ &= \frac{1}{HV} \left[HVr_x^0 + \left(i_L + \frac{1}{2}\right) (HV + Vr_x^x) + \left(j_L + \frac{1}{2}\right) Hr_x^y + \left(i_L + \frac{1}{2}\right) \left(j_L + \frac{1}{2}\right) r_x^{xy} \right] - \frac{1}{2} \\ y_R(i_L, j_L) - \frac{1}{2} &= \left(i_L + \frac{1}{2}\right) + v_y(i_L, j_L) - \frac{1}{2} \\ &= \frac{1}{HV} \left[HVr_y^0 + \left(j_L + \frac{1}{2}\right) Vr_y^x + \left(j_L + \frac{1}{2}\right) (HV + Hr_y^y) + \left(i_L + \frac{1}{2}\right) \left(j_L + \frac{1}{2}\right) r_y^{xy} \right] - \frac{1}{2} \end{aligned}$$

在前导译码参考图像中使用这些公式的近似值，一旦位置已经确定，应使用本附件中稍后所指定的双线性内插生成再采样像素值。

每个再采样矢量均能够分解成两个分量，第一个分量描述几何卷曲及第二个分量计量预测图像（水平尺寸 H 和垂直尺寸 V ）与参考图像（水平尺寸 H_R 及垂直尺寸 V_R ）之间的任意尺寸差。该分解如下：

$$\begin{aligned} \underline{v}^{00} &= \underline{v}_{-warp}^{00} + \underline{v}_{-size}^{00} = \underline{v}_{-warp}^{00} + (0, 0) \\ \underline{v}^{H0} &= \underline{v}_{-warp}^{H0} + \underline{v}_{-size}^{H0} = \underline{v}_{-warp}^{H0} + (H_R - H, 0) \\ \underline{v}^{0V} &= \underline{v}_{-warp}^{0V} + \underline{v}_{-size}^{0V} = \underline{v}_{-warp}^{0V} + (0, V_R - V) \\ \underline{v}^{HV} &= \underline{v}_{-warp}^{HV} + \underline{v}_{-size}^{HV} = \underline{v}_{-warp}^{HV} + (H_R - H, V_R - V) \end{aligned}$$

P.2 句法

每当参考图像再采样比特在该图像帧头的PLUSPTYPE场中设置时，该图像头部的RPRP场包括控制参考图像再采样处理的参数。该场包括两比特卷曲置换精度（WDA）场，可以包括8个卷曲参数或1比特卷曲参数精细，并包括填充模式，如本子节中所描述的。

P.2.1 卷曲置换精度（WDA）（2比特）

2比特卷曲置换精度场WDA在比特流的RPRP场中首先出现，指示每个像素的置换精度。值“10”指示每个像素的x和y置换量化到半像素精度。值“11”指示置换量化到1/16像素精度，其他值的使用保留。

P.2.2 卷曲参数（可变长度）

对INTER或B图像，或改进的PB帧而言，当发送参考图像再采样参数时，图像头部中包括8个卷曲参数，该参数采用表D.3中所示的可变长度码（VLC）表示。对使用SNR可量测性的EP图像而言，采用较低层的卷曲参数并且不传输任何卷曲参数。若参考图像再采样比特在使用空间可量测性的EP图像的图像头部PLUSPTYPE场中设置，则较低层的卷曲参数应被精细到当前层所要求的精度，通过每个较低层的上采样维的卷曲参数（卷曲参数具有下标x和/或y）乘以2，并且添加一个另外的比特值，发送该值来实现替代有关的卷曲参数确定该卷曲参数的最低有效比特。

8个整数卷曲参数（或者它们的一比特精细）依以下顺序发送：

$$w_x^0, w_y^0, w_x^x, w_y^x, w_x^y, w_y^y, w_x^{xy}, \text{和 } w_y^{xy}$$

当无一比特精细时，这些参数以类似于PLUSPTYPE存在时的非受限运动矢量模式的运动矢量差的方式传送，在该参数范围上使用无任何限制的表D.3（即 -4095到+4095的范围）。每当编码运动矢量差对时，作为必需的一步每对卷曲参数发送后应增加仿真防止比特，这样对每对（ w_x^0 和 w_y^0 、 w_x^x 和 w_y^x 、 w_x^y 和 w_y^y 、 w_x^{xy} 和 w_y^{xy} ）中的两个卷曲参数而言，若使用表D.3的全零码字（以半像素为单位，值+1），则该对码字应通过后随等于1的单比特来防止起始码仿真。

这8个卷曲参数被理解为图像各个角的置换，它同去除再采样矢量的尺寸再现分量所引起的置换有关。按比例标度卷曲参数以表示当前图像亮度场中的半像素补偿，这些参数的取值范围为 -4095到+4095。通过以下关系，使用再采样矢量定义卷曲参数：

$$\begin{aligned} w_x^0 &= 2r_x^0 & w_y^0 &= 2r_y^0 \\ w_x^x &= 2(r_x^x - (H_R - H)) & w_y^x &= 2r_y^x \\ w_x^y &= 2r_x^y & w_y^y &= 2(r_y^y - (V_R - V)) \\ w_x^{xy} &= 2r_x^{xy} & w_y^{xy} &= 2r_y^{xy} \end{aligned}$$

P.2.3 填充模式（FILL_MODE）（2比特）

对INTER或B图像或改进的PB帧而言，在图像头部即时跟随VLC编码卷曲参数的是规定对像素值采取填充模式行动的2个比特，该像素值是参考图像中计算位置落在该参考图像区域之外的那些像素值。这些2比特的含义在表P.1中显示，它们的位置在图P.2中显示。对EP图像而言，填充模式行动与对参考层的填充模式行动相同，因此2比特的填充模式不发送。

表 P.1/H.263—填充模式比特/行动

| 填充模式比特 | 填充行动 |
|--------|--------------|
| 00 | <i>Color</i> |
| 01 | <i>Black</i> |
| 10 | <i>Gray</i> |
| 11 | <i>Clip</i> |

若填充模式为clip，前导参考图像中坐标位置作为非受限运动矢量模式单独被限制，以致前导参考图像区域外的像素通过从图像边界上的像素值的外推来估计。若填充模式为black，指定先导参考图像区域外的亮度采样值为 $Y=16$ ，色度采样值为 $C_B=C_R=128$ 。若填充模式为gray，指定亮度和色度值为 $Y=C_B=C_R=128$ 。若填充模式为color，则将发送附加场以指定填充色，如下一子节中所述。

P.2.4 填充色规格 ($Y_FILL, C_B_EPB, C_B_FILL, C_R_EPB, C_R_FILL$) (26比特)

若填充模式为color且图像不是EP图像，那么比特流中填充模式比特由3个8比特整数 Y_fill ， C_B_fill 和 C_R_fill 跟随，它们精确地指定填充色。这些三个8比特整数之间为两个仿真防止比特 (C_B_EPB 和 C_R_EPB)，每个比特均为1。该彩色规格的格式，仅当填充模式为color时才存在，它在图P.2中显示。每个8比特整数场使用其自然二进制表示发送。对EP图像而言，填充模式行动 (和填充色) 与参考图像的填充模式行动 (和填充色) 相同，因此填充色规格不发送。

| | | | | | |
|-----------|--------|---------------------|----------------------|---------------------|----------------------|
| FILL_MODE | Y_FILL | C _B _EPB | C _B _FILL | C _R _EPB | C _R _FILL |
|-----------|--------|---------------------|----------------------|---------------------|----------------------|

图 P.2/H.263—填充模式的格式和填充彩色规格数据

P.3 再采样算法

本子节中所描述的算法结果数学上应等于用于生成再采样参考图像样点的结果。使用整数卷曲参数 $w_x^0, w_y^0, w_x^x, w_y^x, w_x^y, w_y^y, w_x^{xy}$ 和 w_y^{xy} ，整数参数 $u_x^{00}, u_y^{00}, u_x^{H0}, u_y^{H0}, u_x^{0V}, u_y^{0V}, u_x^{HV}$ 及 u_y^{HV} ，代表1/32像素精度的亮度场的四角上的 x 与 y 置换 (实际置换通过这些值除以32获得) 定义为：

$$\begin{aligned}
 u_x^{00} &= 16w_x^0 & u_y^{00} &= 16w_y^0 \\
 u_x^{H0} &= 16(w_x^0 + w_x^x + 2(H_R - H)) & u_y^{H0} &= 16(w_y^0 + w_y^x) \\
 u_x^{0V} &= 16(w_x^0 + w_x^y) & u_y^{0V} &= 16(w_y^0 + w_y^y + 2(V_R - V)) \\
 u_x^{HV} &= 16(w_x^0 + w_x^x + w_x^y + w_x^{xy} + 2(H_R - H)) & u_y^{HV} &= 16(w_y^0 + w_y^x + w_y^y + w_y^{xy} + 2(V_R - V))
 \end{aligned}$$

下一步， H' 和 V' 代表虚帧的水平和垂直尺寸，定义为满足以下条件的最小整数：

$$H' \geq H, V' \geq V, H' = 2^m, V' = 2^n, \text{ 其中 } m, n \text{ 为正整数}$$

通过对亮度场的角矢量采用双线性内插，整数参数 u_x^{LT} , u_y^{LT} , u_x^{RT} , u_y^{RT} , u_x^{LB} , u_y^{LB} , u_x^{RB} 和 u_y^{RB} 代表1/32像素精度的亮度场在虚点 $(x, y) = (0,0)$, $(H',0)$, $(0,V')$ 和 (H',V') 的 x 与 y 置换（实际置换通过这些值除以32获得）定义为：

$$\begin{aligned}
 u_x^{LT} &= u_x^{00} & u_y^{LT} &= u_y^{00} \\
 u_x^{RT} &= \left((H-H')u_x^{00} + H'u_x^{H0} \right) // H & u_y^{RT} &= \left((H-H')u_y^{00} + H'u_y^{H0} \right) // H \\
 u_x^{LB} &= \left((V-V')u_x^{00} + V'u_x^{0V} \right) // V & u_y^{LB} &= \left((V-V')u_y^{00} + V'u_y^{0V} \right) // V \\
 u_x^{RB} &= \left((V-V') \left((H-H')u_x^{00} + H'u_x^{H0} \right) + V' \left((H-H')u_x^{0V} + H'u_x^{HV} \right) \right) // (HV) \\
 u_y^{RB} &= \left((V-V') \left((H-H')u_y^{00} + H'u_y^{H0} \right) + V' \left((H-H')u_y^{0V} + H'u_y^{HV} \right) \right) // (HV)
 \end{aligned}$$

其中，“//”代表整数除法，四舍五入该商为最近的整数，并四舍五入半整数值为1。

在本附件的以下部分中，假设亮度和色度两个场的像素的中心均落在点 $(x,y) = (i+1/2, j+1/2)$ 上。整数参数 i, j 定义为：

- $i = 0, \dots, H-1$ 以及 $j = 0, \dots, V-1$ 对亮度场, 以及
- $i = 0, \dots, H/2-1$ 以及 $j = 0, \dots, V/2-1$ 对色度场。

这意味着对亮度和色度使用不同的坐标系统如图P.3所示。使用色度坐标系统，以上定义的整数参数 u_x^{LT} , u_y^{LT} , u_x^{RT} , u_y^{RT} , u_x^{LB} , u_y^{LB} , u_x^{RB} 和 u_y^{RB} 也能够被认为是1/64像素精度的该色度场在虚点 $(x, y) = (0, 0)$, $(H'/2, 0)$, $(0, V'/2)$, $(H'/2, V'/2)$ 的 x 与 y 置换（实际置换通过这些值除以64获得）。使用这些参数和附加的参数 S ，对亮度场定义为2，对色度场定义为1，亮度和色度场的再采样算法就可使用共同的公式定义。

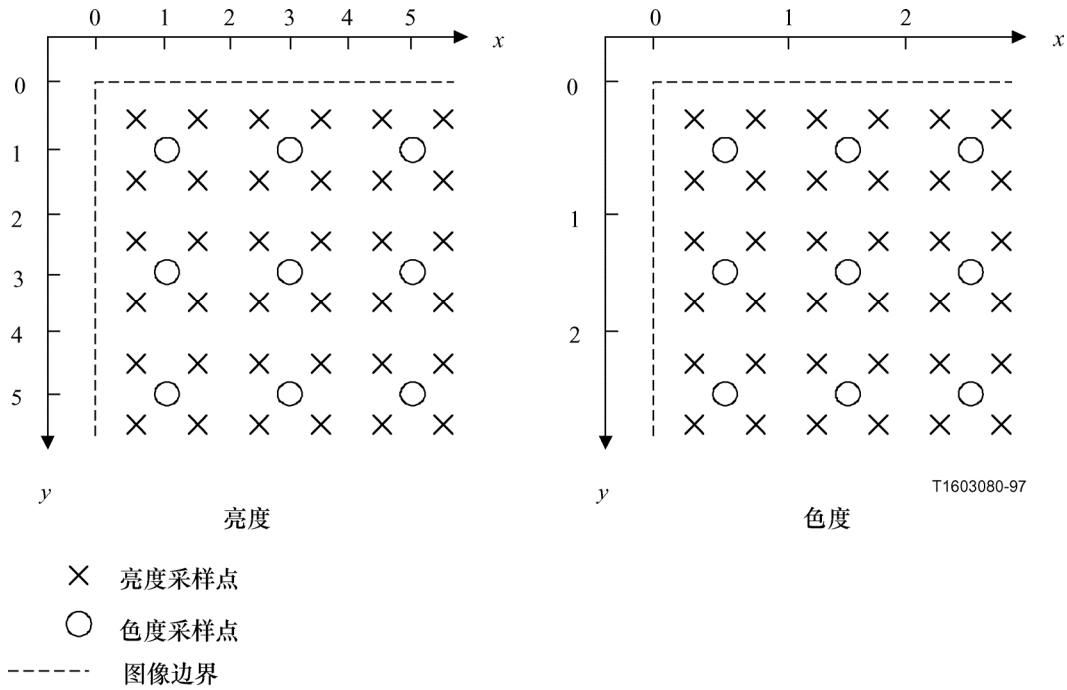


图 P.3/H.263—亮度与色度场的坐标系统

整数参数 $u_x^L(j)$, $u_y^L(j)$, $u_x^R(j)$, $u_y^R(j)$, 代表在 $(x, y) = (0, j + 1/2)$ 及 $(SH'/2, j + 1/2)$ 上 $s/64$ 像素精度的图像场的 x 与 y 置换（实际置换通过这些值除以 $64/S$ 来获得）使用以下一维线性内插定义为：

$$\begin{aligned}
 u_x^L(j) &= ((SV' - 2j - 1)u_x^{LT} + (2j + 1)u_x^{LB}) // (SV') & u_y^L(j) &= ((SV' - 2j - 1)u_y^{LT} + (2j + 1)u_y^{LB}) // (SV') \\
 u_x^R(j) &= ((SV' - 2j - 1)u_x^{RT} + (2j + 1)u_x^{RB}) // (SV') & u_y^R(j) &= ((SV' - 2j - 1)u_y^{RT} + (2j + 1)u_y^{RB}) // (SV')
 \end{aligned}$$

其中，“//”代表整数除法，四舍五入该商为最近的整数，并四舍五入半整数值为1。

最后，在参考图像中指定该变换位置的参数为：

$$\begin{aligned}
 I_R(i, j) &= Pi + ((SH' - 2i - 1)u_x^L(j) + (2i + 1)u_x^R(j) + 32H' / P) // (64H' / P) \\
 J_R(i, j) &= Pj + ((SH' - 2i - 1)u_y^L(j) + (2i + 1)u_y^R(j) + 32H' / P) // (64H' / P) \\
 i_R(i, j) &= I_R(i, j) // P & j_R(i, j) &= J_R(i, j) // P \\
 \varnothing_x &= I_R(i, j) - (I_R(i, j) // P)P & \varnothing_y &= J_R(i, j) - (J_R(i, j) // P)P
 \end{aligned}$$

其中

“//” 整数舍位除法，指向负无穷；

“/” 整数除法（在此情形导致无任何精度丢失）；

P x 和 y 置换的精度 ($P=2$ 只要 $WDA=“10”$, $P=16$ 只要 $WDA=“11”$ 或缺席, 见P.2.1 WDA的定义);

$\left(\frac{I_R(i,j)}{P} + \frac{1}{2}, \frac{J_R(i,j)}{P} + \frac{1}{2}\right)$ (x, y) 位置的变换位置($I_R(i, j)$ 和 $J_R(i, j)$ 均为整数);

$\left(i_R(i,j) + \frac{1}{2}, j_R(i,j) + \frac{1}{2}\right)$ 靠近变换位置的采样点 (x, y) 位置($i_R(i, j)$ 和 $j_R(i, j)$ 均为整数);

(Φ_x, Φ_y) 变换位置的双线性内插系数 (Φ_x 和 Φ_y 均为整数)。

该等式的计算可以通过移位运算替代除法得以简化, 其中 $64H'/P = 2^{m+2}$ 若 $P = 16$ 以及 $64H'/P = 2^{m+5}$ 若 $P = 2$ 。

使用这些参数, 再采样图像中位置在 $(x,y)=(i+1/2, j+1/2)$ 上的像素采样值 $E_p(i, j)$ 使用以下双线性内插获得:

$$E_p(i, j) = \left((P - \Phi_y) \left((P - \Phi_x) E_R(i_R, j_R) + \Phi_x E_R(i_R + 1, j_R) \right) + \Phi_y \left((P - \Phi_x) E_R(i_R, j_R + 1) + \Phi_x E_R(i_R + 1, j_R + 1) \right) + P^2 / 2 - 1 + RCRPR \right) / P^2$$

其中“/”代表舍位除法, i_R 和 j_R 是 $i_R(i, j)$ 和 $j_R(i, j)$ 的简化表示, 且 $E_R(i_R, j_R)$ 代表外推后的参考图像中位置在 $(x, y)=(i_R + 1/2, j_R + 1/2)$ 上的像素采样值, 若必要使用适当填充模式。参数RCRPR的值定义如下:

- 对存在P图像作为其时间后续参照图像的B图像(或者改进的PB帧的B部分)而言, RCRPR等于该时间后续P图像的MPPTYPE(见S.1.43)中的舍入类型(RTYPE)比特, 这意味着对改进的PB帧而言, 对P部分和B部分RCRPR有相同的值。
- 对其他类型的图像而言, RCRPR等于当前图像的RTYPE比特。

P.4 操作实例

本子节中, 作为准码提供以上子节中所描述的算法操作实例。

P.4.1 虚点置换

当大尺寸图像编码时, 为了获得P.3中显示的参数 u_x^{RB} 和 u_y^{RB} 该公式的直接操作可以迫使要求其二进制表示比32比特还要多的变量的使用。对于不能很方便的使用64比特整数或浮点寄存器的系统而言, 为使 u_x^{RB} 和 u_y^{RB} 的计算不要求采用比32比特还要多的变量, 算法实例显示如下。

由于 H, V, H', V' 被除以4, u_x^{RB} 的定义可以改写为:

$$u_x^{RB} = ((V_Q - V_Q')((H_Q - H_Q')u_x^{00} + H_Q'u_x^{H0}) + V_Q'((H_Q - H_Q')u_x^{0V} + H_Q'u_x^{HV})) // A$$

其中 $H_Q = H/4$, $V_Q = V/4$, $H_Q' = H'/4$, $V_Q' = V'/4$, $A = H_Q V_Q$ 以及“//”代表整数除法, 四舍五入该商为最近的整数, 并四舍五入半整数值为1。下一步, 为简化描述, 参数 T_T 和 T_B 定义为:

$$T_T = (H_Q - H_Q')u_x^{00} + H_Q'u_x^{H0}$$

$$T_B = (H_Q - H_Q')u_x^{0V} + H_Q'u_x^{HV}$$

使用运算符“//”代表整数舍位除法, 指向负无穷及运算符“%”定义为 $a \% b = a - (a // b)b$, 经由以下准码, 可以获得 u_x^{RB} 值:

```
q = (V_Q - V_Q') * (T_T // A) + V_Q' * (T_B // A) + ((V_Q - V_Q') * (T_T % A) + V_Q' * (T_B % A)) // A;
r = ((V_Q - V_Q') * (T_T % A) + V_Q' * (T_B % A)) % A;
if (q < 0)
    u_x^{RB} = q + (r + (A - 1) / 2) / A;
else
    u_x^{RB} = q + (r + A / 2) / A;
```

使用该算法 u_y^{RB} 值也能计算。

P.4.2 再采样算法

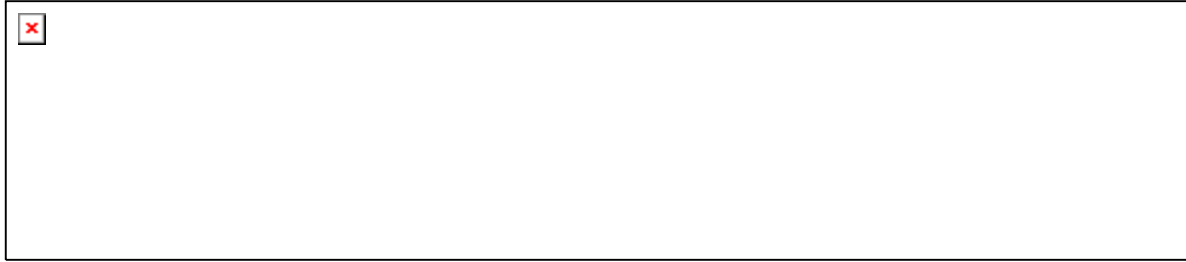
为简化算法描述, 定义函数prior_sample。其目的为产生同先导参考图像采样格栅有关的任何整数位置 (i_p, j_p) 的像素值:

```
clip(x_min, x, x_max) {
    if (x < x_min) {
        return x_min;
    } else if (x > x_max) {
        return x_max;
    } else {
        return x;
    }
}

prior_sample (i_p, j_p) {
    if (FILL_MODE = clip) {
        i_c = clip (0, i_p, S * H_R / 2 - 1);
        j_c = clip (0, j_p, S * V_R / 2 - 1);
        return prior_ref[i_c, j_c];
    } else {
        if ((i_p < 0) OR (i_p > S * H_R / 2 - 1) OR (j_p < 0) OR (j_p > S * V_R / 2 - 1) {
            return fill_value;
        } else {
            return prior_ref[i_p, j_p];
        }
    }
}
```

在准码中, prior_ref[i,j]指示时间先前的参考图像中*i*列和*j*行中的采样点。

下一步，定义滤波器功能实施P.3中描述的双线性内插。假设以下函数的所有自变量均为整数，并且双线性内插系数 ϕ_x 和 ϕ_y 被量化到0,..., P-1的范围（闭区间）。



最后，卷曲参考图像生成当前图像预测的方法可以用这些函数的形式指定。预测图像的像素可以依光栅扫描顺序生成。假设 $u_x^L(j)$, $u_y^L(j)$, $u_x^R(j)$ 和 $u_y^R(j)$ 的值已经计算并承载到变量 u_x^L , u_y^L , u_x^R 和 u_y^R 上。定义参数 D 为 $D = 64H'/P$ ，并且注意 $H' = 2^m$ ，则再采样场的第 j 行中像素的采样值（最顶行定义为第0行）通过以下准码获得：

```

 $a_x^i = D * P + 2 * (u_x^R - u_x^L);$ 
 $a_y^i = 2 * (u_y^R - u_y^L);$ 
 $a_x = u_x^L * S * 2^m + (u_x^R - u_x^L); + D / 2;$ 
 $a_y = j * D * P + u_y^L * S * 2^m + (u_y^R - u_y^L); + D / 2;$ 
for (i = 0; i < S * H / 2; i++) {
     $I_R = a_x /// D;$ 
     $J_R = a_y /// D;$ 
     $i_R = I_R /// P;$ 
     $j_R = J_R /// P;$ 
     $\phi_x = I_R - (i_R * P);$ 
     $\phi_y = J_R - (j_R * P);$ 
    new_ref[i, j] = filter( $i_R$ ,  $j_R$ ,  $\phi_x$ ,  $\phi_y$ );
     $a_x += a_x^i;$ 
     $a_y += a_y^i;$ 
}

```

其中在此码中所使用的所有变量均为整数变量并且new_ref[i,j]指示再采样参考图像中 i 列及 j 行所生成的采样点。依照该参数的定义，此码中所有的除法均可以用二进移位操作替代。例如，当 $P = 16$ ：

```

 $I_R = a_x /// D;$ 
 $J_R = a_y /// D;$ 
 $i_R = I_R /// P;$ 
 $j_R = J_R /// P;$ 
 $\phi_x = I_R - (i_R * P);$ 
 $\phi_y = J_R - (j_R * P);$ 

```

假设 a_x, a_y, I_R 和 J_R 是用2的补码表示的二进制编码整数变量，可以改写为

```

$$I_R = a_x \gg (m+2);$$

$$J_R = a_y \gg (m+2);$$

$$i_R = I_R \gg 4;$$

$$j_R = J_R \gg 4;$$

$$\emptyset_x = I_R \& 15;$$

$$\emptyset_y = J_R \& 15;$$

```

其中“ $\gg N_{\text{移位}}$ ”表示 $N_{\text{移位}}$ 比特右算术二进制移位（ $N_{\text{移位}}$ 为正整数），“ $\&$ ”代表比特方式AND运算。

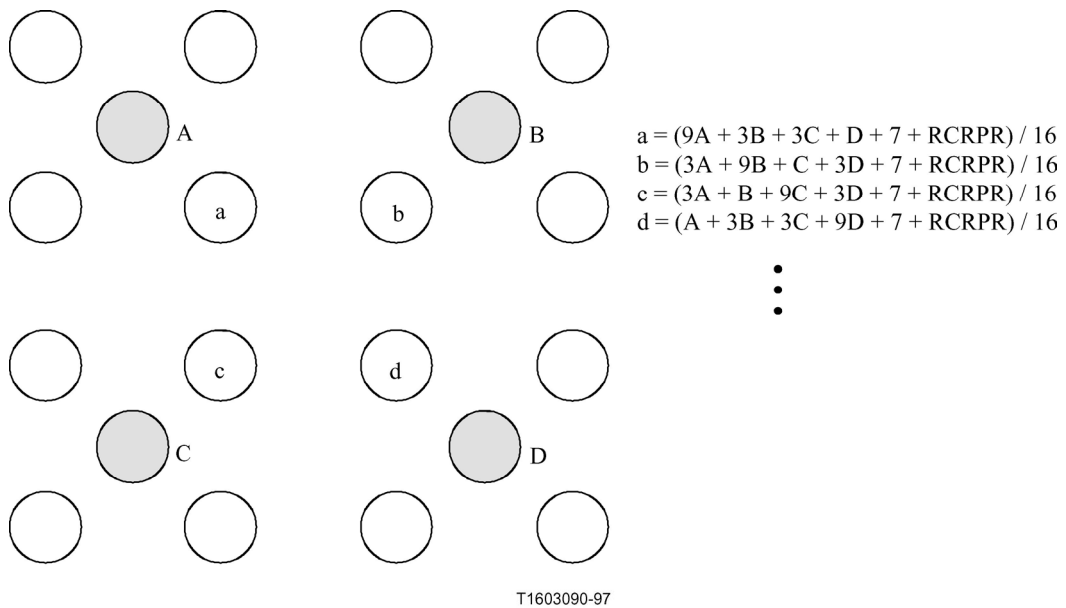
P.5 4因子再采样

4因子再采样，用因子2或1/2变换图像的水平 and 垂直尺寸，它是P.3中描述的再采样算法的特定情形，该特定情形的再采样算法的简化描述在本子节中提供。

图P.4到P.6中所使用的参数RCRPR值由MPPTYPE中的舍入类型（RTYPE）比特确定如P.3中所描述的（见5.1.4.3）。另外，图中“/”代表舍位除法。

P.5.1 4因子上采样

内部像素的4因子上采样中所使用的像素值内插方法在图P.4中显示。按照假设现有的图像外部像素依照选择的填充模式（见P.2.3和P.2.4），因此对边界像素也采用相同的内插方法。当选择clip作为填充模式时，边界像素的内插方法显示在图P.5。由于精确的4因子上采样要求采用至少1/4像素精度的x与y置换，因此为了使用该上采样方法在P.2.1中指出的卷曲置换精度（WDA）场必须置于“11”或者必须隐含地要求再采样。





-  参考图像的像素位置
-  上采样预测图像的像素位置

图 P.4/H.263—图像内部像素的4因子上采样

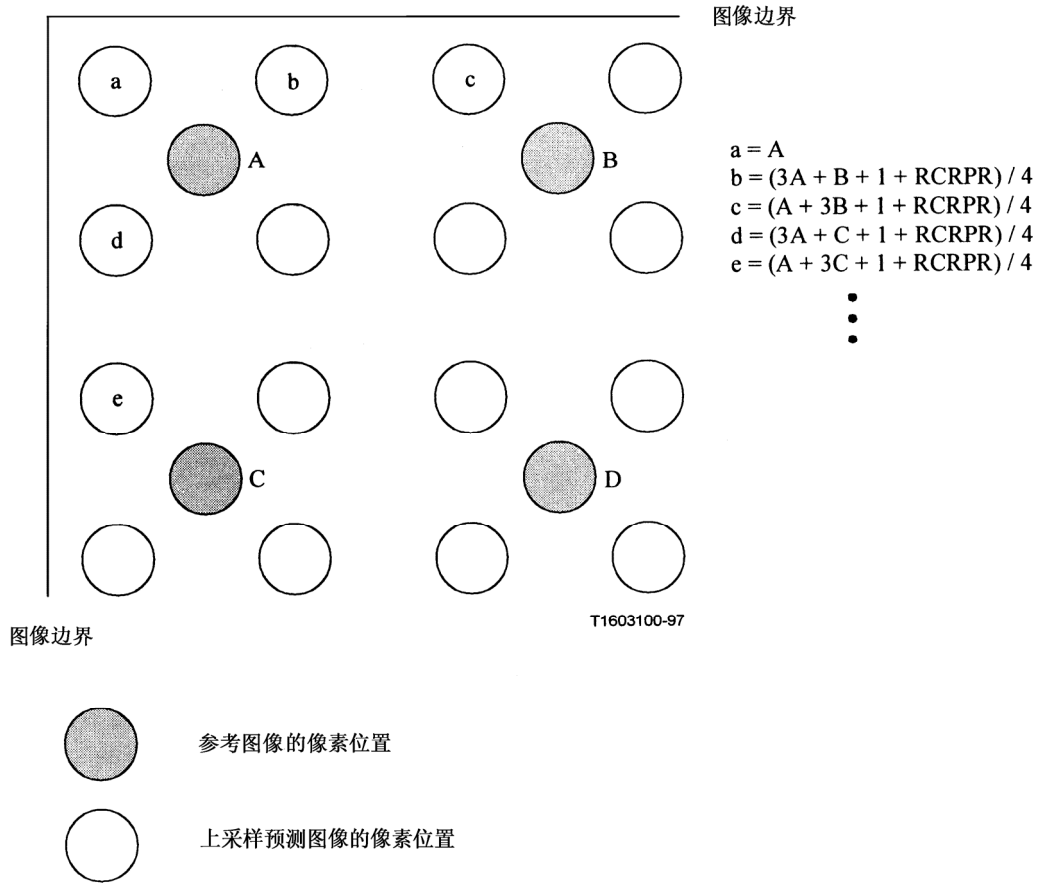


图 P.5/H.263—图像边界上像素的4因子上采样（填充模式=*clip*）

P.5.2 4因子下采样

4因子下采样的像素值内插方法在图P.6中显示。由于具有1/2像素精度的x与y置换对于精确的4因子下采样是足够的，因此允许“10”和“11”两个值作为在P.2.1中指定的卷曲置换精度（WDA）场的值（只要存在）。

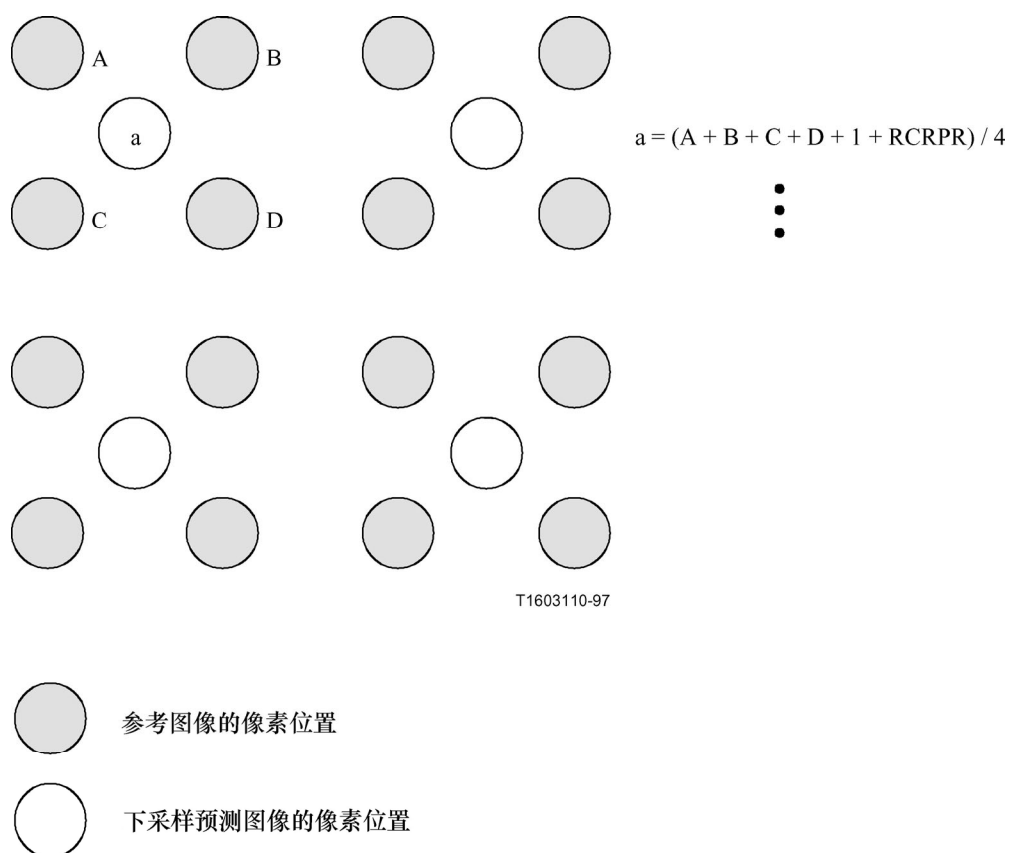


图 P.6/H.263—4因子下采样

附 件 Q

简约分辨率更新模式

Q.1 引言

本附件描述本建议书任选的简约分辨率更新模式。本模式的能力通过外部手段标示（例如ITU-T H.245建议书）。本模式的使用在图像帧头的PLUSPTYPE场中指明。

当编码剧烈活动的场景时，预期使用简约分辨率更新模式，并提供增加编码图像速率同时保持足够满意主观质量的机会。该模式允许编码器发送简约分辨率编码的图像更新信息，同时保存较高分辨率参考图像的详情以生成较高分辨率水平的最终图像。

本模式中比特流的句法同不使用本模式编码的比特流的句法相同，但该比特流的语义或理解有点不同。本模式中，由一个宏块所覆盖的图像部分是原宏块2倍宽度与原宏块2倍高度的图像部分。这样按照不在本模式的宏块数仅存在近似1/4数目的宏块。运动矢量数据也涉及正式高度和宽度两倍的块，或 32×32 与 16×16 替代正规的 16×16 或 8×8 。在其他方面，DCT或纹理数据将被设想为简约分辨率版本图像上所描述的 8×8 块。为生成最终的图像，在简约分辨率上译码纹理数据然后上采样到该图像的全分辨率。上采样之后，全分辨率纹理图像被添加到（已经全分辨率）运动补偿的图像上以生成供显示和进一步参考的图像。

本模式中，如图像帧头中所指明的具有水平尺寸 H 和垂直尺寸 V 的图像被生成供显示的最后图像。

本模式中，供预测所使用的以及供进一步译码所生成的参考图像有水平尺寸 H_R 和垂直尺寸 V_R 如4.1中所规定的它们同缺省模式中的规格相同。即 H_R 和 V_R 为：

$$H_R = ((H + 15)/16)*16$$
$$V_R = ((V + 15)/16)*16$$

其中 H 和 V 如图像帧头中指明的为水平尺寸和垂直尺寸，且“/”定义为舍位除法。

然后本附件中，纹理图像在简约分辨率水平上编码具有 H_C 和 V_C 的宽度和高度，其中：

$$H_C = ((H_R + 31)/32)*32$$
$$V_C = ((V_R + 31)/32)*32$$

且“/”定义为舍位除法。

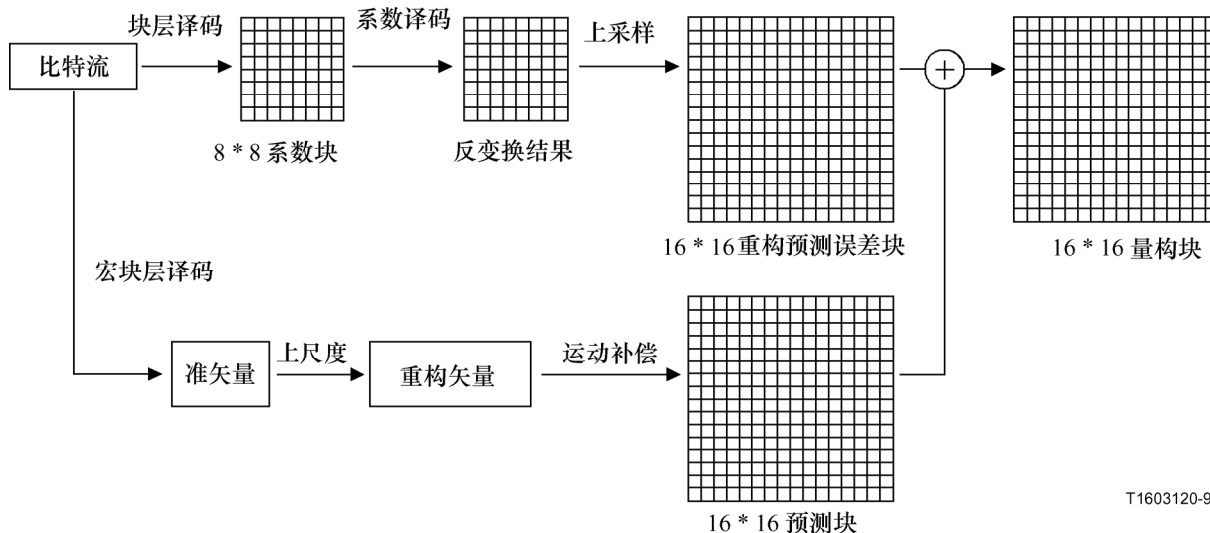
若 H_C 和 H_R ，或 V_C 和 V_R 是相互不一致的，诸如在QCIF格式中，则实施参考图像的扩展，并且图像采用把宽度和高度当做 H_C 和 V_C 的同样方式来译码。然后，由 32×32 宏块所覆盖的生成图像被在其右部和下部裁剪至宽度为 H_R 及高度为 V_R ，并存储该裁剪的图像作为供进一步译码的参考图像。

若 H 和 V 与具有宽度 H_C 和高度 V_C 的生成图像的那些宽、高均一样，则该生成图像供显示使用。否则，该生成图像进一步裁剪至尺寸 $H*V$ 并且这里所裁剪的图像仅供显示的目的使用。若时间的、SNR及空间的可量测性模式（附件O）或参考图像再采样模式（附件P）也同本选择一道使用，则当前图像的源格式可以和参数图像的源格式有所不同。在此情形中，参考图像的再采样应在译码之前实施。

注 — 本模式能够同参考图像选择模式（见附件N）一起使用而无须修正，由于当使用本选择时参考图像（经由参考图像再采样模式可能的再采样之后）具有同当前图像帧头中所指示的相同的尺寸。

Q.2 译码规程

图Q.1显示简约分辨率更新模式中块译码的方框图。



T1603120-97

图 Q.1/ H.263—简约分辨率更新模式中块译码的方框图

译码规程在以下各子节中给出：

Q.2.1 参考准备

在某些情形，可用的参数图像有不同于 H_C 和 V_C 的尺寸。然而，译码规程之前依照Q.2.1.1或Q.2.1.2参数图像应予转换。

Q.2.1.1 参数图像再采样

若时间的、SNR及空间的可量测性模式（附件O）或参数图像再采样模式（附件P）也同本选择一道使用，则当前图像的源格式可以和参数图像的源格式有所不同。在此情形依照每个附件参考图像的再采样应予首先实施。

Q.2.1.2 参数图像的扩展

若 H_R 或 V_R 不能被32除尽，诸如采用QCIF格式，则扩展参考图像。该扩展的详细规程在Q.3中定义。

Q.2.2 宏块层译码

译码可以设想为在“放大的”亮度 32×32 和色度 16×16 尺寸块上的操作。译码每个放大块的纹理与运动数据分别生成 32×32 运动块和 32×32 纹理块，如Q.2.2.1和Q.2.2.2中所述。这些运动与纹理块相加如Q.2.2.3中所述。

Q.2.2.1 运动补偿

首先，宏块运动矢量（或4个宏块运动矢量）补偿根据MVD（以及可能MVD₂₋₄）组成。若在改进的PB帧模式中，则B图像的MV_F和MV_B也根据MVDB组成。该运动矢量组成的详细规程在Q.4中规定。若当前图像模式为B图像或EP图像，则前向和后向运动矢量亦依照Q.4获得。

宏块的两个色度块的运动矢量，依照6.1.1根据宏块运动矢量获得。若先进的预测模式或分块滤波器模式在使用并因此每宏块规定4个运动矢量，则两个色度块的运动矢量依照F.2根据4个运动矢量获得。若在改进的PB帧模式中，则色度矢量的生成在附件M中规定。若B图像或EP图像在使用，则色度矢量的生成在附件O中指定。

然后，预测根据INTER宏块的运动矢量组成。4个 16×16 亮度预测块根据宏块运动矢量获得，并且2个 16×16 色度预测块根据色度运动矢量获得。对于像素的内插预测参阅6.1.2。若先进的预测模式亦在使用，则实施放大的交叠运动补偿，使用放大的加权矩阵获得4个 16×16 亮度预测块，有关的详细规程在Q.5中定义。若当前的图像模式为改进的PB帧，B图像或EP图像，则依照其他的相关附件可获得预测，除预测块的尺寸以 16×16 替代 8×8 之外。

Q.2.2.2 纹理译码

首先，依照5.4译码块层的比特流。然后系数译码并且 8×8 简约分辨率重构的预测误差块依照6.2作为反变换的结果被获得。

然后 16×16 重构的预测误差块通过上采样 8×8 简约分辨率重构的预测误差块获得。对于每个 16×16 重构的预测误差块中边缘像素的生成，使用仅属于相应块的那些像素。详细的规程在Q.6中定义。

Q.2.2.3 块重构

对每个亮度和色度块，实施预测和预测误差的求和。该规程同6.3.1中的规程相同，除块的尺寸以 16×16 替代 8×8 之外。然后依照6.3.2实施限幅操作。

然后，块边界滤波器应用于 16×16 重构块的边界像素。详细的规程在Q.7中描述。

Q.2.3 图像存储

若 H_R 和 V_R 均可被32除尽，诸如采用CIF格式，则如Q.2.2中描述的重构的生成图像作为供进一步译码的参考图像而存储。否则，诸如QCIF格式，恰用 32×32 宏块覆盖的重构的图像被在其右部和下部裁剪至宽度 H_R 及高度 V_R ，并且该裁剪图像作为供进一步译码的参考图像而存储。

Q.2.4 显示

若 H 和 V 与 H_C 和 V_C 均相同，则Q.2.2中生成的图像供显示使用。否则该生成图像进一步裁剪至尺寸 $H \times V$ ，且该裁剪图像仅出于显示的考虑而使用。

Q.3 参考图像的扩展

若 H_R 或 V_R 不被32除尽，诸如采用QCIF格式，则译码宏块/块层之前实施该参考图像的扩展。亮度的扩展的参考图像的宽度和高度是可以被32除尽的下一个较大的尺寸，以及色度的扩展的参考图像的宽度和高度是可以被16除尽的下一个较大的尺寸。

注一 在缺省模式中，参考图像的宽度和高度总被扩展到被16除尽的尺寸，只要图像格式有未被16除尽的宽度或高度，由于将被译码的图像应该视为宽度或高度具有被16除尽的下一个较大的尺寸。见4.1。

若既没有非受限运动矢量模式、先进的预测模式也没有分块滤波器模式同本选择一起使用，则扩展的像素可以为任意值，因为扩展的像素将从不作为即将重构和显示的译码图像的参数像素而使用。

若或非受限运动矢量模式，先进的预测模式或分块滤波器模式也在同本选择一起使用，则当运动矢量指向该图像右部和底部边缘之外时为了确保译码，参考图像的扩展可通过复制参考图像的边缘像素来实施。

例如，若QCIF的简约分辨率更新模式在使用，参考图像的宽度为176，高度为144，不可被32除尽。为了用 32×32 尺寸的宏块覆盖QCIF图像，宏块行数应为6，宏块列数应为5，因此扩展的参数图像的宽度为192，高度为160。

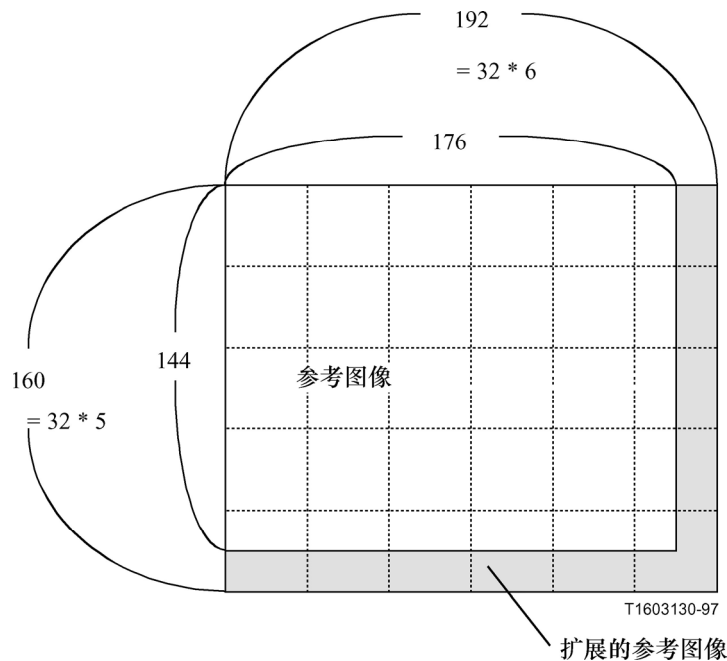


图 Q.2 / H.263—QCIF图像的扩展参考图像尺寸

在QCIF中参考图像的扩展在图Q.2中说明。亮度的扩展参考图像由以下公式给出：

$$R_{RRU}(x, y) = R(x', y')$$

其中：

- x, y = 像素域中扩展参考图像的空间坐标，
- x', y' = 像素域中参考图像的空间坐标，
- $R_{RRU}(x, y)$ = 扩展参数图像在 (x, y) 的像素值，
- $R(x', y')$ = 参数图像在 (x', y') 的像素值，

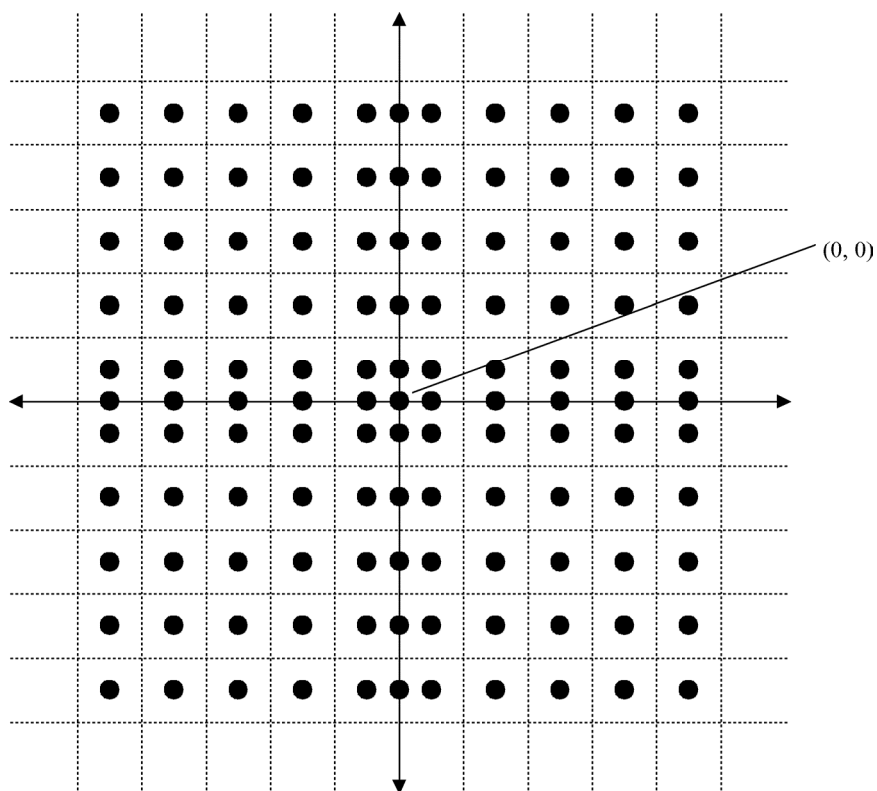
$$x' = \begin{cases} 175 & \text{若 } 175 < x < 192 \\ x & \text{其他,} \end{cases}$$

$$y' = \begin{cases} 143 & \text{若 } 143 < y < 160 \\ y & \text{其他。} \end{cases}$$

色度的参考图像也以相同的方式扩展。

Q.4 运动矢量重构

简约分辨率更新模式中，运动矢量范围放大近似为水平和垂直方向上两倍的尺寸。为了实现放大的范围使用表14中对MVD定义的VLC，每个矢量分量被限制仅为半区间或0值，因此在缺省简约分辨率更新模式中每个运动矢量分量的范围为 $(-31.5, 30.5)$ 。若非受限运动矢量模式在使用，则D.2中所定义的矢量范围 $[-\text{界限}, +\text{界限}]$ 适合于准运动矢量并且该运动矢量范围变换为 $[-(2*\text{界限} - 0.5), 2*\text{界限} - 1.5]$ 。对CIF，这意味着准运动矢量范围为 $[-32, 31.5]$ ，运动矢量范围为 $[-63.5, 62.5]$ 。若UUI场置于“01”，则运动矢量不被限定。然而，运动矢量（刚好不是准运动矢量）总被限制指向不多于编码区域之外的15个像素，如D.1.1中所描述的。图Q.3说明围绕 $(0, 0)$ 矢量值宏块运动矢量或4个运动矢量预测的可能位置。虚线代表整数坐标。



T1603140-97

图 Q.3/H.263—运动矢量重构

B图像中对于宏块使用差分运动矢量，前向和后向运动矢量预测单独获得。在简约分辨率更新模式中，亮度块的运动矢量分量 MV_c 根据 MVD 和 MVD_{2-4} 被重构如下：

- 1) 准预测矢量分量 \hat{P}_c 根据预测矢量 P_c 生成。

$$\begin{aligned} \hat{P}_c &= 0 && \text{若 } P_c = 0 \\ \hat{P}_c &= \text{符号}(P_c) * (|P_c| + 0.5) / 2.0 && \text{若 } P_c \neq 0 \end{aligned}$$

“/”代表浮点除法运算（不损失精度）。预测矢量分量 P_c 定义为矢量分量 $MV1$ 、 $MV2$ 和 $MV3$ 的中值如6.1.1和F.2的规定。

- 2) 准宏块矢量分量 \hat{M}_c 通过把来自表14的运动矢量差 MVD （和 MVD_{2-4} ）添加到 \hat{P}_c 上获得。

在缺省简约分辨率更新模式中 \hat{M}_c 的取值范围限制为 $[-16, 15.5]$ 。每对中仅有一个值将产生落入到许可范围内的 \hat{M}_c ，以类似于6.1.1中规定的方式实施规程。

若非受限运动矢量也同简约分辨率更新模式一起使用，则 \hat{M}_c 通过添加来自表D.3的运动矢量差 MVD （和 MVD_{2-4} ）来获得。

若4个运动矢量存在，则以类似于F2中规定的方式实施规程。

3) 运动矢量分量 MV_c 根据准 MV_c 通过以下公式获得:

$$MV_c = 0 \quad \text{若准}MV_c \text{等于}0$$

$$MV_c = \text{符号(准}MV_c) * (2.0 * |\text{准}MV_c| - 0.5) \quad \text{若准}MV_c \text{不等于}0$$

作为结果, 限制每个矢量分量有半整数或0值, 并且每个运动矢量分量的范围放大为近似两倍的准运动矢量范围。

4) 若当前图像模式为改进的PB帧, 或者B图像中当MBTYPE指示直接模式时, 则生成前向和后向预测的运动矢量分量 MV_F 和/或 MV_B 。

首先准运动矢量分量准 MV_F 和/或准 MV_B 根据附件O或M中定义的预测模式准则计算。

在改进的PB帧模式双向预测的情形中(见M.2.1)或B图像中当MBTYPE指示直接模式时(见O.5.2), 准 MV_F 和准 MV_B 根据准 MV_D 和准 MV_C 来计算并假设准 MV_D 为0准 MV_C 为 MV , 如附件G和M中规定的。

在改进的PB帧模式前向预测的情形中(见M.2.2)准 MV_{DB} 依据表13通过译码可变长度代码 MV_{DB} 获得。然后, 准 MV_D 通过把准 MV_{CB} 添加到准预测值上获得。为了构成准预测值, 依照M.2.2中定义的规程获得的预测值依据本子节1)条款中规定的公式变换成准预测值矢量。

在改进的PB帧模式后向预测的情形中(见M.2.3)准 MV_B 置0。

然后, 前向和后向预测的运动矢量 MV_F 和/或 MV_B 依据本子节3)条款中规定的公式根据准 MV_F 和/或准 MV_B 获得。

Q.5 亮度的放大交叠运动补偿

若先进的预测模式也同简约分辨率更新模式一起使用, 则加权值的放大矩阵被用于实施交叠的运动补偿。除每个块和加权矩阵的尺寸为 16×16 之外, 每个预测块的生成程序同F.3中描述的程序相同。

16×16亮度预测的加权值放大矩阵在图Q.4, Q.5和Q.6中给出。

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 4 | 4 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 4 |
| 4 | 4 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 4 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 6 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 |
| 4 | 4 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 4 |
| 4 | 4 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 5 | 4 | 4 |

图 Q.4/H.263—采用当前16×16亮度块运动矢量预测的加权值H₀

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 |
| 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 |
| 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 |

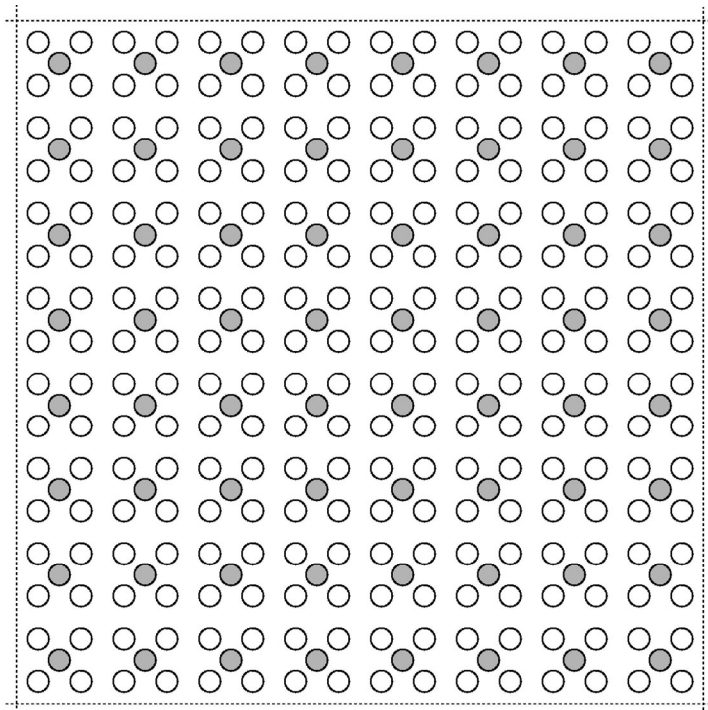
图 Q.5/H.263—在当前16×16亮度块的顶部或底部上采用当前16×16亮度块运动矢量预测的加权值H₁

| | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 |
| 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 |
| 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 |
| 2 | 2 | 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 | 2 | 2 |
| 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 |
| 2 | 2 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 |

图 Q.6/H.263—在当前16×16亮度块的左方或右方采用当前16×16亮度块运动矢量预测的加权值 H_2

Q.6 简约或分辨率重构预测误差的上采样

16×16重构预测误差块通过上采样8×8简约分辨率重构预测误差块获得。为了实现简化操作，滤波被封闭在一个块内，该块能够在块基础上实施单独的上采样操作，图Q.7显示采样点的定位。16×16重构预测误差块内部亮度和色度像素的上采样规程在Q.6.1中定义。对于16×16重构预测误差块边界上亮度和色度像素的生成，规程在Q.6.2中定义。色度块同亮度块一样被上采样。在图Q.8和Q.9中符号“/”代表舍位除法。



T1603150-97

● 8×8 简约分辨率重构预测误差块中样点的位置

○ 16×16 重构预测误差块中样点的位置

..... 块边缘

图 Q.7/H.263— 8×8 简约分辨率重构预测误差块和 16×16 重构预测误差块中样点的定位

Q.6.1 16×16重构预测误差块内部像素的上采样规程

块内部像素的重构预测误差的生成在图Q.8中描述。“/”代表舍位除法。

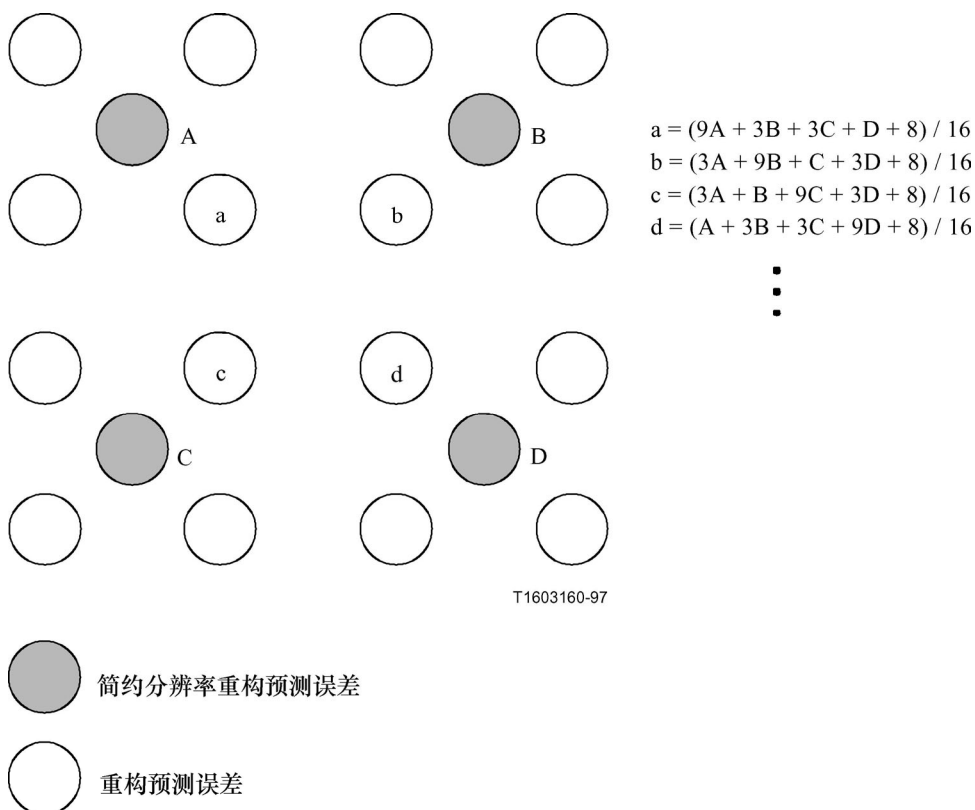


图 Q.8/ H.263—块内部像素的重构预测误差的生成

Q.6.2 16×16重构预测误差块边界上像素的上采样规程

16×16块的像素重构预测误差的生成在图Q.9中显示。

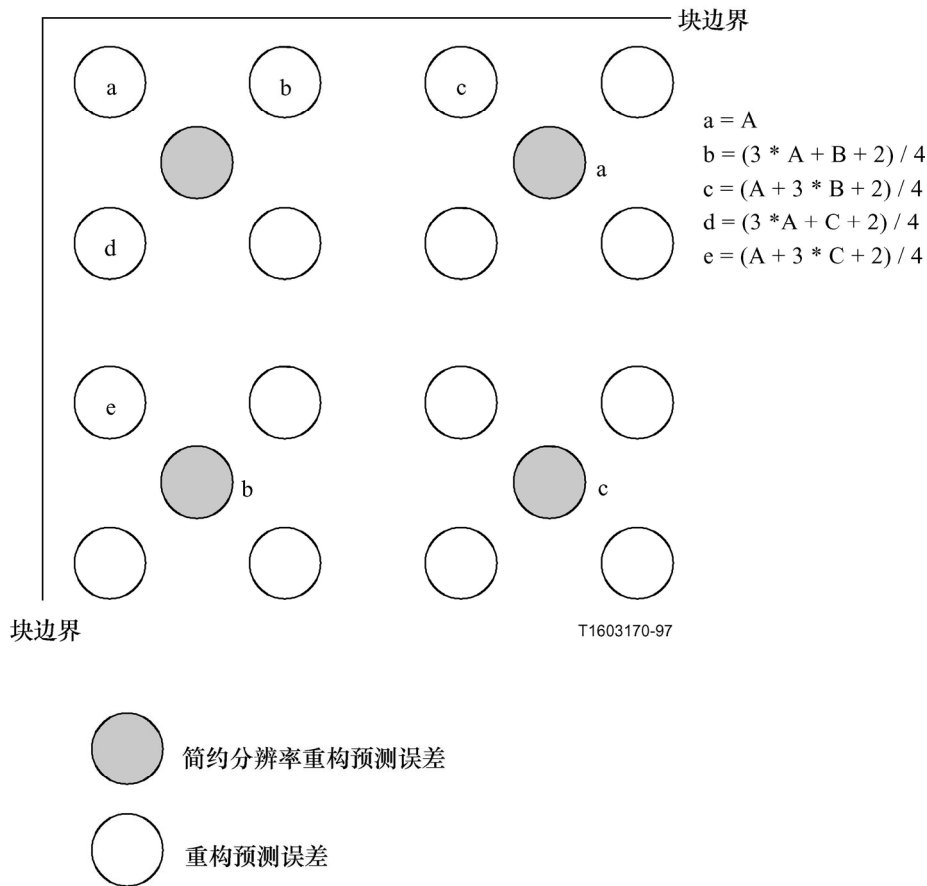


图 Q.9/H.263 一块边界像素的重构预测误差的生成

Q.7 块边界滤波器

在编码器方同样在译码器方沿着16×16重构块的边缘实施滤波器操作。存在两种可供选择的滤波，依赖于分块滤波器模式是否在使用。

在简约分辨率更新模式中缺省滤波依照Q.7.1实施。

若分块滤波器模式也同简约分辨率更新模式一起使用，则滤波依照Q.7.2实施。

以上两种情形中，在存储供未来预测使用的图像数据之前，在完整的重构图像数据上实施滤波。跨越图像边缘、截面结构模式中的截面边缘（见附件K）或独立分段译码模式中具有GOB头的GOB边界（见附件R）的任何滤波均不实施，色度数据同亮度数据一样滤波。

Q.7.1 缺省块边界滤波器定义

在简约分辨率更新模式中，缺省滤波依照本小节（的规程）实施。

若A和B是重构图像的水平线或垂直线上的两个像素值，A属于一个16×16块称之为块1而B属于邻近的16×16块称之为块2，它在块1的右方或者下方。图Q.10显示这些像素位置的实例。

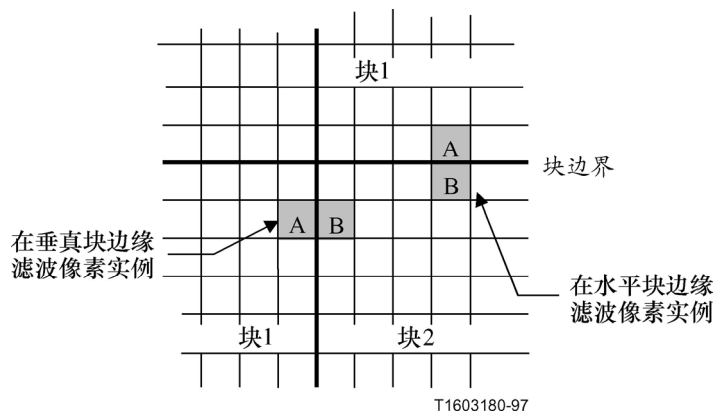


图 Q.10 /H.263—缺省块边界滤波器

为了对特殊的边缘开通滤波器以下条件之一必须满足：

- 块1属于编码的宏块（COD=0 || MB 类型=INTRA）；或
- 块2属于编码的宏块（COD=0 || MB 类型=INTRA）。

A将用A₁替代B将用B₁替代。“/”代表舍位除法。

$$A_1 = (3 * A + B + 2) / 4$$

$$B_1 = (A + 3 * B + 2) / 4$$

实施滤波所在的边缘排序同J.3中提供的描述相同。

Q.7.2 分块滤波器模式使用时块边界滤波器的定义

若分块滤波器模式（见附件J）同简约分辨率更新模式一起使用，则在16×16亮度和色度块边界像素上应实施采用一个修正的附件J中定义的滤波，来代替Q.7.1中描述的滤波。附件J中该滤波的一个修正是给参数STRENGTH正无限的值。这隐含着J.3中定义的UPDownRamp（x, STRENGTH）函数变成为x的线性函数。

作为结果，J.3中描述的分块滤波器程序采用下列形式重新定义：

$$B_1 = \text{clip}(B + d_1)$$

$$C_1 = \text{clip}(C - d_1)$$

$$A_1 = A - d_2$$

$$D_1 = D + d_2$$

$$d_1 = (A - 4B + 4C - D) / 8$$

$$d_2 = \text{clipd1}((A - D) / 4, d_1/2)$$

附件 R

独立分段译码模式

R.1 引言

本附件描述本建议书的任选独立分段译码模式，它允许将要译码的图像无须存在跨越截面边界或跨越具有非空GOB头的GOB边界的任何数据相关性。本模式的使用在图像帧头的PLUSPTYPE场中指示。使用本任选模式的能力通过外部手段协商（例如，ITU-T H.245建议书）。

当指示使用本模式时，视频图像分段边界（由截面的边界或者发送GOB头的GOB的上边界，或者图像边界所定义的，无论那种情况应为最小方式的区域边界）被当作译码时的图像边界来处理，包括跨越那些边界的（当非受限运动矢量模式、先进的预测模式、分块滤波器模式，或时间的、SNR及空间的可量测性模式使用时导致边界外推的，以及当那些任选模式没有一个使用时禁用的）运动矢量的处理。

R.2 模式操作

视频图像分段由以下段落定义。

若截面结构模式（见附件K）未使用，那么一个GOB或者两个以上数目的连续的GOB构成一个视频图像分段。每个视频图像分段的顶部位置由存在的非空GOB头来指示，对于该非空GOB头部而言，视频图像分段的边界刚好落在头部存在的GOB中的宏块上，或者落在图像的顶部，无论那种情况应为顶部较低的哪种位置。每个视频图像分段的底部位置由下一个视频图像分段的顶部，或者图像的底部来定义，无论那种情况应为底部最靠上的哪个位置。

若截面结构模式（见附件K）在使用，那么每个截面组成一个视频图像分段。

在独立分段译码模式中，每个视频图像分段采用完全独立于所有其他视频图像分段的译码，并且也与参考图像中相同视频图像分段位置之外的所有数据无关。这包括：

- 1) 对运动矢量预测而言，不使用任何当前视频图像分段之外的运动矢量（如6.1.1）；
- 2) 当先进的预测模式使用时，不使用任何当前的视频图像分段之外的运动矢量作为交叠的块运动补偿的间接运动矢量（见F.3）；
- 3) 无任何跨越视频图像分段边界的分块滤波器操作（见J.3）；
- 4) 不使用任何当前的视频图像分段以外的参考数据的运动矢量，除非非受限运动矢量模式（见附件D）、先进的预测模式（见附件F）、分块滤波器模式（见附件J）或者时间的、SNR及空间的量测性模式（见附件O）在使用；在那些情形中，先导图像中当前视频图像分段的边界被外推如附件D所描述的构成参照边界区域之外的像素预测；
- 5) 在空间可量测性EI和EP图像中对向上预测而言，无任何双线性内插跨越相应于当前视频图像分段的1/4尺寸或1/2尺寸区域的边界（如附件O中所规定的）；

- 6) 当简约分辨率更新模式（见附件Q）使用时，无任何块边界滤波器操作跨越视频图像分段边界；
- 7) 不同独立分段译码模式一道使用参考图像再采样模式。

R.3 用法限制

当独立分段译码模式使用时，在视频编码句法其他方面使用上设置某些限制。构成这些限制以预防两种致病的情况，否则将使独立分段译码模式的操作难于运行。

R.3.1 分段形状限制

在使用截面结构模式（附件K）而不使用矩形截面子模式时（见K.1），这将可能产生视频图像分段形状可为非凸的情形（具有“内部拐角”，或者甚至组成该图像的两个明显不同的及分离的区域）。

因此，独立分段译码模式不应同不具有矩形截面子模式同时使用的截面结构模式一道使用（见附件K）。为了确定如何及何时实施每个视频图像分段的外推，强制该限制可以防止难以满足的特定情况处理要求。

R.3.2 分段形状变化限制

若允许视频图像分段的形状在比特流中以任意的方式从图像到图像变化，则可能产生比特流将很难译码的情况。这是因为比特流中运动矢量可能出现之前要求适当理解视频图像分段形状的知识，而在此情形中比特流容量对于确定每个视频图像分段的形状是不充分的。

因此，当独立分段译码模式使用时，对所有允许时间预测的图像和帧而言（例如所有的P、B和EP图像及所有的改进的PB帧），视频图像分段应与其时间参考图像中所使用的视频图像分段相同。同样，当独立分段译码模式使用时，所有EI图像的视频图像分段应或是与其参考图像中所使用的视频图像分段相同或者差别仅为子分划。同样，如果独立分段译码模式不在当前图像的所有参考图像中使用，则在使用参考图像的任何图像和帧中（所有图像类型除INTRA外）独立分段译码模式不应使用。作为该限制的结果，在独立分段译码模式中，视频图像分段的形状应该从图像到图像从不变化，作为I和EI图像中的变化除外（并且EI图像能够改变分段的方式自身也有几分限制）。

附件 S

可供选择的INTER VLC模式

S.1 引言

本附件描述本建议书任选的可供选择的INTER VLC模式，当图像中发生明显变化时，它可以改善帧间图像的编码效率。通过允许某些原本为INTRA图像设计的VLC代码提供给某些INTER图像系数和CBPY数据使用以获取编码效率的改善。本模式的使用在图像帧头的PLUSPTYPE场中指示。使用本任选模式的能力通过外部手段协商（例如ITU-T H.245建议书）。该模式包含两个句法改动，其一对INTER系数的编码，其二对INTER CBPY值的编码。

S.2 系数的可供选择的INTER VLC

附件I的INTRA VLC表格设计背后的想法是使用与原始INTER VLC 同样的码字，但具有不同的LEVEL和RUN理解。INTER VLC较好的适合于存在众多和/或大赋值系数的情形。

如此构造INTRA VLC以致在INTER和INTRA表中对LAST（0或1）码字均有相同的值。因此INTRA表通过“改组”具有相同LAST的值的码字含义而生成。更进一步，对于具有大的|LEVEL|的事件而言，INTER表使用在INTER表中具有大RUN的码字。在具有大量的大幅值系数的INTER 块中，使用INTRA表比使用INTER表有时更有效，并在此情形中选择哪种VLC表对译码器而言可以很明确，因为使用INTER表译码将导致RUN值如此巨大以致于每个块指示出现多于64个系数。在这些环境下，可以使用INTRA表改善INTER编码的效率。

S.2.1 编码器行为

编码器可以使用INTRA VLC表编码INTER块，译码器随时能够检测到它的使用 — 换句话说，每当使用INTER VLC表译码将引起块的64个系数之外的系数寻址时。

仅当以上情况被满足并且也当对同一系数值INTRA VLC使用导致比INTER VLC使用更少的比特位时，编码器将正式选择使用INTRA VLC表供编码INTER块使用。由于产生INTRA VLC的方式（由于对同一码字而言，通常的INTER VLC表包含长的游程长度，同样情况下INTRA VLC表包含大的系数幅值），当存在众多大的系数时，这将是经常的选择。

S.2.2 译码器行为

译码过程如下：

- 1) 首先译码器接收块的所有系数编码。
- 2) 然后码字被理解为假设使用INTER VLC，若系数的寻址驻留在块的64个系数之内，则VLC译码完成。
- 3) 若块之外系数被寻址，则码字应理解为依照INTRA VLC。

S.3 CBPY的可供选择的INTER VLC

INTER CBPY码字（表12）设计具有以下假设：具有全零系数的Y块存在比具有至少一个非零系数的Y块存在要多。当 C_B 和 C_R 块均具有至少一个非零系数时，及 $CBPC_5=CBPC_6=1$ ，则该假设不再适用。由于这个理由，当可供选择的INTER VLC模式使用时，INTRA宏块的CBPY码字，如表12中所定义的，也应供INTER宏块使用，无论何时只要 $CBPC_5=CBPC_6=1$ 。

附 件 T

修正的量化模式

T.1 引言

本附件描述本建议书任选的修正量化模式，它修正量化器操作。本模式的使用在图像帧头的PLUSPTYPE场中指示。使用本任选模式的能力通过外部手段协商（例如，ITU-T H.245建议书）。

本模式包括4个关键特征：

- 1) 编码的比特速率控制能力通过改变DQUANT场的句法来改进。
- 2) 色度保真度通过对色度比对亮度数据指定较小的步长尺寸来改进。
- 3) 扩展可表示的系数值范围以允许任何可能的真实系数值表示在量化步长尺寸所允许的精度内。
- 4) 为改进误差检测能力和极小化译码复杂度，限制量化系数等级的范围到能够合理出现的那些范围。

T.2 修正的DQUANT更新

本模式修正DQUANT场的语义。采用本模式，使用DQUANT或通过加或减去一个小的量值修正QUANT或标示任意指定的QUANT新值是可能的。小量值尺寸的修正依赖于QUANT的当前值。通过使用本模式，更为灵活的量化器步长尺寸控制能够在DQUANT场中指定。

本模式中DQUANT的码字不再是2比特固定长度场。它是可变长度场，能够或为2比特或为6比特长度。它是否为2或6比特取决于该代码的首比特。因此以下描述被分成两部分，取决于首比特。

T.2.1 小步长QUANT改动

当DQUANT场的首比特为1时，DQUANT中仅发送一个附加的比特。使用该单个附加比特用差值修正QUANT。QUANT值的改变取决于DQUANT的第二比特以及QUANT的先前值，如表T.1中所示。

实例：若QUANT的先前值为29，且DQUANT用码字“11”标示，那么差值为+2，这样产生的新QUANT值为31。

表 T.1/H.263—小步长QUANT的语义改动

| 先前的QUANT | QUANT的改变 | |
|----------|-----------|-----------|
| | DQUANT=10 | DQUANT=11 |
| 1 | +2 | +1 |
| 2-10 | - 1 | +1 |
| 11-20 | - 2 | +2 |
| 21-28 | - 3 | +3 |
| 29 | - 3 | +2 |
| 30 | - 3 | +1 |
| 31 | - 3 | +5 |

T.2.2 任意的QUANT选择

当DQUANT场的首比特为0时，DQUANT中发送5个附加比特，随后的5个比特表示新的QUANT如5.1.19中所定义的。

实例：不管QUANT的当前值，若DQUANT用码字“001111”标示，那么QUANT的新值为15。

T.3 改动的色度系数量化步长尺寸

当修正的量化模式使用时，色度系数的量化参数不同于亮度的量化参数。亮度量化参数在比特流中标示。它被称为QUANT。当本模式使用时，称为QUANT_C的不同的量化参数供色度系数的反量化所使用。QUANT和QUANT_C之间的关系在表T.2中给出。若分块滤波器模式（见附件J）在使用，则QUANT_C也应用于分块滤波器对色度数据的使用。在此处，在任何其他正文中无论何时讨论QUANT时，它应意味着亮度的步长尺寸。

表 T.2/H.263—QUANT和QUANT_C之间的关系

| QUANT的范围 | QUANT_C的值 |
|----------|-------------------|
| 1-6 | QUANT_C=QUANT |
| 7-9 | QUANT_C=QUANT - 1 |
| 10-11 | 9 |
| 12-13 | 10 |
| 14-15 | 11 |
| 16-18 | 12 |
| 19-21 | 13 |
| 22-26 | 14 |
| 27-31 | 15 |

T.4 修正的系数范围

当修正的量化模式使用时，具有大于127的量化等级幅度的量化DCT系数也能够表示。这两个优点：

- 1) 编码器操作通过允许表示真正全标度的可能系数值达到改善。
- 2) 在面对某些大的系数值不增加量化步长尺寸将不能表示的情况下，编码器复杂度通过消除增加量化步长尺寸的需求得以减少。

编码器中量化之前DCT系数的准确值将达到幅度2040是完全可能的。这样，无论何时只要量化参数QUANT或QUANT_C小于8。LEVEL的-127到+127的取值范围对于覆盖全部可能的系数值范围是不能胜任的。扩展的系数范围扩大了LEVEL的取值范围允许任何真实系数值更为适当地编码。

当修正的量化模式使用时，跟随ESCAPE码（000011，按照5.4.2）的LEVEL场的含义被改动。本模式中，比特序列10000000未被禁用，它用于表示EXTENDED_ESCAPE代码。幅度大于127的AC系数表示通过发送EXTENDED_ESCAPE码，即时跟随11比特固定长度的EXTENDED_LEVEL场来实现。扩展系数值被编码成EXTENDED_LEVEL场，该场通过取LEVEL的二的补码的二进制表示的最低有效11比特并且以右移5比特位周期地循环它们来实现。该循环对于防止起始码仿真是必要的。周期循环在图T.1中说明。



图 T.1/H.263一周期循环的系数表示

T.5 用法限制

当修正的量化模式使用时，在编码的系数值上设置某些限制。这有几点好处：

- 1) 比特误差的检测能力通过禁止某些不合理的系数值得以改善，这样允许译码器将这些值识别为比特差错；并且
- 2) 译码器复杂度通过限幅之前减少反量化的必要字长得以降低。
- 3) 对编码系数使用T.4中描述的EXTENDED_ESCAPE方法防止起始码仿真。

这些限制如下。当修正的量化模式使用时：

- 1) 对任何系数而言，如适当使用QUANT或者QUANT_C的当前值，以及LEVEL的编码值，由6.2.1中描述的反量化处理所生成的重构电平幅度|REC|应小于4096。此附加的限制适用于所有系数，不管该系数是否使用EXTEDNED_ESCAPE机制来发送。

- 2) 比特流将不使用或正规的ESCAPE代码或者EXTENDED_ESCAPE代码来编码LAST、RUN和LEVEL的组合，对于这些组合在可使用的VLC表中存在适当的码字条目，该表或为表16（见5.4.2）或为表I.2（见I.3）。
- 3) 仅当系数（QUANT或QUANT_C）的量化参数小于8时，才使用EXTENDED_ESCAPE代码。
- 4) 仅当EXTENDED_ESCAPE代码后随EXTENDED_LEVEL场表示处于-127到+127范围之外的LEVEL值时，才使用EXTENDED_ESCAPE代码。

附 件 U

增强参考图像选择模式

U.1 引言

本附件描述增强参考图像选择（ERPS）模式，该模式在本建议书中是可选的。如果要使用ERPS模式，需要通过外部手段进行能力协商（例如ITU-T H.245建议书）。译码器中容纳的附加图像存储的数量也可以通过外部手段标示。如果要使用该模式，应该将PLUSPTYPE(OPPTYPE)可选部分中预留的第16比特前置为1。该模式通过使用参考图像存储缓存改善了错误恢复和编译码效率。

子图像移除是由ERPS模式的一个子模式规定的。子图像移除的目的是减少储存多个参考图像所需要的存储器的数量。将每个参考图像分割成较小的矩形单元来减少存储器的数量，这些较小的矩形单元称为子图像。然后编码器指示译码器特定参考图像的特定子图像区域将不被用做后续图像的参考，这样就可以使译码器中存储这些区域的存储器能够存储其他参考图像。本附件规定的对于ERPS模式的支持以及为子图像移除而分解图像存储器成为多个最新图像单元（MPUs）也需要通过外部手段进行能力协商（例如ITU-T H.245建议书）。

B图像中两个图像的后向预测是由ERPS模式的一个子模式规定的。通过为B图像提供使用多个图像前向预测以及使用超过一个参考图像进行反向预测能力的编码器，该子模式可以增强编译码性能。对于该子模式的支持也需要通过外部手段进行能力协商（例如ITU-T H.245建议书）。

对于错误恢复，ERPS模式可以使用通过外部手段标示的反向信道信息（例如ITU-T H.245建议书），该信息由译码器发到编码器，通知编码器哪个图像或者图像的哪一部分不能正确译码。ERPS模式能够提供相对于附件 N规定的参考图像模式（RPS）增强的性能。该模式不能与RPS同时使用（它能够被用来提供本质上与RPS模式相同的功能）。

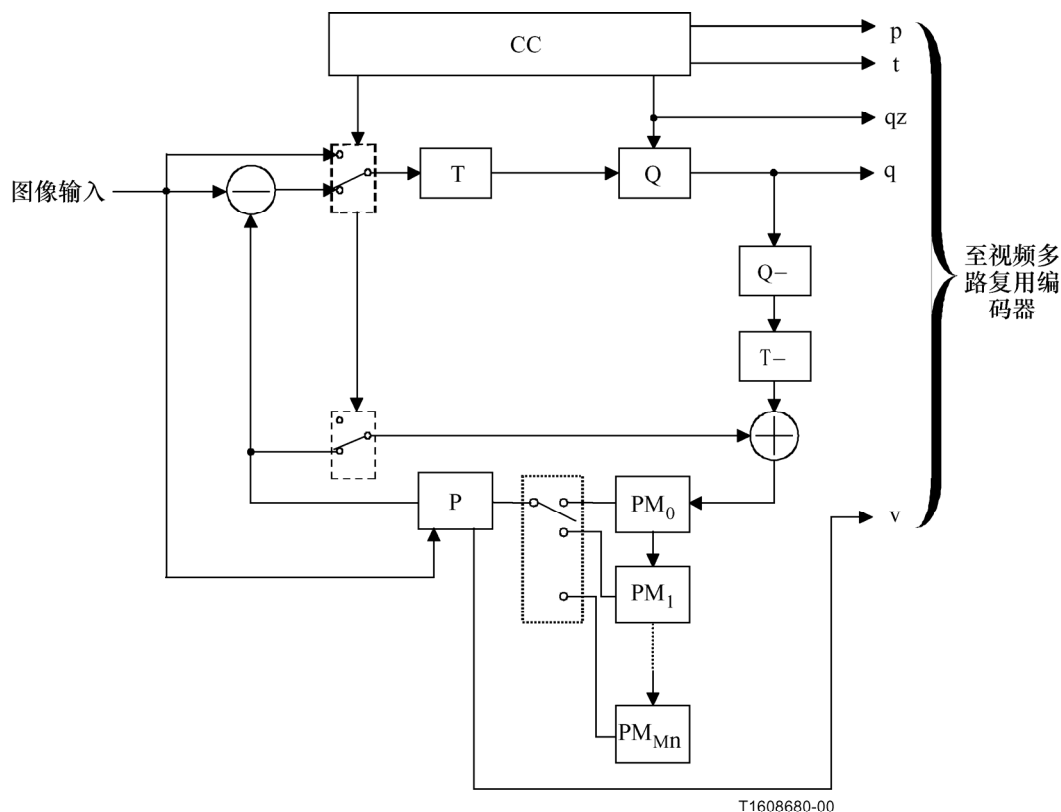
对于编码效率，运动补偿可以扩展为由多个图像预测。运动补偿扩展为多图像预测是通过扩展每个运动矢量的一个图像参考参数完成的，该图像参考参数是用来定位在任一多参考图像中用作运动补偿的一个宏块或者块预测区。该图像参考参数是可变长度编码，用来说明一个相关缓冲索引。参考图像集中到一个由编码器控制的缓冲方案中。

ERPS模式不能与基于句法的算术编码模式（见附件E）或者数据分割截面模式（见附件V）一起使用。

ERPS模式一旦激活，在后续比特流图像中将不能去激活，除非初始去激活出现在一个I 或者EI图像中，并且任何的重新激活也出现在I或者EI图像中，通过缓存器复位来完成（RESET等于1）。如果去激活成功，ERPS多图像缓冲区的所有内容应设为“未使用”模式。

U.2 视频源编码算法

图U.1显示源编码器的一般形式，主要是使用多个图像存储器的源编码器结构。



- T 变换
 Q 量化器
 P 具有运动补偿可变延迟的图像存储器
 PM 图像存储器
 CC 编码控制
 p INTRA/INTER标记
 t 传输与否标记
 qz 量化器指示
 q 变换系数的量化索引
 v 运动向量

图 U.1/H.263—增强参考图像选择模式的源编码器

视频源编码算法可以扩展为多图像的运动补偿。允许在宏块级上的参考图像选择可以提高编码效率。使用相关索引的图像缓存机制可以在多图像的缓存中有效的定位图像。这种多图像缓存控制机制可以以完全不同的两类方式工作。

在第一种工作方式中，缓存控制单元中容纳一个随时间变化的“滑动窗口”。在这种机制下， M 个图像内存 $PM_0 \dots PM_{M-1}$ 中存储最近处理中（最多 M 个）译码和重构的图像，这些图像可以用于作为译码的参考图像。如果多图像缓存中图像的最大数量与 M 相符合，运动预测在对图像 m 编码时，如果 $0 \leq m \leq M - 1$ ，就可以利用 m 图像。“适用存储器控制”可以作为第二种可选的操作，它可以作为一种相对简单“活动窗口”机制更为灵活和具体的控制图像存储器的方式。

ERPS模式的使用会导致某些已经发送到译码器的图像或者图像的子图像域状态被指定为“未使用”。一旦某些图片或者图片域被指定为“未使用”，比特流中就不能够包含任何指向“未使用”域进行后续图像预测的数据。通过管理先前图像的“未使用”状态的分配，编码器应该保证译码器有足够的存储器来储存显示后续图像所需要的所有数据。缓存器的全部尺寸和结构都经由比特流传送给译码器，这样编码器能够控制缓存器使存储的没有分配为“未使用”状态的图像数据量不会超出缓存器的总容量。

源编码器可以选择图像存储器的一个或者几个来防止图像内编码导致的临时错误的传播。独立分段译码模式（见附件 R）将非空头GOBs或者截面的边界作为图像边界，当这种模式被应用到比图像更小的单元上，例如一个GOB或者截面，由于运动补偿跨越GOBs或者截面边界，可以防止错误在空间上的传播。用来标识哪个图像被选择预测的信息包含在编码后的比特流中。

编码器选择哪个或者哪几个图像作为预测用图像超出本建议书的范围。

U.3 前向信道句法

该句法可以在图像层，块组层和截面层改变。如果参数MRPA等于"1"，该句法也可以在宏块层中改变。在图像层，块组层和截面层中，插入了增参考图像选择层（ERPS层）。在宏块层中，图像参考参数的插入是为了在某种条件下激发多图像运动补偿。

U.3.1 图像层，块组层和截面层的句法

图U.2所示为PLUS头的增强参考图像选择句法（其他的如图8所示）。RPSMF, PN 和ERPS层域插入到PLUS头中。TRPI, TRP, BCI和BCM域在这里没有列出（它们仅在附件N的RPS模式中需要，但是RPS模式在ERPS模式使用的时候是禁止的）。

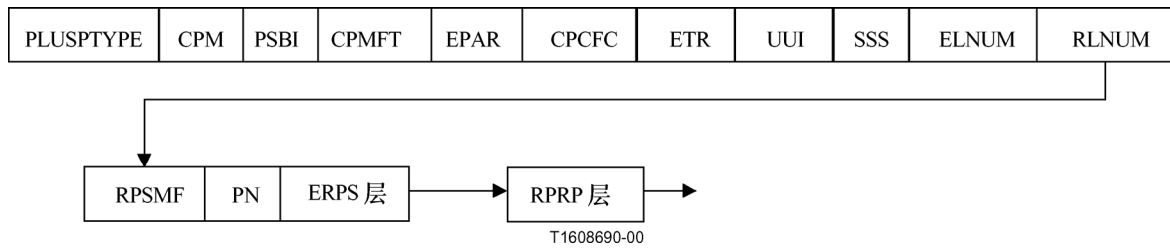


图 U.2/H.263—ERPS模式的PLUS头结构

GOB层的句法如图3所示。句法中加入了PNI, PN, NOERPSL和ERPS层域（其他在图9中定义）。

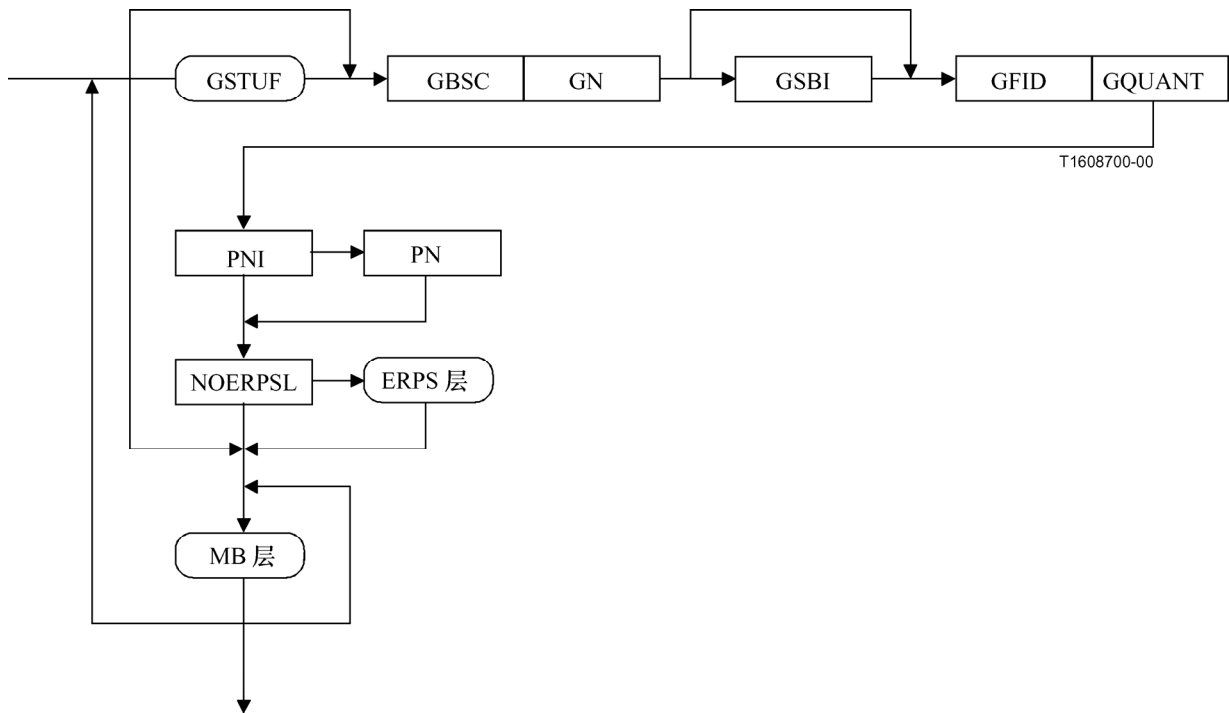


图 U.3/H.263—ERPS模式的GOB层结构

当使用可选截面结构模式（见附件K）时，截面层句法的改动和GOB层的方式相同。截面层句法如图U.4所示。在比特流中，截面紧随在图像起始码之后，包括了所有的附加域：PNI, PN, NOERPSL和ERPS层。

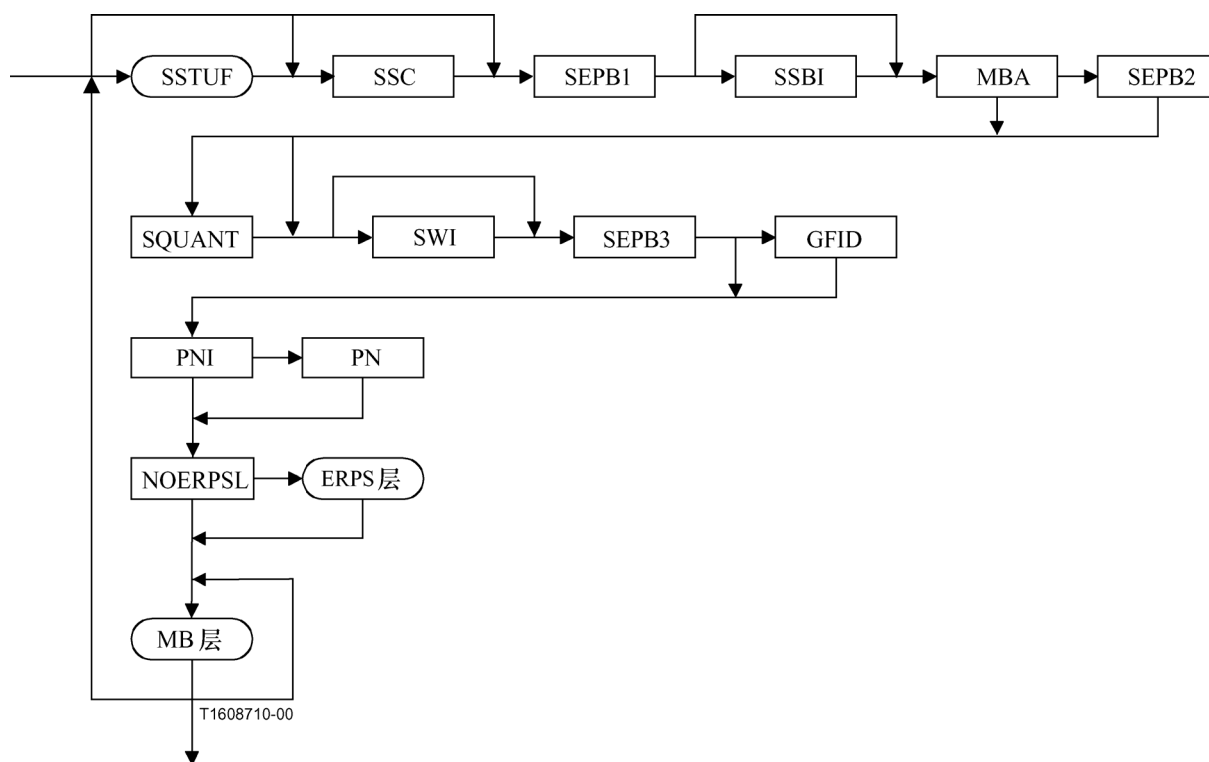


图 U.4/H.263—ERPS模式的截面层结构

ERPS层如图U.5所示。

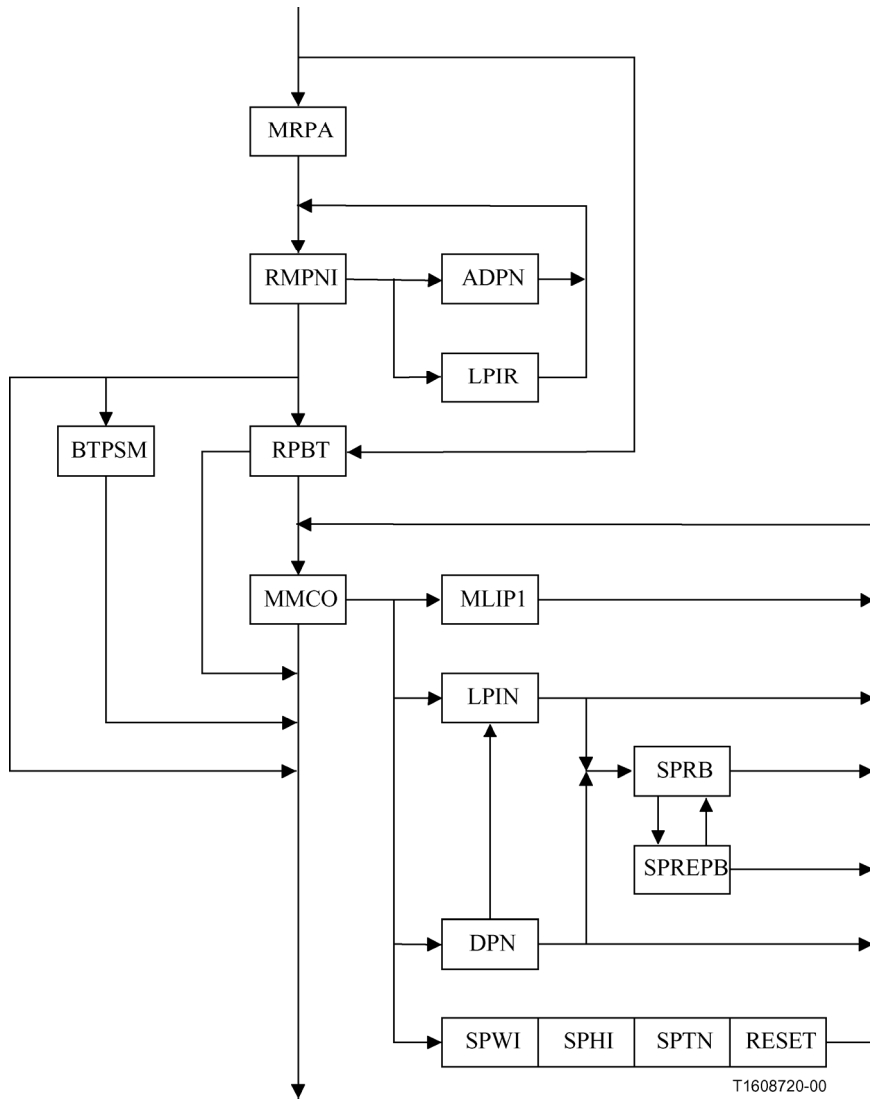


图 U.5/H.263—ERPS层的结构

ADPN, LPIR, MLIP1, DPN, LPIN, SPTN, PR, PR₀, PR₂, PR₃, PR₄, PRB and PRFW 域的可变长度编码如表U.1所示。

表 U.1/H.263—ADPN, LPIR, MLIP1, DPN, LPIN, SPTN, PR, PR₀, PR₂, PR₃, PR₄, PRB和 PRFW可变长度编码

| 绝对位置 | 比特数 | 编 码 |
|---|-----|--|
| 0 | 1 | 1 |
| "x ₀ " ⁺¹ (1:2) | 3 | 0x ₀ 0 |
| "x ₁ x ₀ " ⁺³ (3:6) | 5 | 0x ₁ 1x ₀ 0 |
| "x ₂ x ₁ x ₀ " ⁺⁷ (7:14) | 7 | 0x ₂ 1x ₁ 1x ₀ 0 |
| "x ₃ x ₂ x ₁ x ₀ " ⁺¹⁵ (15:30) | 9 | 0x ₃ 1x ₂ 1x ₁ 1x ₀ 0 |
| "x ₄ x ₃ x ₂ x ₁ x ₀ " ⁺³¹ (31:62) | 11 | 0x ₄ 1x ₃ 1x ₂ 1x ₁ 1x ₀ 0 |
| "x ₅ x ₄ x ₃ x ₂ x ₁ x ₀ " ⁺⁶³ (63:126) | 13 | 0x ₅ 1x ₄ 1x ₃ 1x ₂ 1x ₁ 1x ₀ 0 |
| "x ₆ x ₅ x ₄ x ₃ x ₂ x ₁ x ₀ " ⁺¹²⁷ (127:254) | 15 | 0x ₆ 1x ₅ 1x ₄ 1x ₃ 1x ₂ 1x ₁ 1x ₀ 0 |
| "x ₇ x ₆ x ₅ x ₄ x ₃ x ₂ x ₁ x ₀ " ⁺²⁵⁵ (255:510) | 17 | 0x ₇ 1x ₆ 1x ₅ 1x ₄ 1x ₃ 1x ₂ 1x ₁ 1x ₀ 0 |
| "x ₈ x ₇ x ₆ x ₅ x ₄ x ₃ x ₂ x ₁ x ₀ " ⁺⁵¹¹ (511:1022) | 19 | 0x ₈ 1x ₇ 1x ₆ 1x ₅ 1x ₄ 1x ₃ 1x ₂ 1x ₁ 1x ₀ 0 |
| "x ₉ x ₈ x ₇ x ₆ x ₅ x ₄ x ₃ x ₂ x ₁ x ₀ " ⁺¹⁰²³ (1023:2046) | 21 | 0x ₉ 1x ₈ 1x ₇ 1x ₆ 1x ₅ 1x ₄ 1x ₃ 1x ₂ 1x ₁ 1x ₀ 0 |
| "x ₁₀ x ₉ x ₈ x ₇ x ₆ x ₅ x ₄ x ₃ x ₂ x ₁ x ₀ " ⁺²⁰⁴⁷ (2047:4094) | 23 | 0x ₁₀ 1x ₉ 1x ₈ 1x ₇ 1x ₆ 1x ₅ 1x ₄ 1x ₃ 1x ₂ 1x ₁ 1x ₀ 0 |

U.3.1.1 参考图像选择模式标志 (RPSMF) (3比特)

RPSMF是一个3比特固定长度的码字。只要使用EPRS模式，它就会出现在PLUS头中。RPSMF模式不应该出现在GOB或者截面层中。如果出现，RPSMF指示的是编码器需要哪种类型的后向信道消息。RPSMF的值遵守5.1.13的定义。

U.3.1.2 图像编号指示符 (PNI) (1比特)

PNI是一个单比特固定长度码字。当使用EPRS模式时，它会一直出现在GOB或者截面层中，不应该出现在PLUS头中。如果出现，PNI指示的是后续PN域是否出现。

"0": PN域不出现。

"1": PN域出现。

U.3.1.3 图像编号 (10比特)

PN是一个10比特固定长度码字。当使用EPRS模式时，它会一直出现在PLUS头中。

PN的递增步长为1。对于每一个编码并传输的图像，PN值是通过前面存储图像的PN值加1后取10比特模的操作得到的。在U.3.1.5.7中给出术语“存储图像”的定义。对于EI和EP图像，PN值应该是由相同可测量增强层中的前一个存储的EI或者EP图像值递增得到的。对于B图像，PN值是由B图像参考层中当前最近存储的非B帧图像的值递增得到，这里所指的B帧图像在比特流顺序中位于B图像之前（该图像在时间域上滞后于B图像）。由于B图像不作为后续图像的参考，在多图像缓存中并不存储B图像。这样，在B图像参考层中紧随B图像之后的那个图像或者紧随B图像后的其他B图像与该B图像有相同的PN值。类似的，如果比特流中有一个不需要存储的非B图像，该非B图像后的图像（在相同的增强层，在附件O的操作案例中）也应该和它有相同的PN值。

在使用通称为“视频冗余度编码”方案中，参考图像选择模式可以由某些编码器使用采用一种对同一时间瞬间描绘的场景发送多种表示的方式（通常使用不同的参考图像）。在这样的情形中，参考图像选择模式在使用并且比特流中毗邻的图像具有相同的时域参考，译码器应把此事当作近似同一描绘场景内容的冗余复制曾经发送的指示，并将译码和使用最先接收的图像而丢弃后续的冗余图像。

图像缓存器中1024个编码和存储的每一个图像都有唯一的PN值。因此，图像缓存器中的图像超过1023之后就不能再存储图像了（在相同的增强层，在附件O的操作案例中），除非该图像已经按照下面的说明被分配了一个长效图像索引。编码器应该确保比特流不需要再说明任何超出1023个并发存储图像之外的短期图像。如果遇到当前存储图像的图像号码和多图像缓存中的其他图像相同的情况，译码器可以把这种情况作为错误处理。

U.3.1.4 非增强参考图像选择层（NOERPSL）（1比特）

NOERPSL是一个单比特固定长度码字。无论ERPS模式是否使用，NOERPSL都会出现在GOB或者截面级中，但是不应该出现在PLUS头中。NOERPSL的值如下：

"0": 发送ERPS层。

"1": 不发送ERPS层。

如果NOERPSL为1，所有对于某个图像有效的ERPS设置和重映射应该能够应用到GOB或者截面级数据之后的相关视频图像片断。在GOB或者截面上发送的ERPS层信息负责GOB或者截面级数据之前的视频图像片断的译码处理，但是并不影响其他视频图像片断的译码过程（视频图像片断的定义见附件R）。

U.3.1.5 增强参考图像选择层（ERPS）（可变长度）

当ERPS模式使用时，ERPS层会一直出现在图像级中；如果NOERPSL为"0"，ERPS层会出现在GOB或者截面级中。它说明用来译码当前图像或者视频图像片断的缓存索引，并管理图像缓存的内容。

U.3.1.5.1 多参考图像激活（MRPA）（1比特）

MRPA是一个单比特固定长度码字。只有当图像编码类型指示为P图像，EP图像，改进的PB帧或者B图像时，MRPA才会出现。只要MRPA存在，它就是ERPS层的第一个元素。MRPA指定当前图像或者视频图像片断的前向预测或者反向预测译码的活动参考图像的数量可以大于一个。MRPA的值如下：

"1": 可能超过一个参考图像用作前向或者反向运动补偿。

"0": 只有一个参考图像用作前向或者反向运动补偿。在这种情况下，U.3.2节的宏块扩展语法不能使用。

MRPA在由一个视频图像片断到另一个视频图像片断时可以改变，因此不同的视频图像片断可能对应不同的参考图像数量。

在所有激活参考图像再采样模式（见附件P）的图像中，MRPA应该为"0"；同一图像应该被指做所有该类当前图像的图像级和GOB或者截面级的前向参考图像。如果当前图像为B图像，反向参考图像的尺寸应该与当前图像相同，任何参考图像再采样的处理只应该应用到前向参考图像。只要多图像缓存中包含足够的“未使用”能力来储存再采样前向参考图像，参考图像再采样就应该被激活，但是在再采样参考图像用作当前图像译码之后，再采样前向参考图像就不应该继续存储再多图像缓存中。

U.3.1.5.2 图像数量指示器的重映射（RMPNI）（可变长度）

RMPNI是一个可变长度码字。如果图像是P，EP，改进的PB或者B图像时，RMPNI将会出现在ERPS层中。RMPNI指示是否需要为当前图像或者视频图像片断的运动补偿重映射缺省某一图像索引，在有指示的情况下，如何将相关的索引重映射到多缓存中去也需要进行说明。RMPNI使用表U.2进行传输。如果RMPNI显示ADPN或者LPIR存在，一个附加的RMPNI域将会紧随在ADPN或者LPIR域之后。

表 U.2/H.263—参考图像重映射的RMPNI 操作

| 值 | 重映射说明 |
|-------|----------------------------------|
| "1" | ADPN 域存在，相应的操作是在一个图像编号预测值上增加负的差值 |
| "010" | ADPN 域存在，相应的操作是在一个图像编号预测值上增加正的差值 |
| "011" | LPIR 域存在，说明一个参考图像的长效索引 |
| "001" | 结束重映射图像相关索引缺省顺序的循环 |

一个图像参考参数是一个相关索引，可以在已经排序的图像序列中索引。为实现对特殊图像或者视频图像片断的译码，RMPNI, ADPN和 LPIR域允许多图像缓存的相关索引的顺序由缺省索引顺序暂时改变。缺省索引顺序是为了在参考索引顺序中，短期图像（例如，那些没有被分配长效索引的图像）的索引先于长效图像。在短期图像序列中，缺省的编号顺序是从最近缓存的参考图像开始，但是从最早的参考图像开始处理（例如，十比特图像编号域缺少封装时图像编号的降序）。在长效图像序列中，缺省的编号顺序时从最小的长效索引开始，但是从长效索引等于最近的值（MLIP1 - 1）的图像处理。

例如，如果缓存器中包含了三个短期图像和两个长效图像，短期图像编号分别为300, 302, 和303（这些图像编号是以升序传送的），长效图像编号为0和3，缺省的索引顺序为：

- 默认相关索引0指向编号为303的短期图像；
- 默认相关索引1指向编号为302的短期图像；
- 默认相关索引2指向编号为300的短期图像；
- 默认相关索引3指向索引为0的长效图像；
- 默认相关索引4指向索引为3的长效图像；

收到的第一个ADPN或者LPIR域将某一特定图像移出缺省的顺序，相关索引置为0；第二个域将某一特定图像移出缺省的顺序，相关索引置为1。剩余没有移到相关索引顺序的图像序列应该继续保留它们之间的缺省顺序，并且跟随移动到相关索引顺序缓存器前面的图像之后。

如果MRPA为"0"，除非当前图像是B图像，否则不会超过一个ADPN或者LPIR域出现在相同的ERPS层中。如果当前图像为B图像，并且MRPA为"0"，不会超过两个ADPN或者LPIR域出现在相同的ERPS层中。

任何特定图像编号的重映射不应该影响其他图像的译码处理。任何视频图像片断编号的重映射不应该影响其他视频图像片断的译码处理。一个图像的图像编号的重映射对图像内的视频图像片断译码过程的影响应该只有如下两种方式：

- 如果在GOB或者截面级，NOERPSL为"1"，图像级的重映射也可以应用到相关的视频图像片断；
- 如果图像是B图像，在图像级的重映射应该指定用于直接双向预测的 TR_B 和 TR_D 值的计算方法。

如果MRPA为"0"，RMPNI“循环结束”指示是B图像ERPS层的最后一个元素。在B图像中的MRPA等于"1"时，在RMPNI“循环结束”指示的后面为BTPSM。在P或者EP图像或者改进的PB帧中，RMPNI“循环结束”指示的后面为PRBT。

在一个ERPS层中，RMPNI不应该指定任意单独参考图像的位置到相关图像索引中超过一个的重定向位置。

U.3.1.5.3 图像编号的绝对差值（ADPN）（可变长度）

ADPN是一个可变长度码字，RMPNI指示时才出现。ADPN出现时在RMPNI之后。ADPN使用表U.1进行传送，表中的索引相当于ADPN-1。ADPN代表了当前重定向图像的图像编号和图像编号预测值之间绝对差值。如果在当前ERPS层中先前没有ADPN域，预测值应该是当前图像的图像编号。如果已经先前已经有ADPN域，预测值为使用ADPN重映射的最后一个图像的图像编号。

如果图像编号预测指示为PNP，待定的图像编号为PNQ，译码器应该按照如下算法由PNP和ADPN来决定PNQ的值：

```
if (RMPNI == "1") { // 一个负差
    if (PNP - ADPN < 0)
        PNQ = PNP - ADPN + 1024;
    else
        PNQ = PNP - ADPN;
} else { // 一个正差
    if (PNP + ADPN > 1023)
        PNQ = PNP + ADPN - 1024;
    else
        PNQ = PNP + ADPN;
}
```

编码器应该控制RMPNI和ADPN，使ADPN的译码值不大于等于1024。

下述过程为一个操作示例，编码器可以决定ADPN和RMPNI的值来说明待定的重定向编号，PNQ：

```
DELTA = PNQ - PNP;  
if (DELTA < 0) {  
    if (DELTA < -511)  
        MDELTA = DELTA + 1024;  
    else  
        MDELTA = DELTA;  
}else{  
    if (DELTA > 512)  
        MDELTA = DELTA - 1024;  
    else  
        MDELTA = DELTA;  
}
```

```
ADPN = abs(MDELTA);
```

这里abs()是取绝对值的操作。注意表U.1的索引等于ADPN - 1，而不是ADPN。

然后RMPNI将会由MDELTA的符号决定。

U.3.1.5.4 重映射的长效图像索引 (LPIR) (可变长度)

LPIR是一个可变长度码字。在RMPNI指示时出现。LPIR出现时在RMPNI之后。LPIR使用表U.1传送。它代表了需要重映射的长效图像索引。用作后续ADPN重映射预测的值不受LPIR的影响。

U.3.1.5.5 B-图像双图像预测子模式 (BTPSM) (1 比特)

BTPSM是单比特固定长度码字，它只有在MRPAN为"1"时出现在B图像中（见附件O）。它在RMPNI“循环结束”之后，是B图像ERPS层的最后元素。它指示双图像反向预测子模式是否使用：

"0": 单图像反向预测。

"1": 双图像反向预测。

如果BTPSM不出现时，其默认为"0"（当MRPA为"0"时）。

用作前向预测参考的前向预测图像序列是在多图像缓存器中的图像序列而不是反向参考图像。反向参考图像序列由BTPSM的值决定。如果单图像反向预测是由BTPSM规定，那么相关索引顺序中（或者是重映射的）第一个图像是唯一的反向参考图像。如果双图像反向预测是由BTPSM规定，那么相关索引顺序中（或者是重映射的）前两个图像是两个反向参考图像。这样前向预测的相关索引就变成了一个前向参考图像序列的相关索引。

多图像缓存器中的内容不受B图像存在的影响。B图像并不储存在多图像缓存中，不能用作后续图像编译码的参考。

U.3.1.5.6 参考图像缓存类型 (RPBT) (1 比特)

RPBT是一个单比特固定长度码字，用来说明当前译码图像的缓存类型。当图像类型不是I，EI或者B图像时，RPBT出现在RMPNI指示之后。如果图像是I或者EI图像，RPBT是ERPS层的第一个元素。如果图像为B图像，RPBT将不会出现。RPBT的值定义如下：

"1": 滑动窗。

"0": 适用存储器控制。

在“滑动窗”缓存类型中，当前译码图像以索引0加入进缓存器中，缓存器中任何标为“未使用”状态的图像自动按照短期图像序列中先入先出的模式操作。在这种情况下，如果缓存器中有足够“未使用”的容量来存储当前图像，缓存器中应该不会有另外被标“未使用”状态的图像。如果缓存器中没有足够的“未使用”的容量来存储当前图像，缓存器里短期图像中具有最大缺省索引（或者在子图像移除时必须的索引）的图像（或者在子图像移除时必须用来释放需要的存储器容量的图像）应该被标为“未使用”。在“滑动窗”缓存类型中，不会传送附加信息来控制缓存器中的内容。

在“适用存储器控制”缓存类型中，编码器需要明确说明任何增加到缓存器中的内容或者缓存器中标为“未使用”状态的数据，并且还可能需分配长效索引给短期图像。缓存器中的当前图像和其他图像可以明确标为“未使用”状态，这需要由编码器说明。这种缓存类型需要由存储管理控制操作参数控制更进一步的信息。

RPBT如果在GOB或者截面层出现，应该和在图像层出现相同。任何出现在GOB或者截面层的MMCO命令应该和图像层中的某些MMCO命令指示相同的操作。

如果图像是B图像，则不应该出现RPBT，并且译码图像不应该出现在多图像缓存器中。这可以保证B图像不会影响到多图像缓存器中的内容。

类似的，一个改进PB帧的B图像部分不应该存储在缓存器中。与控制一个改进的PB帧存储相关的所有控制域都应该考虑到和改进的PB帧P图像部分的存储控制相关。

U.3.1.5.7 存储管理控制操作（MMCO）（可变长度）

MMCO是一个可变长度码字，只有在RPBT指示“适用存储器控制”时出现，可以出现多次。它说明用来管理多图像缓存器中的控制操作。MMCO参数的后面是由MMCO值规定的操作所需要的数据，然后是一个附件的MMCO参数，直至MMCO值指示结束此类操作。MMCO命令不影响缓存器的内容或者用作当前图像译码的译码过程—更准确的说，它规定比特流中后续图像译码必须的缓存器状态。与MMCO相关的值和控制操作在表U.3中定义。

表 U.3/H.263—存储管理控制操作（MMCO）值

| 值 | 存储管理控制操作 | 后面的相关数据域 |
|---------|----------------|---------------------------|
| "1" | 结束 MMCO 循环 | 无 (ERPS 层结束) |
| "011" | 标识一个短期图像为“未使用” | DPN |
| "0100" | 标识一个长效图像为“未使用” | LPIN |
| "0101" | 分配长效索引给一个图像 | DPN 和 LPIN |
| "00100" | 标识短期子图像域为“未使用” | DPN 和 SPRB |
| "00101" | 标识长期子图像域为“未使用” | LPIN 和 SPRB |
| "00110" | 说明最大长效图像索引 | MLIP1 |
| "00111" | 说明缓存器尺寸和结构 | SPWI, SPHI, SPTN, 和 RESET |

所有使用MMCO规定的存储器管理控制操作应该在图像层中规定。一些或者所有在图像层中规定的操作也可以在GOB或者截面层中规定（使用相同的相关数据）。MMCO不应该在GOB或者截面层中规定存储器操作，也不应该使用图像层中的相同操作来规定。

规定缓存器尺寸和结构的MMCO命令，如果有的话，应该是第一个MMCO命令。在一个给定的ERPS层中，不会有超过一个规定缓存器尺寸和结构的MMCO命令。规定缓存器尺寸和结构的MMCO命令，如果RESET等于“1”，应该出现在比特流中一系列ERPS激活模式图像的第一个图像中。规定缓存器尺寸和结构的MMCO的命令，如果RESET等于“1”，具有优先于任何其它的MMCO命令的权利，标记出在短期或长期图像中的子图像区域为“未使用”。除非当前图像为I或EI图像且RESET等于“1”，规定缓存器尺寸和结构的MMCO命令中携带的子图像宽度和高度参数应与前一个规定缓存器尺寸和结构的MMCO命令中的相同。除非在图像中包含了RESET等于“1”的规定缓存器尺寸和结构的MMCO命令(或图像中不使用ERPS模式)，图像的高度和宽度在比特流中不应当改变。

如果一个使用单图像反向预测的B图像出现在比特流中，严格来说，B图像的参考层中的后续非B图像在比特流顺序中应该领先于B图像，遵守Q.2的规定。由于在B图像译码后，显示时需要参考层图像，无内存管理控制操作应该出现在所有位于B图像参考层中的后续非B图像的ERPS层中，在B图像的参考层标识随后的非B图像的任何部分都是“未使用”。

O.2节中规定的传输顺序限制可以根据使用双图像反向预测的B图像的需要进行调整。如果使用双图像反向预测的B图像出现在比特流中，B图像参考层中的两个后续非B图像在比特流中应该在B图像之后。应该使用O.2节中规定的比特流顺序中传输B图像顺序的其他限制，但是应调整后为两个后续参考层图像使用。由于在B图像译码后，显示时需要这些参考层图像，无内存管理控制操作应该出现在所有位于B图像参考层中的后续两个非B图像的ERPS层中，在B图像的参考层标识随后的两个非B图像的任何部分都是“未使用”。

如果一个非B图像，在它的ERPS层中不包含一个MMCO命令将该图像标为“未使用”，这个图像可以定义为“存储图像”。如果当前图像不是一个存储图像，它的ERPS层不应包含任何MMCO命令的如下类型：

- 用以说明缓存器尺寸和结构，并且RESET等于1的MMCO命令；
- 将所有先前存储图像的EPRS层中为标为“未使用”状态的所有其他图像标为“未使用”的MMCO命令；
- 为先前存储图像的EPRS层中没有分配一个相同长效索引的图像分配一个长效索引的所有MMCO命令；或者
- 任何将所有图像的子图像域标为“未使用”的MMCO命令，这些子图像域在先前存储图像的ERPS层中没有被标为“未使用”。

U.3.1.5.8 图像编号差 (DPN) (可变长度)

DPN根据MMCO的指示出现，出现时DPN在MMCO之后。DPN使用表U.1中的码字进行传输，DPN用来计算用作存储器控制操作的图像的PN值。它用来给图像分配一个长效索引，标识一个短期图像为“未使用”，或者标识短期图像的子图像域为“未使用”。如果当前译码图像数字未PNC，U.1的译码值是DPN，等效于如下等式的数学计算应该被用作PNQ的计算，具体的图像编号待定：

```
if (PNC - DPN < 0)
    PNQ = PNC - DPN + 1024;
else
    PNQ = PNC - DPN;
```

类似的，译码器可以使用如下关系式计算DPN值来进行译码：

```
if (PNC - PNQ < 0)
    DPN = PNC - PNQ + 1024;
else
    DPN = PNC - PNQ;
```

例如，如果DPN的译码值为0，MMCO指示标识一个短期图像为“未使用”，当前译码图像应该被标识为“未使用”。

U.3.1.5.9 长效图像索引 (LPIN) (可变长度)

LPIN受MMCO的指示出现。LPIN使用表U.1的码字传输，用来图像的规定长效图像索引。如果是分配一个长效索引给一个图像的操作，LPIN在DPN之后。如果是标识一个长效图像为“未使用”或者标识一个长效图像的子图像域为“未使用”的操作，LPIN在MMCO之后。

U.3.1.5.10 子图像移除比特图 (SPRB) (固定长度)

SPRB是一个固定长度码字，为每一个子图像包含1比特，受MMCO的指示出现。SPRB数据的比特数量由SPWI和SPHI最近的值决定。SPRB用来指示缓存图像的哪一个子图像域被标识为“未使用”。如果是标识短期图像的子图像域为“未使用”状态的操作，SPRB在DPN之后；如果是标识长效图像的子图像域为“未使用”状态的操作，SPRB在LPIN之后。

子图像在光栅扫描顺序中被编号，由图像的左上角开始。例如，考虑如下情况：一个由DPN规定的参考图像被分为六个子图像。由" $s_1 s_2 s_3 s_4 s_5 s_6$ "代表SPRB的六个比特。如果比特 s_i 为"1"，译码器应该标识相应的子图像为“未使用”。例如，如果SPRB是"000110"，那么第四个和第五个子图像域应该被标识为“未使用”。

为防止起始码仿真，所有的SPREPFB仿真防止比特应该按照U.3.1.5.11.的规定插入到SPRB数据中或者跟随在SPRB数据之后。

如果SPRB存在，并且指定的图像已经受到前面的SPRB比特图的影响，对那些在前面的SPRB比特图中包含了“1”的子图像域，由SPRB规定的比特图都应该包含“1”。所有的SPRB比特图应该包含至少一个值为“0”的比特和至少一个值为“1”的比特。

U.3.1.5.11 子图像移除仿真防止比特 (SPREPFB) (1比特)

SPREPFB是一个单比特固定长度码字，它应该被插入到一个连续8个比特都为0的SPRB数据字符串之后。

U.3.1.5.12 最大长效图像索引加1 (MLIP1) (可变长度)

MLIP1是一个可变长度码字，受MMCO的指示出现。MLIP1出现在MMCO之后。MLIP1使用表U.1进行传输。MLIP1出现时用来决定长效参考图像允许的最大索引（直至收到另一个MLIP1值）。译码器应该置MLIP1的初始值为“0”，直至接收到其他的值。根据接收到的MLIP1参数，MLIP1应该将所有索引大于MLIP1 - 1的长效图像置为“未使用”状态，用作后续图像的译码处理的参考。对于多图像缓存器中的所有其他图像，按照MLIP1的指示应该不做改变。

U.3.1.5.13 子图像宽度指示 (SPWI) (7比特)

SPWI是一个固定长度7比特码字，按照MMCO的指示出现。SPWI出现时跟随在MMCO之后。SPWI以16亮度采样为单位规定子图像的宽度，这样指定的子图像宽度为 $16 \cdot (\text{SPWI} + 1)$ 亮度采样。当前图像宽度以 $\lceil \text{ceil}(\text{pw}/16) / (\text{SPWI} + 1) \rceil$ 当前图像]为单位，这里pw是图像宽度，“/”代表浮点除法运算。对于正数，如果x是正式取上限函数 $\text{ceil}(x)$ 等于x，否则 $\text{ceil}(x)$ 等于x的整数部分加1。如果一个定义了子图像最小宽度和高度的最小图像单元 (MPU) 的尺寸已经通过外部手段进行了协商（例如，ITU-T H.245建议书），由SPWI定义的子图像宽度应该是MPU宽度的整数倍；否则，由SPWI定义的子图像宽度应该是 $\text{SPWI} = \text{ceil}(\text{pw}/16) - 1$ 。

U.3.1.5.14 子图像高度指示 (SPHI) (7比特)

SPHI是一个固定长度7比特码字，在SPWI出现时出现（按照MMCO的指示）。SPHI出现时在SPWI之后。SPHI以16亮度采样为单位规定子图像的高度，这样指定的子图像宽度为 $16 \cdot \text{SPHI}$ 。SPHI允许值的范围是从1到72。当前图像的高度由 $\text{ceil}(\text{ceil}(\text{ph}/16) / \text{SPHI})$ 子图像为单位，这里ph是图像高度，“/”代表浮点除法运算。如果一个定义了子图像最小宽度和高度的最小图像单元 (MPU) 的尺寸已经通过外部手段进行了协商（例如，ITU-T H.245建议书），由SPHI定义的子图像高度应该是MPU宽度的整数倍；否则，由SPWI定义的子图像宽度应该是 $\text{SPHI} = \text{ceil}(\text{ph}/16)$ 。

U.3.1.5.15 子图像总数 (可变长度)

SPTN是一个可变长度码字，在SPWI 和SPHI出现时出现（按照MMCO的指示）。出现时SPTN在SPHI之后。SPTN的编码使用表U.1，表U.1的索引对应于SPTN-1的译码值。SPTN的译码值为以特定SPWI和SPHI的子图像为单位的可操作的多图像缓存的总长度。为当前图像译码所需的内存的容量并未包含在SPTN中，仅仅是存储用于预测其他图像的参考图像的内存容量。如果不使用子图像移除（例如，当SPWI和SPHI是整个图像的尺寸），则活动的短期参口图像的最大数量（例如用于滑动窗口操作）为SPTN减去被分配到长期索引中并且还没有被标记为“未使用”的图像数。

U.3.1.5.16 缓存复位指示 (RESET) (1比特)

RESET是一个单比特固定长度码字，当SPWI，SPHI，SPTN出现时出现。如果出现，RESET紧随在SPTN之后。RESET的值如下所示：

“0”： 缓存内容不复位。

“1”： 缓存内容被复位。

如果RESET为“1”，所有在多图像缓存中的图像（但不包括当前图像，除非单独指定）应被标记为“未使用”（包括短期和长期图像）。

U.3.2 宏块层句法

U.3.2.1 P-图像和改进的PB帧宏块句法

如果ERPS层为P图像和改进的PB帧存在，当的选择的前向参考图像如MRPA指示可能大于一个时，则宏块层句法被改变。ERPS层的MRPA域被赋值。当MRPA为“1”时，宏块层句法如图U.6所示。否则，P图像和改进的PB图像的宏块层句法如图10所示，不做改变。



图 U.6/H.263 – ERPS模式的P图像和改进的PB帧宏块层结构

U.3.2.1.1 COD的解释

如果COD比特为“1”，则对于宏块没有进一步的信息发送。在这种情况下，译码器应把宏块作为运动向量为0，图像参考参数为0，并且没有系数数据的INTER宏块对待。如果COD比特为“0”，则指示宏块的编码句法如图U.6所描述，包括了PR₀, PR, PR₂, PR₃, PR₄和PRB在句法中。PR₀, PR, PR₂, PR₃, PR₄和PRB每个都包括了一个可变长的码字，码字如表U.1所示。

U.3.2.1.2 图像参考参数0 (PR₀) (可变长度)

如表U.1所示，PR₀是一个可变长的码字。当COD为“0”时其出现。如果PR₀的译码值为0（码字“1”），则指示宏块将有后续信息跟随。如果为非零，则指示宏块的编码仅使用一个图像参考参数。

如果PR₀域译码值为非零，并且没有向宏块发送后续的信息，这种情况下译码器应把宏块作为运动向量为0，图像参考参数为PR₀，并且没有系数数据的INTER宏块对待。

如果MCBPC, CBPB, CBPY和DQUANT域的译码值为0（码字“1”），宏块为已编码的。域MCBPC, CBPB, CBPY和DQUANT的含义和使用仍不改变。对于所有的INTER宏块（和在改进的PB帧模式下的INTRA宏块），域PR和域MVD是一起提供的。在改进的PB帧下的MODB的使用的描述见U.3.2.1.4。

U.3.2.1.3 宏块竞争防止比特0 (MEPB₀) (1比特)

MEPB₀是一个单比特固定长度码字。当且仅当PR₀出现并译码值为“1”（码字“000”）时，MEPB₀跟在PR₀后面并且值为“1”，否则下面两个条件应被满足：

- 1) 使用截面结构模式(见附件 K)；或
- 2) 当前宏块和其紧临的前一个宏块都有COD=“0”，并且PR₀=1(码字“000”)并且前一个宏块没有MEPB₀比特。

MEPB₀的目的是防止起始码竞争以及在截面结构模式下帮助判断截面中的宏块的数量。

U.3.2.1.4 宏块图像参考参数 (PR, PR₂₋₄, and PRB) (可变长度)

PR是一个初始图像参考参数，只要MCD出现PR就会出现。如果PTYPE出现，并且如果MCBPC规定了一个INTER4V或INTER4V+Q宏块（在表8和9中的宏块类型2或5），则三码字PR₂₋₄与MVD₂₋₄一起出现。仅当处于高级预测模式（见附件 F）或去除分块效应滤波器模式（见附件 J）时，PR₂₋₄和MVD₂₋₄才出现。

仅当在改进PB帧中当MODB指示MVDB出现时，PRB才出现。PR, PR₂₋₄, 和PRB每个都规定了一个指向多图像缓存中的参考图像的索引。如果宏块不是INTER4V或INTER4V+Q宏块，则PR用于作为整个宏块的运动补偿的图像参考参数。如果宏块是INTER4V或INTER4V+Q宏块，PR用于作为宏块中第一个8×8亮度块和两个色度块的运动补偿预测。PR₂₋₄用于宏块中剩下的三个8×8块的亮度数据的运动补偿。如果MODB指示MVDB出现，则PRB作为改进的PB帧的B部分的前向预测的图像参考参数。

在改进的PB帧中，当MODB指示B_{PB}双向预测，基于当前图像以及最近前面的参考图像（无论这些图像已经被映射到不同的相对索引顺序标记为“未使用”还是已分配到长期索引中）的时间参考数据，TR_D和TR_B应被计算作为时间参考增量。在改进的PB帧中用于双向预测的前向参考图像应是PR指定的图像。

U.3.2.1.5 宏块竞争防止比特 (MEPB, MEPB₂₋₄ 和MEPBB) (每个1比特)

如果出现的话，MEPB, MEPB₂₋₄ 和MEPBB每一个都是单比特并且值为“1”。当且仅当非受限运动矢量模式（见附件 D）未使用并且关联的PR, PR₂₋₄, 或PRB域出现且译码值为1（码字“000”）时，这些域才出现。它们的目的是防止起始码竞争。

U.3.2.2 B图像和EP图像宏块句法

B和EP图像的宏块层句法（见附件O）与P图像以类似的模式修改。COD比特如果等于1，指示有附件O定义的跳过的宏块，使用图像参考参数0作为EP图像中的前向（跳过的）预测以及B图像中直接（跳过的）双向预测的前向预测部分，并且使用第一个反向预测图像作为B图像中直接（跳过的）双向预测的后续预测部分（在双图像反向预测的情况中，当BSBBW出现并且为“0”时）。如果COD为“0”，PR₀参数应该插入到句法中，用法和U.3.2.1.2节描述的类似。如果PR₀出现，但译码值并不包含0（码字“1”），它指示宏块将会以前向INTER预测来进行预测，使用一个0值运动矢量和一个图像参考参数PR₀。如果PR₀译码值中有0，MBTYPE跟随其后来说明宏块类型。CBPC, CBPY和DQUANT域的格式不发生改变。当不使用ERPS时，MVDFW和MVDBW域以相同方式编码，但每一个都绑定一个图像参考，并且可能是一个仿真防止比特。

对于B图像，多图像缓存中的反向参考图像定义如下：

- 在单图像反向预测的情况中，仅有一个反向参考图像，它是相关索引顺序中（也可能是重映射）的第一个图像；
- 在双图像反向预测的情况中，有两个反向预测图像，它是相关索引顺序中（也可能是重映射）的前两图像。

多图像缓存中的前向参考图像定义为多图像缓存中的图像而不是反向参考图像。前向预测图像的相关索引可以索引到前向参考图像序列，后向预测图像的相关索引可以索引到后向参考图像序列。

例如，如果缓存中有三个短期图像和两个长效图像，短期图像编号为300、302和303（它们以图像编号的升序进行传输），长效图像索引为0和3。在双图像反向预测中，缺省的索引顺序是：

- 缺省反向索引0指向编号为303的短期图像；
- 缺省反向索引1指向编号为302的短期图像；
- 缺省前向索引0指向编号为300的短期图像；
- 缺省前向索引1指向索引为0的长效图像；
- 缺省前向索引2指向索引为3的长效图像；

在单图像反向预测的情况中，缺省的索引顺序是：

- 单缺省反向参考图像是编号为303的短期图像；
- 缺省相关索引0指向编号为302的短期图像；
- 缺省相关索引1指向编号为300的短期图像；
- 缺省相关索引2指向索引为0的长效图像；
- 缺省相关索引3指向索引为3的长效图像；

并且如果这些图像已经重映射到一个新的相关索引顺序中：首先是短期图像302，后面是短期图像303，后面是长效图像0，后面是短期图像300，后面长效3。在双图像反向预测的情况中，新的相关索引顺序是：

- 重映射反向索引0指向编号为302的短期图像；
- 重映射反向索引1指向编号为303的短期图像；

- 重映射前向索引0指向索引为0的长效图像；
- 重映射前向索引1指向编号为300的短期图像；
- 重映射前向索引2指向索引为3的长效图像；

在单图像反向预测的情况下，新的索引顺序是：

- 单重映射反向参考图像是编号为302的短期图像；
- 重映射相关索引0指向编号为303的短期图像；
- 重映射相关索引1指向索引为0的长效图像；
- 重映射相关索引2指向编号为300的短期图像；
- 重映射相关索引3指向索引为3的长效图像；

B图像中用作直接双向预测的 TR_D 应该作为相关索引顺序中（也可能是重映射的）第一个前向参考图像和第一个反向参考图像之间的时域参考增量被计算（例如，如果使用双图像反向预测，按照U.3.2.2.3的描述这应该是在BSBBW为“0”时的参考图像）。B图像中用作直接双向预测的 TR_B 应该作为B图像和相关索引顺序中（也可能是重映射的）第一个前向参考图像之间的时域参考增量被计算。在计算 TR_D 和 TR_B 使用的相关索引顺序应该是由B图像句法中图像级别中的ERPS层来规定（例如，在GOB或者截面级别的重映射不应该影响 TR_D 和 TR_B 的值）。（见图U.7）

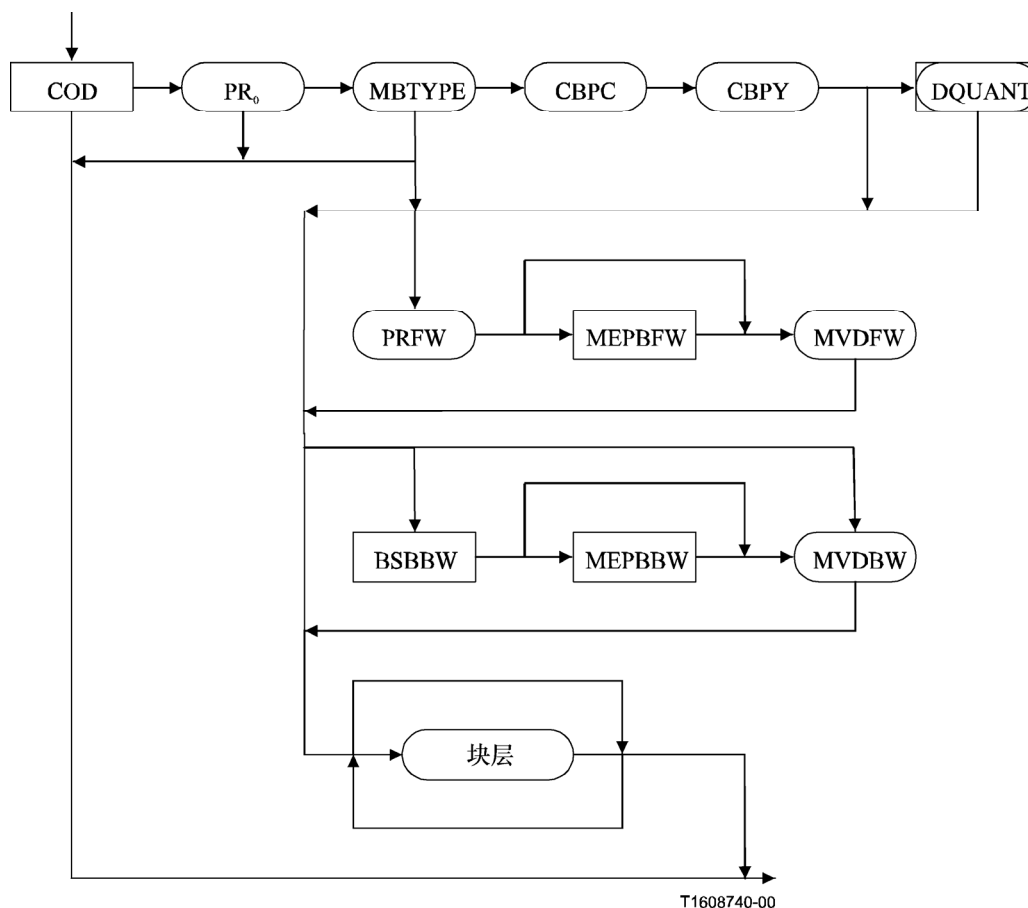


图 U.7/H.263—ERPS模式的EP-和B-图像宏块层结构

U.3.2.2.1 前向预测的图像参考（PRFW）（可变量）

PRFW是可变量的图像参考参数，使用表U.1编码。当前向运动向量数据出现时出现。PRFW指向前向参考图像的一个相对索引。

U.3.2.2.2 前向预测的仿真防止比特（MEPBFW）（1比特）

MEPBFW是一个值为“1”的单比特定长码字，当且仅当PRFW出现且译码值为“1”（码字“000”）且非受限运动矢量模式（见附件D）未使用时出现。

U.3.2.2.3 后向预测的B-图像选择比特（BSBBW）（1比特）

BSBBW是一个单比特定长码字。当MVDBW出现且仅当双图像后向预测被指定为B-图像操作时，BSBBW出现。这个比特的含义定义为：

- “0”：从相对索引顺序的第一个后向参考图像进行预测（在缺省的顺序下，如果图像没有被分配到长期索引或标记为“未使用”，这应是最近的短期参考图像）。
- “1”：从相对索引顺序的第二个后向参考图像进行预测（在缺省的顺序下，如果图像没有被分配到长期索引或标记为“未使用”，这应是第二最近的短期参考图像）。

U.3.2.2.4 后向预测的仿真防止比特 (MEPBBW) (1比特)

MEPBBW是值为“1”的单比特定长码字，它仅在下列条件下出现：

- BSBW出现且等于“0”，并且
- 未使用非受限运动矢量模式；并且BSBW的五个前导比特的值为“00000”。

U.4 译码器处理

在多图像缓存器中用作中间图像译码的参考图像由ERPS模式的译码器存储。译码器可能需要额外的存储器容量来储存多个已译码的图像（与不支持ERPS模式的存储器容量有关）。译码器根据参考图像缓存类型和比特流中说明的所有存储器管理控制操作来复制编码器的多图像缓存器。当部分错误图像被译码时，缓存机制也可以运行。

每一个传输和存储图像都可以被分配一个图像号码 (PN)。在多图像缓存中，该号码和图像存储在一起。PN代表了存储图像的一个连续图像计数标识。PN的长度受限，计算方法是使用1024取模。对于第一个传输图像，PN应该为“0”。对于每一个和所有的其他传输和存储图像，PN应该按1递增（在一个给定的可测量层中，如果附件O正在使用）。如果收到和存储的两个连续图像的差值取1024模后不为1，译码器应该认为有图像丢失或者数据中断。在这种情况下，一个指示图像丢失的反向信道信息可能会发送给编码器。

除了PN以外，每一个存储在多图像缓存中的图像都有一个相关索引，称为默认索引。第一个加入到多图像缓存器中的图像，其默认索引为0—除非它以及被分配了一个长效索引。当有图像加入或者移出多图像缓存器时，多图像缓存器中的图像索引将会被修改。

多图像缓存中存储的图像可以分为两类：长效图像和短效图像。一个长效图像可以在多图像缓存中停留较长的时间（超过1023个编码和存储的图像间隔）。当前图像最初会被当作一个短期图像。根据比特流中提供的信息，任意短期图像都可以通过分配一个长效索引变成长效图像。对于多台缓存中的所有短期图像而言，PN是唯一的ID。当一个短期图像变成一个长效图像，它同时会被分配一个长效图像索引 (LPIN)。通过关联图像的PN值到一个LPIN值，可以为图像分配一个长效图像索引。一旦一个长效图像索引分配给一个图像，长效图像的PN值唯一可能的后续使用应该是在长效索引分配的循环中。长效图像的PN值在1024个传输和存储的图像中是唯一的。这样，在1023个已经传输的连续存储图像后，长效图像的PN也不能用作长效索引的分配。LPIN在一个长效图像的生存期中成为唯一的ID。

PN（短期图像）或者LPIN（长效图像）可以用来重定向图像到重定向索引，以便有效的进行参考图像寻址。

U.4.1 长/短效图像管理的译码器处理

译码器的缓存中可能同时会有长效图像和短期图像。MLIP1域用来指示缓存器中允许的最大长效图像索引。如果先前没有MLIP1值发送，应该不会有长效图像在使用。例如，MLIP1最开始在ERPS模式启动时应该默认为“0”。当收到一个MLIP1参数时，一个新的MLIP1将会生效，直到收到另一个MLIP1为止。根据收到的比特流中的一个新MLIP1参数，所有的长效索引大于或等于MLIP1的长效图像将会被标为“未使用”。传输MLIP1的频率不在本建议书的范围之内。然而，编码器应该在收到一个错误信息时发送一个MLIP1参数，例如一个INTRA请求消息。

通过使用一个带有DPN和LPIN的MMCO命令可以将一个短期图像变为长效图像。短期图像号码从DPN得到，长效图像索引则是LPIN。当收到这样一个MMCO命令时，译码器应该将由DPN指示的具有PN的短期图像变为一个长效图像，并且为该图像分配由LPIN指示的长效图像索引。如果一个具有相同长效索引的长效图像已经存在于缓存器中，这个已经存在的图像将会被标为“未使用”。一个编码器不应该分配大于MLIP1 - 1的长效图像索引给任何图像。如果LPIN大于MLIP1 - 1，译码器应该认为这种情况是一个错误。对于错误恢复，编码器可能重复发送相同的长效索引分配操作或者MLIP1说明信息。如果在一个长效分配操作中说明的图像已经和请求的LPIN相关，译码器不会采取任何行动。译码器不能为相同图像分配超过一个长效索引值。如果在一个长效分配操作中说明的图像已经和其他的长效索引值相关，这种情况应该被看作是一个错误。一个编码器改变一个短期图像为一个长效图像，应该只限于1024个已经传输的联系存储的图像。换言之，在超过1023个后续存储图像已经传输完成之后，一个短期图像就不应该继续停留在短期缓存中。如果一个短期图像已经通过比特流中先于第一个此类分配信息的译码处理标为“未使用”状态的，译码器不应该分配长效索引给该短期图像。编码器不应该分配一个长效索引给一个还未发送的图像号码。

U.4.2 参考图像缓存映射的译码处理

当使用域PR₀, PR, PR₂, PR₃, PR₄, PRB, PRFW和BSBBW参考一个图像用作宏块层上的运动补偿时，译码器使用索引。在图像而不是B图像中，当域ADPN 和LPIR不存在于可用的当前图像，GOB，或者截面层时，这些索引是多图像缓存中的缺省相关图像索引，当这些域存在时，这些索引是重定向的索引。在B图像中，第一个或者前两个相关索引顺序中的图像（依赖于BTSPM）用作反向预测，前向图像参考参数规定一个相关索引到剩余的图像上，用作前向预测。

多图像缓存中的图像索引可以通过传输RMPNI, ADPN和LPIR域被重映射为新的指定索引上，RMPNI指示ADPN 或者LPIR是否存在。如果ADPN存在，RMPNI说明将要加到图像号码预测值上的差值的符号。ADPN则是将要被重新映射的图像的PN和该PN的预测值之间的绝对差值。下一个传输的ADPN域代表使用ADPN重映射的先前图像的PN与其他重映射图像的PN值的差值。这个过程一直持续到所有必须的重映射过程结束。使用LPIR规定的重映射的存在并不影响后续使用ADPN重映射的预测值。如果RMPNI指示LPIR域存在，重映射图像就是一个LPIR作为长效索引的长效图像。如果没有图像通过RMPNI重映射到一个特定顺序，这些剩余的图像应该跟随在索引配置中具有重映射顺序的所有图像之后，跟随在非重映射图像的缺省顺序之后。

如果译码器检测到一个丢失的图像，它可能会激活某些隐藏处理，并且可能插入一个隐藏了错误的图像到多图像缓存中，如果一个或者几个图像号码丢失或者如果在已传送的ADPN或者LPIR中指示的图像没有存储在多图像缓存中，丢失的图像能够被识别。隐藏可能通过复制最近存储在多图像缓存中的先前图像到丢失图像的位置上完成。多图像缓存中短期图像的临时顺序能够有他们的默认索引顺序PN域推断出来。作为附加或者替代方案，译码器可以通过带外方式发送一个强制的INTRA更新信号到编码器（例如，ITU-T H.245建议书），或者译码器可以使用带外方式或者反向信道消息（例如，ITU-T H.245建议书）指示编码器图像丢失。当使用“滑动窗”缓存类型时，隐藏图像可以插入到多图像缓存中。如果当译码GOB或者截面时发现图像丢失，隐藏的应用和在图像层检测到丢失图像时相同。

U.4.3 子图像移除的译码处理

子图像移除可能会用来减少保存多参考图像的内存数量。在子图像移除中，每一个参考图像被分为更小的尺寸相同的子图像。内存的减少通过标识不需要的子图像为“未使用”状态来完成。编码器决定哪个子图像应该被标为“未使用”的策略不在本建议书的研究范围之内。编码器使用ERPS层的MMCO命令来通知译码器子图像的尺寸和那个子图像被标为“未使用”。编码器不能在比特流中发送任何会引起参考图像或者子图像抽样的信息，这样会导致标为“未使用”的图像被用做后续图像的预测。

子图像移除的能力协商通过外部手段进行（例如，ITU-T H.245建议书）。另外，译码器也通过外部手段发送由最小宽度和高度（以16个亮度抽样为单位）描述的子图像最小分割单元（MPU）以及可用的多图像缓存的数量。内存管理使用如下所述的分割原则。

每个参考图像被分割为同等大小的矩形子图像。编码器规定子图像的尺寸必须是MP的整数倍。子图像的宽度和高度应该是同MPU一样通过外部协商后得到的最小宽度和高度的整数倍。第一子图像的左上角和参考图像的左上角相同。这样，整个的分割可以通过规定子图像的宽度和高度来进行描述。如果图像尺寸不是子图像尺寸的整数倍某些子图像可以超出参考图像左边和底部的边界。当超出参考图像边界的子图像被保存时，约定的内存管理策略是设置足够内存来保存整个子图像，而不是仅保存位于子图像的参考图像的一部分。决定内存是否为满的任何计算缓存器剩余容量的方法都应遵守此惯例（例如，在“滑动窗”操作中为了判定是否将缓存图像标为“未使用”）。译码器设计成这种每个子图像占据相同数量内存的方式可以防止内存碎片的产生。

下面简述了一个使用子图像移除时，用来访问参考图像采样的方法。辨别每个子图像采样在内存中存储位置的机制是参考图像访问技术中的一个重要元素。如果有R个参考图像，并且每个参考图像都被分成了S个子图像，就会得到总共 $K = R \cdot S$ 个子图像。例如，位于第一个参考图像号码左上角的子图像作为子图像0，那么它右边的子图像就作为子图像1，依此类推按照光栅扫描顺序从参考图像1到R进行直至所有K个子图像都有标识。缓存总容量是SPTN子图像存储缓存，SPTN通常都小于K。一个K元素阵列可以定义为subPicMem[K]，这样 $t = \text{subPicMem}[k]$ 就对应于包含了第K个子图像采样的子图像存储域。例如如下情况中， $R=5$ ，每个参考图像有 $S = 12$ 个子图像，这样在参考图像3中的第六个子图像采样就可以在子图像存储域 $t = \text{subPicMem}[k]$ 得到，这里 $k = 3 \cdot S + 6 = 42$ 。

例如，当参考一个用作运动补偿的亮度或者色度数据块的采样时，同时高级预测和简约分辨率更新可选模式没有启用，则必须获的 $n \times m$ 个采样，这里 n 和 m 取值8或者9，以使用半整数运动补偿。由于一个块中的采样可能有赖于四个不同子图像，就必须考虑四种不同情况。在所有的情况中，第一步需要寻找到包含参考块左上角采样（U）的子图像在存储器中的位置。通过子图像宽度或者高度来区分U的水平或者垂直位置，可以辨别出包含U的子图像。如果U存在于子图像 k ，那么采样就会位于 $\text{subPicMem}[k]$ 子图像存储器区域。下一步，如果采样至U右侧的采样 $m-1$ （例如，块的右上角）和采样到U下方的 $n-1$ 采样（例如，块的左下方）都位于子图像 k ，这种情况作为情况1。如果采样到U下方的 $n-1$ 采样位于子图像 k ，但采样至U右侧的采样 $m-1$ 没有位于子图像 k ，这种情况作为情况2。如果采样至U右侧的采样 $m-1$ 位于子图像 k ，但采样到U下方的 $n-1$ 采样没有位于子图像 k ，这种情况作为情况3。否则，如果采样至U右侧的采样 $m-1$ 和采样到U下方的 $n-1$ 采样都位于子图像 k 之外，这种情况作为情况4。

在情况1中，参考块中的所有采样都位于第 k 个子图像。在这种情况下，所有相关的 $n \times m$ 个采样都可以在子图像存储器域 $\text{subPicMem}[k]$ 中找到，访问这些图像则比较简单。在情况2中，位于第 k 个子图像的采样都可以在子图像存储器域 $\text{subPicMem}[k]$ 中找到，其余的都可以在子图像存储器域 $\text{subPicMem}[k_r]$ 中找到，这里 k_r 是 k 图像右侧的子图像。在情况3中，位于第 k 个子图像的采样都可以在子图像存储器域 $\text{subPicMem}[k]$ 中找到，其余的都可以在子图像存储器域 $\text{subPicMem}[k_d]$ 中找到，这里 k_d 是 k 图像下面的子图像。在情况4中，位于第 k 个子图像的采样都可以在子图像存储器域 $\text{subPicMem}[k]$ 中找到，其余的都可以在子图像存储器域 $\text{subPicMem}[k_d]$ 、 $\text{subPicMem}[k_r]$ 和 $\text{subPicMem}[k_{rd}]$ 中找到，这里 k_r 和 k_d 如上定义， k_{rd} 是 k 下面和右边的子图像。

U.4.4 多图像运动补偿的译码处理

如果MRPA指示使用了超过一个参考图像，就要应用多图像运动补偿。对于多图像运动补偿，译码器选择参考图像遵守宏块层域 PR_0 , PR , PR_2 , PR_3 , PR_4 , PRB , PRFW 和 BSBBW 的指示。一旦参考图像被选定，运动补偿的译码处理过程遵守6.1节的规定。

在每个宏块的四个矢量都被使用并且MRPA域指示超过一个参考图像被使用的情况下，两个色度块的图像参考索引与四个运动矢量的第一个矢量相关（其他的运动补偿处理如6.1节所示）。

U.4.5 参考图像缓存的译码处理

当前译码图像的缓存策略可以使用非B图像的参考图像缓存类型(RPBT)来说明。缓存策略可以是先进先出(“滑动窗”)模式。作为可选方式,缓存策略也可以是由前向信道译码器规定的定制适应缓存策略(“适应存储器控制”)模式。B图像不影响缓存内容。

“滑动窗”缓存类型按照如下方式进行操作:首先,译码器决定图像是否能保存到“未使用”的缓存器容量中。如果没有足够的“未使用”缓存器容量,缺省索引最大的(例如,缓存器中时间最长的图像)短期图像应该标为“未使用”。如果需要,这个处理过程可以一直重复直到有足够的缓存器容量来保存当前译码图像。当前图像存储在缓存器中,并分配一个缺省相关缓存索引0。所有其他短期图像的缺省索引加1。所有长效图像的缺省相关索引加1减去移除短期图像的数量。

在“适应存储器控制”缓存类型中,特定图像或者子图像域可以从多图像缓存中移出。当前译码图像,最开始时当作短期图像,可以插入到缓存器中,缺省相关索引为0;可以分配一个长效索引;或者可以由编码器标为“未使用”。其他短期图像可以分配一个长效索引。缓存处理可以按照和下面功能等同的方式来进行:首先,当前图像加入到多图像缓存中,缺省相关索引为0,并且所有其他图像的缺省相关索引逐一增加。然后,MMCO命令按如下方式处理:

- 如果MMCO通过使用RESET等于1来指示缓冲器内容重置,除当前图像(按照U.3.1.5.7节的规定,缓存器重置命令必须是第一个MMCO命令,缺省相关索引为0的图像为当前图像)外,缓存中的所有图像都标为“未使用”。
- 如果MMCO命令使用MLIP1来指示一个最大长效索引,所有长效索引大于或者等于MLIP1的长效图像都将标为“未使用”,剩余图像的缺省相关索引顺序不受影响。
- 如果MMCO指示多图像缓存中的一个图像标为“未使用”,并且该图像没有已经被标为“未使用”,该指定图像将会标为“未使用”,缺省顺序中该图像之后的所有图像的缺省相关索引都减1。
- 如果MMCO指示多图像缓存中某个图像的子图像域标为“未使用”,指定子图像域将标为“未使用”,但图像的缺省相关索引顺序不受影响。按照U.3.1.5.10节的要求,并不是任意给定图像的所有子图像域都能够由子图像移除MMCO命令标为“未使用”(相反的,译码器将会发送一个MMCO命令将整个图像标为“未使用”)。
- 如果MMCO命令指示分配一个长效索引给一个指定短期图像,并且该指定长效索引还没有被分配给指定短期图像,指定短期图像在缓存中将会被标识为一个长效图像,并具有指定的长效索引。如果缓存器中已经有其他图像具有和该指定长效索引相同的长效索引,其余的图像标为“未使用”。所有缺省索引顺序中位于该指定短期图像后的短期图像索引都减1,所有小于指定长效索引的长效图像的长效索引也都减1。指定图像分配一个缺省相关索引,为减少后的缺省相关索引的最高值加1;如果没有减少后的索引,指定图像分配一个缺省相关索引为0。

图像或者子图像区域中没有标为“未使用”的在缓存器中的最终数量不应该超过最新的SPTN值指示的缓存容量。如果译码器检测到此种情况，应该作为出错处理。

U.5 反向信道信息

用来传送反向信道信息的外部信道，可以是不可靠信道。该外部信道（可以是一个分开的逻辑信道，例如使用ITU-T H.223建议书或者ITU-T H.225.0建议书）的语法这里定义的其中之一。在ERPS模式中不支持附件N中定义的反向信道信息的"videomux"（图像混合）操作。

U.5.1 BCM分离逻辑信道层

U.5.2节定义的BCM层应如图U.8所示的BCM分离逻辑通道承载。

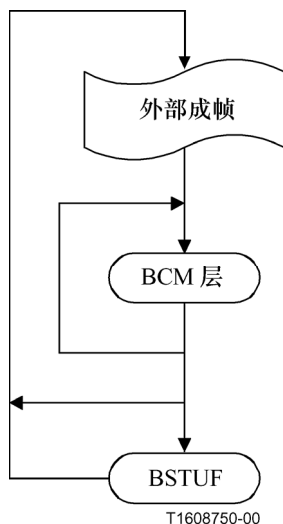


图 U.8/H.263—ERPS模式的BCM分离逻辑信道层结构

U.5.1.1 外部成帧

如图U.8所示，BCM的外部成帧功能应被提供。外部成帧功能用来决定BCM的起始点和后续的BCM数据的数量。

U.5.1.2 反向信道填充（BSTUF）（可变长）

BSTUF是一个变长的码字，其在外部帧中的最后一个BCM之后有可能出现。可变长的BSTUF由一个或更多的比特“0”组成。

U.5.2 反向信道消息层语法

反向信道消息（BCM）层的语法定义如图U.9所示。

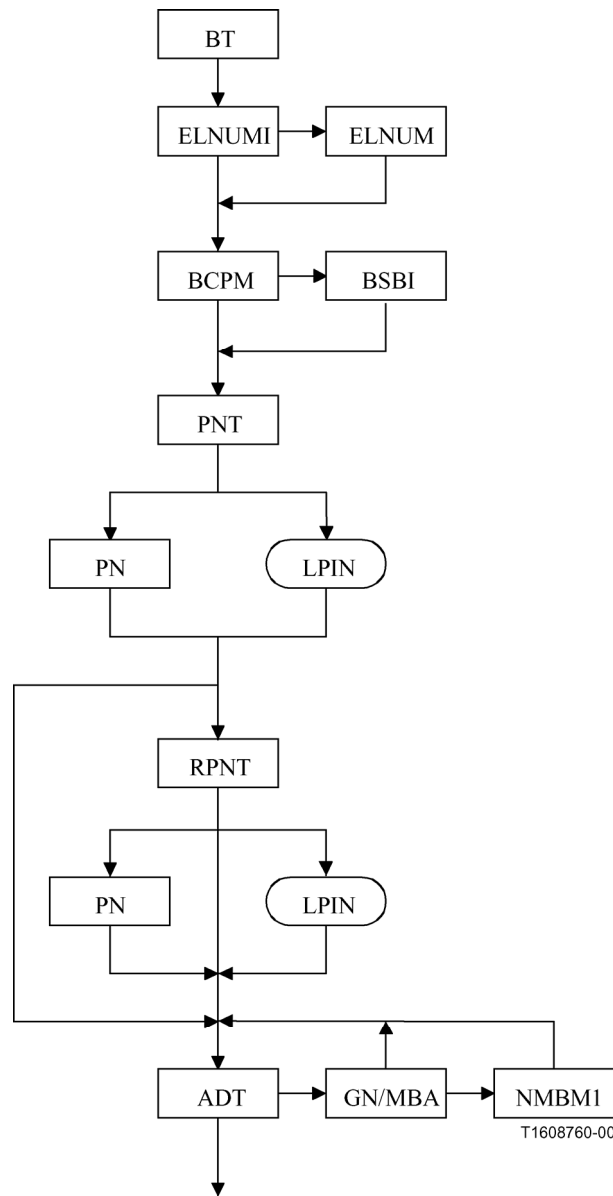


图 U.9/H.263—ERPS模式的反向信道消息（BCM）层结构

U.5.2.1 反向信道消息类型（BT）（2比特）

BT为两比特固定长度的码字，指示反向信道消息的类型。BT是每个反向信道消息的最开始的码字。编码器请求的是何种消息类型在前向信道语法的RPSMF域中指示。BT的值定义为：

- "00": 保留。
- "01": 保留。
- "10": NACK. 指示相应部分的前向信道数据丢失或译码错误。
- "11": ACK. 指示相应部分的前向信道正确译码。

U.5.2.2 增强层数字指示 (ELNUMI) (1比特)

ELNUMI是一个单比特固定长度码字，在反向信道消息中跟随在BT之后。ELNUMI通常应该为0，以下情况除外：时间的、SNR及空间的可量测性模式（见附件O）在前向信道中使用；前向信道的某些增强层被合并成为一个逻辑信道；反向信道消息涉及特定增强层（而不是基本层），这种情况下ELNUMI为“1”。

U.5.2.3 增强层编号(ELNUM) (4 比特)

ELNUM是一个4比特固定长度码字，当且仅当ELNUMI为“1”时，增强层编号才存在。ELNUM出现时在ELUMI之后。如果ELNUM存在，它包含反向信道消息中所涉及的该增强层的层编号。

U.5.2.4 反向信道CPM指示 (BCPM) (1比特)

BCPM是一个单比特固定长度码字，在反向信道消息中跟随在ELNUMI 或 ELNUM之后。BCPM为“0”，除非在前向信道数据中使用CPM模式，在这种情况下中BCP为“1”，若BCPM为“1”，它指示BSBI应该出现。

U.5.2.5 反向信道子比特流指示 (BSBI) (2比特)

BSBI是一个2比特固定长度码字，出现时在BCPM之后。BSBI为前向信道数据中在反向信息中指示ACK/NACK消息的适当的子比特流编号的自然二进制表示（如5.2.4和附件C中所描述的）。

U.5.2.6 图像编号类型 (PNT) (1比特)

PNT是一个单比特固定长度码字，该参数一直出现，在反向信道消息中跟随在BCPM 或 BSBI之后。PNT值的定义如下：

"0": 该消息相关的是由短期图像编号 (PN) 规定的图像。

"1": 该消息相关的是由长效期图像编号(LPIN)规定的图像。

PNT后面可以是PN或者LPIN，这取决于PNT的值。PN或者LPIN应该分别遵守U.3.1.3 和 U.3.1.5.9节规定在前向信道数据中的用法。

U.5.2.7 请求图像编号类型 (RPNT) (2比特)

RPNT是一个2比特固定长度码字，只有在BT指示NACK消息时出现。出现时在PN或者LPIN之后。它用来确定在多图像缓存中那个图像用作后续图像编码的参考。RPNT的值定义如下：

"00": 缓存器中没有有效图像—应该置RESET="1"，使用一个I或者EI图像来重置缓存器。

"01": 没有特定图像确定作为参考图像。

"10": 用作参考图像的图像由短期图像编号 (PN) 确定。

"11": 用作参考图像的图像由长效图像索引(LPIN)确定。

如果RPNT为"10" 或者"11"，RPNT后面应该是PN或者LPIN，这取决于RPNT的值。PN或者LPIN应该分别遵守U.3.1.3 和 U.3.1.5.9节规定在前向信道数据中的用法。

U.5.2.8 附加数据类型 (ADT) (2比特)

ADT是一个2比特固定长度码字，出现在PN、LPIN或者 RPNT之后，由PNT（在ACK消息中）或者RPNT（在NACK消息中）决定。它可能会多次出现。ADT规定用以确定和应用哪个反向信道消息相关的图像域的附加数据类型。ADT的值定义如下：

"00": 附加数据结束。

"01": 仅由GN/MBA 域确定的区域。

"10": 由GN/MBA 和NMBM1确定的在图像内可以作为栅格扫描域的区域。

"11": 由GN/MBA 和NMBM1确定的在矩形截面内可以作为栅格扫描域的区域。

如果ADT为"00"，后向信道消息中，在该参数之后不会再有其他信息。如果ADT为"01"，那么其后为GN/MBA，然后是另外一个ADT。如果ADT为"10" 或 "11"，其后为GN/MBA和NMBM1，然后是另外一个ADT。

如果ADT为"10"，确认作为区域的区域从GN/MBA规定的特定空间位置开始，包含了图像栅格扫描顺序中特定的宏块数量。如果ADT为"11"，确认作为区域的区域从GN/MBA规定的特定空间位置开始，包含了矩形译码栅格扫描顺序中特定的宏块数量。如果ADT只出现一次并且为"00"，确定的区域是整个图像。如果ADT出现超过一次，值"00"用来结束循环而不是确认区域。

U.5.2.9 GOB 编号/宏块地址 (GN/MBA) (5/6/7/9/11/12/13/14比特)

GN/MBA是一个固定长度码字，用来规定GOB编号或者宏块地址。出现时跟随在ADT之后。GN/MBA受ADT的指示出现。如果没有使用可选译码结构模式（见附件K），GN/MBA包含在反向信道消息中指示NACK/ACK消息视频图像分段起始的GOB编号；如果使用可选截面结构模式，则GN/MBA包含截面起始的宏块地址，对该截面而言在反向信道消息中指NACK/ACK消息。GN/MBA的长度为本建议书中别处规定的GN或MBA长度。

U.5.2.10 宏块减1的数量 (NMBM1) (5/6/7/9/11/12/13/14比特)

NMBM1是一个固定长度码字，用来说明宏块的数量。NMBM1受ADT的指示出现，出现时在GN/MBA之后。它是宏块数减1后所得数量的自然数表示。该域的长度应该是K.2.5节和表K.2宏块地址中定义的长度

附 件 V

数据分裂截面模式

V.1 范围

本附件描述数据分裂截面模式，该模式在H.263中是可选的。此模式通过外部手段（如ITU-T H.245建议书）进行能力协商。此模式的使用应该用保留的PLUSPTYPE (OPPTYPE)第17位置“1”来标识。此模式使用附件K中定义的头部分结构。

数据分裂增强了易出错环境下码流的健壮性。通过重新安排H.263的句法结构，使得在传输过程中能够引入错误早期探测和恢复机制。

V.2 数据分裂的结构

当使用了数据分裂时，数据以视频图像片断为单元进行排列，如R.2所定义的。片断中的MB被重新排列，以便于片断中所有MB的头部信息一起被传送，紧接着传送片断中所有MB的运动矢量（MV）数据，最后传送片断内所有MB的DCT系数。片断头部的语法规则见K.2。头部、MV和DCT系数这3部分以特殊的标记分隔，允许出现错误时在某部分的末尾重新同步。所有片断均应包含整数个MB。当使用此模式时，必须采用图V.1所示的语法。

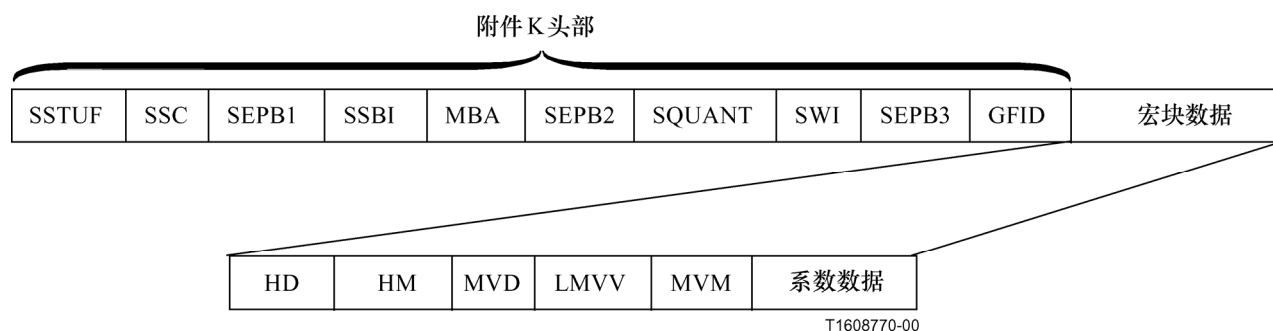


图 V.1/H.263—数据分裂语法

注意，如果不采用本附件，视频图像片断中所有MB的MV和DCT数据将混合传送。在这种情况下，一个错误通常会导致数据包中其余MB的所有信息全部丢失。

V.2.1 头部数据（HD）（可变长度）

头部数据字段包含了数据包中所有宏块（MB）的COD和MCBPC信息，如果是PB帧或改进的PB帧，还包括MODB数据。分组中所有MB的COD和MCBPC字段通过可逆变长编码（RVLC）进行合并。表V.1到V.5给出了RVLC码字。如果使用了附件O，对于B图像和EB图像，则COD字段仅与宏块类型（MB TYPE）合并，生成RVLC，使用的编码表为表V.3和V.4；CBPC则使用表O.4中的码字进行编码。如果COD = 0，并且使用了附件G或M，则COD+MCBPC的码字应该紧跟在该宏块的MODB所对应的可逆变长编码之后。表V.6用于PB帧，表V.7用于改进的PB帧。

V.2.2 头部标记 (HM) (9比特)

9比特码字，其值为1010 0010 1。HM标志着头部部分的结束。译码器在反向译码时搜索该标记。头部标记 (HM) 的值不能在头部数据 (HD) 字段中出现。

V.2.3 运动矢量数据层 (可变长度)

V.2.3.1 运动矢量差值的编码

表D.3中的可逆变长编码码字用来对运动矢量和它的预测值之差进行编码。本附件仅使用了附件D中的熵编码方法，附件D的其他方面则没有使用 (除非同时使用附件D)。

V.2.3.2 运动矢量值的预测

数据包中的第一个运动矢量水平和垂直分量均用0作为预测值，而后续MB的运动矢量则用运动矢量差值 (MVD) 进行预测编码。以跳过模式 (skipped) 或INTRA模式编码的宏块，后面宏块的运动矢量编码方式与此不同，水平和垂直分量均用0作为预测值。

前向: $MV_i = MV_{i-1} + MVD_i = MV_{i-1} + (MV_i - MV_{i-1})$

后向: $MV_{i-1} = MV_i - MVD_i = MV_i - (MV_i - MV_{i-1})$ 。

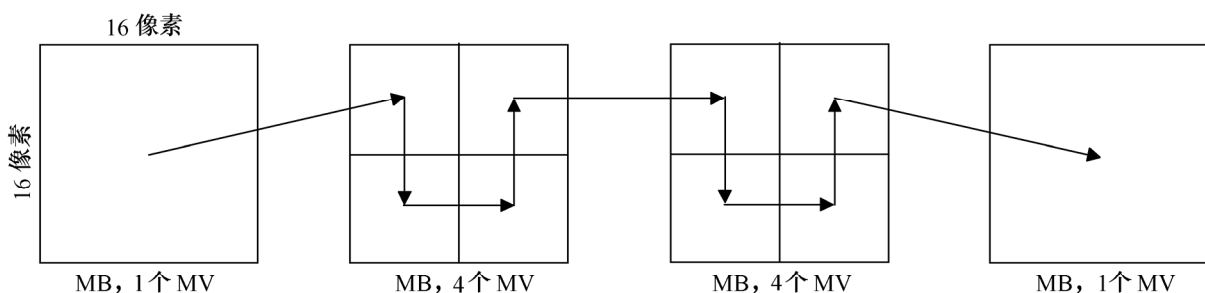
(MV_i 和 MVD_i 分别为每个数据包的第i个运动矢量和第i个运动矢量差值)。

数据包的最后一个运动矢量信息按照上述方法编码，然后再按照V.2.4的规定编码到LMVV字段中。译码器可以独立地对两种不同预测路径的运动矢量进行译码:

- 1) 对于前向预测，从包中运动数据的起点开始;
- 2) 对于后向预测，从包中运动数据的终点开始。

该方法能够提供更好的错误探测和隐藏效果，提高了码流的健壮性。

注1 — 当使用数据分裂截面 (DPS) 模式时，运动矢量采用预测编码，如图6.1.1中所示，当前运动矢量的值采用相邻的3个运动矢量的平均值作为预测值。由于本附件中的数据包中所包含的宏块数目是变化的，用平均值预测编码方法 (参与预测的运动矢量处于一帧中的不同行中) 将使得译码器不能对同一截面中的运动矢量进行反向译码。当采用数据分裂的截面模式时，整个数据包仅有一个预测路径。如图V.2所示。



T1608780-00

图 V.2/H.263—单一运动矢量预测路径

对于B图像或EB图像 (附件O) 的情况，通过MBTYPE按照表V.3和V.4规定的码字指示MVDFW和MVDBW是否出现。MVDFW按照前面所说明的方法进行单路径预测编码，MVDBW (当B图像出现时) 应该按照O.4.6进行编码。MVDFW 和MVDBW的编码应该采用表D.3中的码字。

对于PB-帧（附件G）和改进的PB-帧（附件M），MVDB数据应该按照相应附件编码，并且必须使用表D.3中的码字。

注 — 如果反向译码模式用于B帧（附件O）或改进的PB-帧（附件M），译码器应该丢弃MVDB和MVDBW，因为后向预测的运动矢量数据可能在数据包的边界上没有被正确恢复。

V.2.3.3 在运动矢量差值编码中防止产生伪起始码

MVD编码中防止产生伪起始码的方法与D.2相比有一些变化，以便简化独立后向解析过程。MV部分应该从左到右顺序扫描，在任何两个连续为1（码字为“000”）的MVD之后插入一个MVD=0（码字为“1”）。如果原始的比特流中还有一个值是1的MVD码字跟随在这两个连续为1的码字之后，则应该认为它是后续比特流中的第一个MVD = 1的码字，而不应该认为它是第二个MVD = 1的码字。并且不应在它之后插入MVD = 0的码字。这一点与附件D是不同的，附件D中仅在每个连续出现的MVD = 1（码字为“000”）配对（例如第一个MVD是水平分量，第二个是垂直分量）时才会插入MVD = 0的码字。如果附件D和附件V同时使用，则应该使用附件V中的方法防止伪起始码的产生。

V.2.4 末尾运动矢量值(LMVV)（可变长度）

LMVV字段包含数据包中的最后一个运动矢量。编码时以0作为运动矢量水平和垂直分量的预测值。如果当前数据包中没有其他的运动矢量，或者仅有一个运动矢量，LMVV应该出现。（这里为了能够进行反向译码，使用了固定的0值预测。）

V.2.5 运动矢量标记（MVM）（10比特）

10比特码字，值为“0000 0000 01”。MVM表示运动矢量部分的结束。译码器在反向译码时搜索该标记。如果数据包中不含运动矢量数据（如果包中所有的宏块都使用帧内编码模式或COD等于1），则MVM不应该出现。

V.2.6 系数数据层（可变长度）

DCT数据层包含INTRA_MODE（如果出现）、CBPB（如果出现）、CBPC（如果出现）、CBPY、DQUANT（如果出现）、和DCT系数，它们分别按照I.2、5.3.4、O.4.3、5.3.5、5.3.6、以及5.4.2的规定编码。语法图见图V.3。表V.3和表V.4中指示了CBPC的出现。

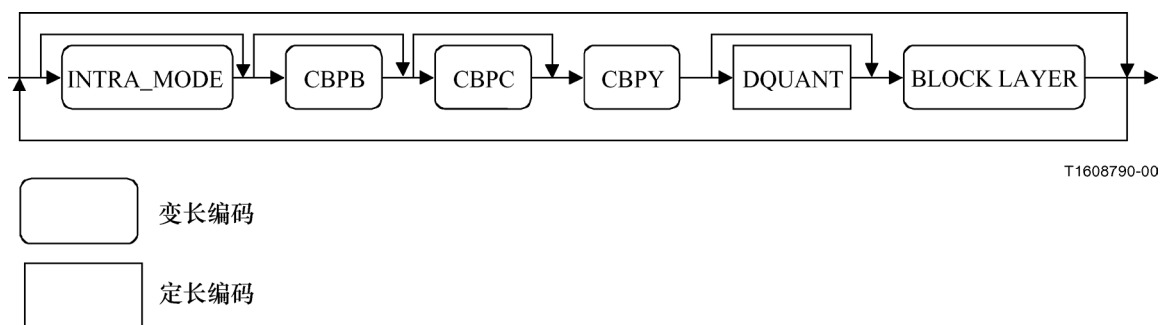


图 V.3/H.263—系数数据的语法

V.3 与其他可选模式的关系

数据分裂截面（DPS）模式可以作为截面结构模式（附件K）一个很有用的子模式使用，可以利用其外部的图像和截面头部结构。因此截面结构（SS）模式必须指示出DPS模式的使用。SS模式其他两个子模式（任意截面顺序和矩形截面子模式）可以与DPS模式一起使用。

附件E规定的基于语法的算术编码模式不能与本附件一起使用，因为采用了附件E就不能进行反向译码。

附加H前向纠错（FEC）不应该和本附件一起使用，因为FEC会导致比特流在某些位置上被打断。但由于附件H所规定的FEC方法在一些现存的系统中是必须的，因此并不禁止附件H与DPS模式一起使用。

附件O中规定的时域、SNR和空间域可分级（TSSS）模式可以与DPS模式一起使用。当TSSS和DPS联合使用时，应该使用表V.3、V.4和V.5所提供的码字，而不是附件O中的码字。

附件U不应该与本模式一起使用。

表 V.1/H.263—用于INTRA MB的COD + MCBPC RVLC表

| MB类型 | CBPC (56) | 码字 (用于联合的 COD + MCBPC) | 比特数 |
|---------------|-----------|------------------------------|-----|
| 3 (INTRA) | 00 | 1 | 1 |
| 3 | 01 | 010 | 3 |
| 3 | 10 | 0110 | 4 |
| 3 | 11 | 01110 | 5 |
| 4 (INTRA + Q) | 00 | 00100 | 5 |
| 4 | 01 | 011110 | 6 |
| 4 | 10 | 001100 | 6 |
| 4 | 11 | 0111110 | 7 |
| 填充 | | 0011100 | 7 |

表 V.2/H.263—用于INTER MB的COD + MCBPC RVLC表

| MB类型 | CBPC (56) | 码字 (用于联合的 COD+MCBPC) | 比特数 |
|-----------------|-----------|----------------------------|-----|
| 跳过 | | 1 | 1 |
| 0 (INTER) | 00 | 010 | 3 |
| 0 | 10 | 00100 | 5 |
| 0 | 01 | 011110 | 6 |
| 0 | 11 | 0011100 | 7 |
| 1 (INTER + Q) | 00 | 01110 | 5 |
| 1 | 10 | 00011000 | 8 |
| 1 | 01 | 011111110 | 9 |
| 1 | 11 | 0111111110 | 11 |
| 2 (INTER4V) | 00 | 0110 | 4 |
| 2 | 10 | 01111110 | 8 |
| 2 | 01 | 00111100 | 8 |
| 2 | 11 | 000010000 | 9 |
| 3 (INTRA) | 00 | 001100 | 6 |
| 3 | 11 | 0001000 | 7 |
| 3 | 10 | 001111100 | 9 |
| 3 | 01 | 000111000 | 9 |
| 4 (INTRA + Q) | 00 | 0111110 | 7 |
| 4 | 11 | 0011111100 | 10 |
| 4 | 10 | 0001111000 | 10 |
| 4 | 01 | 0000110000 | 10 |
| 5 (INTER4V + Q) | 00 | 00111111100 | 11 |
| 5 | 01 | 00011111000 | 11 |
| 5 | 10 | 00001110000 | 11 |
| 5 | 11 | 00000100000 | 11 |
| 填充 | | 0111111110 | 10 |

表 V.3/H.263—用于B宏块的MBTYPE RVLC代码

| 索引 | 预测类型 | MVDFW | MVDBW | CBPC + CBPY | DQUANT | MBTYPE | 比特数 |
|----|-----------|-------|-------|-------------|--------|-----------|-----|
| - | 直接(跳过) | | | | | 1 (COD=1) | 1 |
| 0 | 直接 | | | X | | 010 | 3 |
| 1 | 直接+Q | | | X | X | 001100 | 6 |
| 2 | 前向(无纹理) | X | | | | 00100 | 5 |
| 3 | 前向 | X | | X | | 011110 | 6 |
| 4 | 前向+Q | X | | X | X | 01111110 | 8 |
| 5 | 后向(无纹理) | | X | | | 0110 | 4 |
| 6 | 后向 | | X | X | | 01110 | 5 |
| 7 | 后向+Q | | X | X | X | 00111100 | 8 |
| 8 | 双向直接(无纹理) | X | X | | | 0011100 | 7 |
| 9 | 双向直接 | X | X | X | | 0001000 | 7 |
| 10 | 双向直接+Q | X | X | X | X | 0111110 | 7 |
| 11 | INTRA | | | X | | 00011000 | 8 |
| 12 | INTRA+Q | | | X | X | 011111110 | 9 |
| 13 | 填充 | | | | | 001111100 | 9 |

表 V.4/H.263—用于EP宏块的MBTYPE RVLC表

| 索引 | 预测类型 | MVDFW | MVDBW | CBPC + CBPY | DQUANT | MBTYPE | 比特数 |
|----|------------|-------|-------|-------------|--------|-----------|-----|
| - | 前向(跳过) | | | | | 1 (COD=1) | 1 |
| 0 | 前向 | X | | X | | 010 | 3 |
| 1 | 前向+Q | X | | X | X | 0110 | 4 |
| 2 | 向上(无纹理) | | | | | 01110 | 5 |
| 3 | 向上 | | | X | | 00100 | 5 |
| 4 | 向上+Q | | | X | X | 011110 | 6 |
| 5 | 双向—直接(无纹理) | | | | | 001100 | 6 |
| 6 | 双向—直接 | X | | X | | 0111110 | 7 |
| 7 | 双向—直接+Q | X | | X | X | 0011100 | 7 |
| 8 | INTRA | | | X | | 0001000 | 7 |
| 9 | INTRA+Q | | | X | X | 01111110 | 8 |
| 10 | 填充 | | | | | 00111100 | 8 |

表 V.5/H.263—用于EI宏块的COD + MCBPC RVLC表

| 预测类型 | QCBP (56) | 码字 (用于联合的 COD+MCBPC) | 比特数 |
|---------------|-----------|----------------------------|-----|
| 向上 (跳过) | | 1 | 1 |
| 0 (向上) | 00 | 010 | 3 |
| 0 | 01 | 0110 | 4 |
| 0 | 10 | 01110 | 5 |
| 0 | 11 | 00100 | 5 |
| 1 (向上 + Q) | 00 | 011110 | 6 |
| 1 | 01 | 001100 | 6 |
| 1 | 10 | 0111110 | 7 |
| 1 | 11 | 0011100 | 7 |
| 2 (INTRA) | 00 | 0001000 | 7 |
| 2 | 01 | 01111110 | 8 |
| 2 | 10 | 00111100 | 8 |
| 2 | 11 | 00011000 | 8 |
| 3 (INTRA + Q) | 00 | 011111110 | 9 |
| 3 | 01 | 001111100 | 9 |
| 3 | 10 | 000111000 | 9 |
| 3 | 11 | 000010000 | 9 |
| 填充 | | 0111111110 | 10 |

表 V.6/H.263—用于MODB的RVLC表

| 索引 | CBPB | MVDB | 比特数 | Code |
|----|------|------|-----|-------|
| 0 | | | 3 | 010 |
| 1 | | X | 4 | 0110 |
| 2 | X | X | 5 | 01110 |

注 — “X” 表示该项在宏块中出现。

表 V.7/H.263—用于改进的PB-帧模式下MODB的RVLC表

| 索引 | CBPB | MVDB | 比特数 | 码字 | 编码模式 |
|----|------|------|-----|--------|------|
| 0 | | | 3 | 010 | 双向预测 |
| 1 | X | | 4 | 0110 | 双向预测 |
| 2 | | X | 5 | 01110 | 前向预测 |
| 3 | X | X | 5 | 00100 | 前向预测 |
| 4 | | | 6 | 011110 | 后向预测 |
| 5 | X | | 6 | 001100 | 后向预测 |

注 — “X” 表示相关语法元素的出现。

附 件 W

附加的补充增强信息规范

W.1 范围

本附件描述了在H.263图像层的PSUPP字段中传送附加补充增强信息的格式，它扩展了附件L所定义的功能。本附件所描述的译码器所提供的某些或全部能力可以通过外部手段（如ITU-T H.245建议书）告知对端。不支持附加能力的译码器可以丢弃这些出现在比特流中的新定义的PSUPP信息。此附加增强信息的出现由PEI位和后面的PSUPP字节（其中的FTYPE字段取新定义的2个值之一）共同标识。对于PEI、PSUPP、FTYPE和DSIZE的解释与附件L和5.1.24以及5.1.25相同。

W.2 参考文献

下列ITU-T建议书和其他参考文献的条款，通过在本建议书中的引用而构成本建议书的条款。在出版时，所指出的版本是有效的。所有的建议书和其他参考文献都面临修订，使用本建议书的各方应探讨使用下列建议书和其他参考文献最新版本的可能性。当前有效的ITU-T建议书清单定期出版。本建议书中引用某个独立文件，并非确定该文件具备建议书的地位。

- ISO/IEC 10646:2003，信息技术通用多八位编码字符集（UCS）。
- IETF RFC 2396 (1998)，统一资源标识符（URI）：一般性语法。

W.3 附加的FTYPE取值

在附件L表L.1中保留未用的2个值，这里重新定义在表W.1中。

表 W.1/H.263—FTYPE功能类型取值

| | |
|----|-----------------|
| 13 | 定点离散余弦反变换（IDCT） |
| 14 | 图像消息 |

W.4 PSUPP字节的最大推荐数目

当使用本附件定义的上述FTYPE功能时，每一帧的PSUPP字节总数目应该控制在合理的范围内，这与编码图像大小有关，但是对于任何图像，尺寸不得超过256字节。

注 — 为了增强容错性能，某些用来运载视频比特流的数据传输协议可能会重复传输图像头部数据，并可能对来自图像头部的可重复数据的量（如IETF RFC 2429打包格式规定的504比特）进行限制。如果PSUPP字节过大，可能会导致没有传输协议能完全重复图像头部内容。

W.5 定点离散余弦反变换（IDCT）

定点离散余弦反变换（IDCT）表示在比特流的重建中使用特定的IDCT近似值。对于定点IDCT，DSIZE应该等于1。后续的PSUPP数据说明了这一特殊的IDCT实现。如W.5.3所述，值为0表示参考IDCT 0，1到255的取值保留。

W.5.1 译码器操作

译码器进行定点IDCT的能力可以通过外部手段（如ITU-T H.245建议书）告知编码器。译码器在收到含有定点IDCT指示的比特流时，如果其具有这个能力，则应该使用定点IDCT。

W.5.2 强制更新的去除

附件A规定了离散余弦变换（IDCT）的精度要求，允许有多种实现方法。为了控制由于编码器和译码器的IDCT失配引起的误差，4.4节—强制更新，要求在系数传输时宏块至少每132次使用一次帧内（INTRA）模式的编码。

如果比特流中指出了定点IDCT功能，那么也就不需要强制更新，并且也没有INTRA编码频率的限制。但是除非编码器通过外部手段确知译码器具有这里所规定的定点IDCT能力，编码器仍须继续采用强制更新；否则将出现失配。

W.5.3 参考IDCT 0

参考IDCT 0是对于所有输入块都产生相同输出的实现，如下列C程序代码所示。

注—定点IDCT符合附件A，但是不符合ITU-T H.262建议书|ISO/IEC 13818-2附件A中扩展的取值范围的规定。

```
/*
 *
 *                               FIXED-POINT IDCT
 *
 * Fixed-point fast, separable idct
 * Storage precision: 16 bits signed
 * Internal calculation precision: 32 bits signed
 * Input range: 12 bits signed, stored in 16 bits
 * Output range: [-256, +255]
 * All operations are signed
 *
 *****/

/*
 * Includes
 */

#include <stdlib.h>
#include <stdio.h>

/*
 * Typedefs
 */

typedef short int REGISTER; /* 16 bits signed */
typedef long int LONG; /* 32 bits signed */

/*
 * Global constants
 */
```

```

const REGISTER cpo8   = 0x539f;    /* 32768*cos(pi/8)*1/sqrt(2) */
const REGISTER spo8   = 0x4546;    /* 32768*sin(pi/8)*sqrt(2)   */
const REGISTER cpo16  = 0x7d8a;    /* 32768*cos(pi/16)         */
const REGISTER spo16  = 0x18f9;    /* 32768*sin(pi/16)         */
const REGISTER c3po16 = 0x6a6e;    /* 32768*cos(3*pi/16)       */
const REGISTER s3po16 = 0x471d;    /* 32768*sin(3*pi/16)       */
const REGISTER Oor2   = 0x5a82;    /* 32768*1/sqrt(2)         */

/*
 * Function declarations
 */

void Transpose(REGISTER block[64]);
void HalfSwap(REGISTER block[64]);
void Swap(REGISTER block[64]);
void Scale(REGISTER block[64], signed char sh);
void Round(REGISTER block[64], signed char sh,
           const REGISTER min, const REGISTER max);
REGISTER Multiply(const REGISTER a, REGISTER x, signed char sh);
void Rotate(REGISTER *x, REGISTER *y,
            signed char sha, signed char shb,
            const REGISTER a, const REGISTER b,
            int inv);
void Butterfly(REGISTER column[8], char pass);
void IDCT(REGISTER block[64]);

/*
 * Transpose():
 *     Transpose a block
 * Input:
 *     REGISTER block[64]
 * Output:
 *     block
 * Return value:
 *     none
 */
void Transpose(REGISTER block[64])
{
    int i, j;
    REGISTER temp;

    for (i=0; i<8; i++) {
        for (j=0; j<i; j++) {
            temp = block[8*i+j];
            block[8*i+j] = block[8*j+i];
            block[8*j+i] = temp;
        }
    }
    return;
}

/*
 * HalfSwap():
 *     One-dimensional swap
 * Input:
 *     REGISTER block[64]
 * Output:
 *     block
 * Return value:
 *     none
 */
void HalfSwap(REGISTER block[64])
{

```

```

int i;
REGISTER temp;

for (i=0; i<8; i++) {
    temp = block[8+i];
    block[8+i] = block[32+i];
    block[32+i] = temp;
    temp = block[24+i];
    block[24+i] = block[48+i];
    block[48+i] = temp;
    temp = block[40+i];
    block[40+i] = block[56+i];
    block[56+i] = temp;
}
return;
}

/*
 * Swap():
 *     Swap and transpose a block
 * Input:
 *     REGISTER block[64]
 * Output:
 *     block
 * Return value:
 *     none
 */
void Swap(REGISTER block[64])
{
    HalfSwap(block);
    Transpose(block);
    HalfSwap(block);
}

/*
 * Scale():
 *     Scale a block
 * Input:
 *     REGISTER block[64]
 *     signed char sh
 * Output:
 *     block
 * Return value:
 *     none
 */
void Scale(REGISTER block[64], signed char sh)
{
    int i;

    if (sh>0) {
        for (i=0; i<64; i++)
            block[i] >>= sh;
    }
    else {
        for (i=0; i<64; i++)
            block[i] <<= -sh;
    }
}

/*
 * Round():
 *     Performs the final rounding of an 8x8 block
 * Input:
 *     REGISTER block[64]

```

```

*      signed char sh
*      const REGISTER min
*      const REGISTER max
* Output:
*      block
* Return value:
*      none
*/
void Round(REGISTER block[64], signed char sh,
           const REGISTER min, const REGISTER max)
{
    int i;

    for (i=0; i<64; i++) {
        if (block[i] < 0x00007FFF - (1<<(sh-1)))
            block[i] += (1<<(sh-1));
        else
            block[i] = 0x00007FFF;
        block[i] >>= sh;
        block[i] = (block[i]<min) ? min : ((block[i]>max) ? max : block[i]);
    }
    return;
}

/*
* Multiply():
*      Multiply by a constant with shift
* Input:
*      const REGISTER a
*      REGISTER x
*      signed char sh
* Output:
*      none
* Return value:
*      REGISTER, the result of the multiply
*/
REGISTER Multiply(const REGISTER a, REGISTER x, signed char sh)
{
    LONG tmp;
    REGISTER reg_out;

    /* multiply */
    tmp = (LONG)a * (LONG)x;

    /* shift */
    if (sh > 0)
        tmp >>= sh;
    else
        tmp <<= -sh;

    /* rounding and saturating */
    if (tmp < 0x7FFFFFFF - 0x00007FFF)
        tmp = tmp + 0x00007FFF;
    else
        tmp = 0x7FFFFFFF;

    reg_out = (REGISTER)(tmp >>16);

    return(reg_out);
}

/*
* Rotate():
*      Perform rotate operation on two registers

```

```

* Input:
*   REGISTER *x           pointer to the 1st register
*   REGISTER *y           pointer to the 2nd register
*   signed char sha       shift associated with factor a
*   signed char shb       shift associated with factor b
*   const REGISTER a     factor a
*   const REGISTER b     factor b
*   int inv              1 for inverse dct, 0 for forward dct
* Output:
*   *x, *y
* Return value:
*   none
*/
void Rotate(REGISTER *x, REGISTER *y,
            signed char sha, signed char shb,
            const REGISTER a, const REGISTER b,
            int inv)
{
    LONG tmp1xa, tmp1ya, tmp1xb, tmp1yb;
    LONG tmp11, tmp12;

    /*
     * intermediate calculation
     */

    tmp1xa = (LONG)(*x) * (LONG)a;
    if (sha > 0)
        tmp1xa >>= sha;
    else
        tmp1xa <<= -sha;

    tmp1ya = (LONG)(*y) * (LONG)a;
    if (sha > 0)
        tmp1ya >>= sha;
    else
        tmp1ya <<= -sha;

    tmp1xb = (LONG)(*x) * (LONG)b;
    if (shb > 0)
        tmp1xb >>= shb;
    else
        tmp1xb <<= -shb;

    tmp1yb = (LONG)(*y) * (LONG)b;
    if (shb > 0)
        tmp1yb >>= shb;
    else
        tmp1yb <<= -shb;

    /*
     * rounding and rotation
     */

    if (inv) {
        tmp1xa += 0x00007FFF;
        tmp1xb += 0x00007FFF;

        tmp11 = tmp1xb - tmp1ya;
        tmp12 = tmp1xa + tmp1yb;
    }
    else {
        tmp1ya += 0x00007FFF;
        tmp1yb += 0x00007FFF;
    }
}

```



```

    tmp11 = tmp1xb + tmp1ya;
    tmp12 = -tmp1xa + tmp1yb;
}

/*
 * final rounding
 */

*x = (REGISTER) (tmp11 >>16);
*y = (REGISTER) (tmp12 >>16);

return;
}

/*
 * Butterfly():
 *     Perform 1D IDCT on a column
 * Input:
 *     REGISTER column[8]
 *     char pass
 * Output:
 *     column
 * Return value:
 *     none
 */
void Butterfly(REGISTER column[8], char pass)
{
    int i;
    REGISTER shadow_column[8];

    /*
     * For readability, we use a shadow column
     * that contains the state of column at the
     * preceding stage of the butterfly.
     */

    /*
     * Initialization
     */

    for (i=0; i<8; i++)
        shadow_column[i] = column[i];

    /*
     * First Phase
     */

    Rotate(column+2, column+6, pass-2, pass-1, cpo8, spo8, 1);
    Rotate(column+1, column+7, pass-1, pass-1, cpo16, spo16, 1);
    Rotate(column+3, column+5, pass-1, pass-1, c3po16, s3po16, 1);

    if (pass) {
        int a, tmp=column[4], b=column[0];
        a = b+tmp;
        b = b-tmp;
        column[0] = (a - ((tmp<0) ? 1 : 0)) >> 1;
        column[4] = (b - ((tmp<0) ? 1 : 0)) >> 1;
    }
    else {
        column[0] = shadow_column[0] + shadow_column[4];
        column[4] = shadow_column[0] - shadow_column[4];
    }
}

```

```

for (i=0; i<8; i++)
    shadow_column[i] = column[i];

/*
 * Second Phase
 */

column[1] = shadow_column[1] - shadow_column[3];
column[3] = shadow_column[1] + shadow_column[3];

column[7] = shadow_column[7] - shadow_column[5];
column[5] = shadow_column[7] + shadow_column[5];

column[0] = shadow_column[0] + shadow_column[6];
column[6] = shadow_column[0] - shadow_column[6];

column[4] = shadow_column[4] + shadow_column[2];
column[2] = shadow_column[4] - shadow_column[2];

for (i=0; i<8; i++)
    shadow_column[i] = column[i];

/*
 * Third Phase
 */

column[7] = shadow_column[7] - shadow_column[3];
column[3] = shadow_column[7] + shadow_column[3];

column[1] = Multiply(OoR2, shadow_column[1], -2);
column[5] = Multiply(OoR2, shadow_column[5], -2);

for (i=0; i<8; i++)
    shadow_column[i] = column[i];

/*
 * Fourth Phase
 */

column[4] = shadow_column[4] + shadow_column[3];
column[3] = shadow_column[4] - shadow_column[3];

column[2] = shadow_column[2] + shadow_column[7];
column[7] = shadow_column[2] - shadow_column[7];

column[0] = shadow_column[0] + shadow_column[5];
column[5] = shadow_column[0] - shadow_column[5];

column[6] = shadow_column[6] + shadow_column[1];
column[1] = shadow_column[6] - shadow_column[1];

return;
}

/*
 * IDCT():
 *     Perform 2D IDCT on a block
 * Input:
 *     REGISTER block[64]
 * Output:
 *     block
 * Return value:
 *     none
 */

```

```

void IDCT(REGISTER block[64])
{
    int i;

    Scale(block, -4);

    for (i=0; i<8; i++)
        Butterfly(block+8*i, 0);

    Transpose(block);

    for (i=0; i<8; i++)
        Butterfly(block+8*i, 1);

    Round(block, 6, -256, 255);

    Swap(block);
}

```

下面给出前向余弦变换（FDCT）的实现，以供参考。定点FDCT不包括在本建议书中。

```

/*****
 *
 *                               FIXED-POINT FDCT
 *
 * Fixed-point fast, separable fdct
 * Storage precision: 16 bits signed
 * Internal calculation precision: 32 bits signed
 * Input range: 9 bits signed, stored in 16 bits
 * Output range: [-2048, +2047]
 * All operations are signed
 *
 *****/

/*
 * Function declarations
 */

void FButterfly(REGISTER column[8]);
void FDCT(REGISTER block[64]);

/*
 * FButterfly():
 *     Perform 1D FDCT on a column
 * Input:
 *     REGISTER column[8]
 * Output:
 *     column
 * Return value:
 *     none
 */
void FButterfly(REGISTER column[8])
{
    int i;
    REGISTER shadow_column[8];

    /*
     * For readability, we use a shadow column
     * that contains the state of column at the
     * preceding stage of the butterfly.
     */
}

```

```

/*
 * Initialization
 */

for (i=0; i<8; i++)
    shadow_column[i] = column[i];

/*
 * First Phase
 */

for (i=0; i<4; i++) {
    column[i]    = shadow_column[i] + shadow_column[7-i];
    column[7-i] = shadow_column[i] - shadow_column[7-i];
}

for (i=0; i<8; i++)
    shadow_column[i] = column[i];

/*
 * Second Phase
 */

column[0] = shadow_column[0] + shadow_column[3];
column[3] = shadow_column[0] - shadow_column[3];

column[1] = shadow_column[1] + shadow_column[2];
column[2] = shadow_column[1] - shadow_column[2];

column[4] = Multiply(OoR2, shadow_column[4], -2);
column[7] = Multiply(OoR2, shadow_column[7], -2);

column[6] = shadow_column[6] - shadow_column[5];
column[5] = shadow_column[6] + shadow_column[5];

for (i=0; i<8; i++)
    shadow_column[i] = column[i];

/*
 * Third Phase
 */

column[0] = shadow_column[0] + shadow_column[1];
column[1] = shadow_column[0] - shadow_column[1];

column[6] = shadow_column[6] - shadow_column[4];
column[4] = shadow_column[6] + shadow_column[4];

column[7] = shadow_column[7] - shadow_column[5];
column[5] = shadow_column[7] + shadow_column[5];

for (i=0; i<8; i++)
    shadow_column[i] = column[i];

/*
 * Fourth Phase
 */

Rotate(column+2, column+3, -2, -1, cpo8, spo8, 0);
Rotate(column+4, column+5, -1, -1, cpo16, spo16, 0);
Rotate(column+6, column+7, -1, -1, c3po16, s3po16, 0);

return;
}

```

```

/*
 * FDCT():
 *     Perform 2D FDCT on a block
 * Input:
 *     REGISTER block[64]
 * Output:
 *     block
 * Return value:
 *     none
 */
void FDCT(REGISTER block[64])
{
    int i;

    for (i=0; i<8; i++)
        FButterfly(block+8*i);

    Transpose(block);

    for (i=0; i<8; i++)
        FButterfly(block+8*i);

    Round(block, 3, -2048, 2047);

    Swap(block);
}

```

W.6 图像消息

图像消息函数指示出一个或多个表示消息数据的字节的出现。消息的第一个字节是消息头，具有如下结构，如图W.1所示

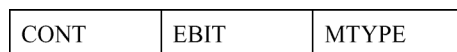


图 W.1/H.263 – 消息的第一个字节结构

对应图像消息函数，DSIZE应该等于消息数据的字节数，包括图W.1所示的字节在内。

译码器应该按照基本的PSUPP语法要求解析图像消息数据，但不定义译码器对图像消息的具体响应。

W.6.1 接续 (CONT) (1比特)

如果CONT字段为“1”，表示与此图像消息函数关联的消息数据，和与下一个图像消息功能关联的消息数据都是同一逻辑消息的一部分。如果CONT字段为“0”，则表示与此图像消息函数关联的消息数据是当前逻辑消息的终结。CONT也可以有其他的用处，比如用来表示超过14个字节的逻辑消息。

W.6.2 结尾比特位置或轨道号 (EBIT) (3 比特)

对于非文本的图像消息，EBIT规定了消息末尾字节中应该忽略的最低比特数。在非文本图像消息中，如果CONT为“1”，或者如果仅有一个消息字节（例如图W.1中的字节），EBIT应该等于“0”。非文本图像消息函数中有效的消息比特数等于 $(DSIZE - 1) \times 8 - EBITS$ ，不包括CONT/EBIT/MTYPE字段。由于接续的存在，逻辑消息中的有效比特数可能更大。

对包含文本信息的消息类型，EBIT应该包含文本轨道数。这里不规定文本轨道的精确含义，但是应该指出文本的类型（如语言）。轨道零应该被当做缺省轨道。

W.6.3 消息类型（MTYPE）（4比特）

MTYPE指出消息的类型。定义见表W.2。

表 W.2/H.263—MTYPE消息类型的取值

| | |
|--------|-----------------------|
| 0 | 任意二进制数据 |
| 1 | 任意文本 |
| 2 | 版权文本 |
| 3 | 标题文本 |
| 4 | 视频描述文本 |
| 5 | 统一资源标识符（URI）文本 |
| 6 | 当前图像头部重复 |
| 7 | 前一图像头部重复 |
| 8 | 后一图像头部重复，可靠的时域参考（TR） |
| 9 | 后一图像头部重复，不可靠的时域参考（TR） |
| 10 | 顶场标识符 |
| 11 | 底场标识符 |
| 12 | 图像编号 |
| 13 | 备用参考图像 |
| 14..15 | 保留 |

W.6.3.1 任意二进制数据

任意二进制数据用于传递任何非ISO/IEC 10646 UTF-8编码的二进制消息。对任意二进制数据内容的解释不在本建议书的范围之内，但是应该以某些确定的模式（如一个4字节的标识码）开头，以便于和其他形式的数据相区分。

W.6.3.2 任意文本

任意文本用于传递通用的ISO/IEC 10646 UTF-8编码消息。版权信息等具体的文本消息应该用其他合适的消息类型（如版权文本）表示。

W.6.3.3 版权文本

版权文本应该只被用于传递关于比特流中源或编码表示内容的知识产权信息。版权消息应该按照ISO/IEC 10646 UTF-8进行编码。

W.6.3.4 标题文本

标题文本应该只被用于传递与比特流中当前或后续图像相关联的标题信息。标题消息应该按照ISO/IEC 10646 UTF-8进行编码。标题文本应该插入到比特流中，如同在一个单独的文本区域播放一样，新文本衔接在先前文本的后面，先前文本从插入点向另一端滚动。应该使用换页（十六进制 "0x000C"）控制码来指示可见文字区域的清除。应该使用媒体结尾（十六进制 "0x0019"）控制码指示“标题结束”状态。然而，本建议书没有对标题文本在实际使用中如何显示和存储做出任何限制。

W.6.3.5 视频描述文本

视频描述文本应该只用于传递当前比特流中与信息内容相关的描述信息。视频描述应该按照ISO/IEC 10646 UTF-8进行编码。视频描述文本应该插入到比特流中，如同在一个单独的文本区域播放一样，新文本衔接在先前文本的后面，先前文本从插入点向另一端滚动。应该使用换页（十六进制 "0x000C"）控制码来指示可见文字区域的清除。应该使用媒体结尾（十六进制 "0x0019"）控制码指示“描述结束”状态。然而，本建议书没有对描述文本在实际使用中如何显示和存储做出任何限制。

W.6.3.6 统一资源标识符（URI）文本

此消息由一个统一资源标识符构成，定义见RFC 2396。URI应该按照ISO/IEC 10646 UTF-8进行编码。

W.6.3.7 当前图像头部重复

当前图像的头部在此消息中被重复。重复的比特中不包含任何增补的增强信息（PEI/PSUPP）。所有其他直至块组层或截面层的位都应该被重复，同时符合W.4的规定。

W.6.3.8 前一图像头部重复

前一幅被传输图像的头部在此消息中被重复。重复的比特中不包含前两个字节的图像起始代码（PSC）和任何增补的增强信息（PEI/PSUPP）。所有其他直至块组层或截面层的位都应该被重复，同时符合W.4的规定。

W.6.3.9 后一图像重复，可靠的时域参考（TR）

下一幅将要被传输图像的头部在此消息中被重复。重复的比特中不包含前两个字节的图像起始代码（PSC）和任何增补的增强信息（PEI/PSUPP）。所有其他直至块组层或截面层的位都应该被重复，同时符合W.4的规定。

W.6.3.10 后一图像重复，不可靠的时域参考（TR）

下一幅将要被传输图像的头部在此消息中被重复。重复的比特中不包含前三个字节和任何增补的增强信息（PEI/PSUPP）。所有其他直至块组层或截面层的位都应该被重复，同时符合W.4的规定。所重复的图像头部中的任何TR或ETR位都不一定与下一幅图像头部中的相应位一致。

W.6.3.11 隔行场标识

标识隔行场的消息由一个代表隔行场编码的指示符构成。这个指示符表明当前图像不是逐行扫描的，但并不影响译码过程。它指示当前编码图像只包含了一幅完整图像的一半的行。对于隔行场指示符，DSIZE应该为1，CONT应该为0，EBIT应该为0。在隔行场编码情况下，时域参考每增加一次，表示交替两场之间的时间，而不是两个完整帧之间的时间。标识顶场时，当前图像包含完整帧中的第一、三、五……行。标识底场时，当前图像包含完整帧中的第二、四、六……行。当发送隔行场标识时，编码器应该遵照如下规定：

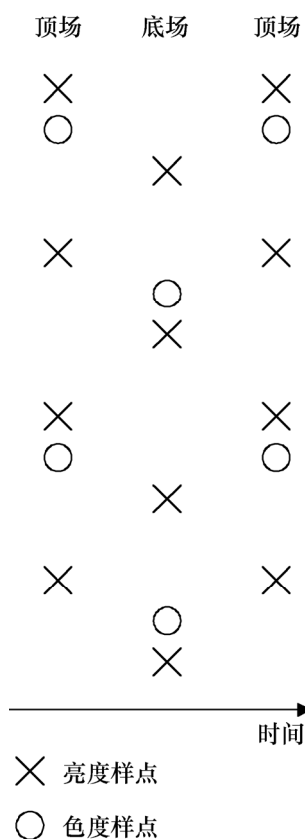
- 1) 编码器应该使用一个合适的图像时钟频率（如果需要，可以为定制的时钟频率），使得原图像每产生一个新场对应时域参考增加1。
- 2) 编码器应该使用合适的图像尺寸（如果需要，可以为定制尺寸），使得图像尺寸与单个场一致。
- 3) 编码器应该使用合适的像素宽高比（如果需要，可以为定制的宽高比），使得图像宽高比与当前编码图像所代表的单个场的宽高比的衍生值相对应。

隔行场扫描最初是作为模拟技术引进的。虽然逐行图像扫描适合于数字压缩和显示，但是隔行场扫描仍然在很多摄像机和显示器中使用。所以这里也支持隔行场扫描（实现中，比隔行全图像编码和逐行扫描图像编码的时延低）的标识。

编码器只有在译码器建立了接收和正确处理基于场的图像的能力（例如通过ITU-T H.245建议书）之后才能发送隔行标识。译码器此能力如果没有成功建立，将会在译码器上受到并显示的译码图像中产生小幅度的垂直抖动。

例如，编码器可能伴随参考图像选择模式（附件N）或增强的参考图像选择模式（附件U）一起使用隔行场编码，以允许对先前的多个场进行寻址。对宽高比为4：3，每行704个亮度样点，每场240个亮度行，“520/60”隔行场编码的图像，编码器应该使用704样点每行，240行每场，宽高比为5：11的定制图像大小，并使用时钟转换代码“1”，时钟约数为30。对宽高比为4：3，每行704个亮度样点，每场288个亮度行，“625/60”隔行场编码的图像，编码器应该使用704样点每行，288行每场，宽高比为6：11的定制图像大小，并使用时钟转换代码“0”，时钟约数为36。

这里规定：隔行场编码的顶场中色度样点的垂直采样位置，相对于场采样网格上移1/4个亮度样点位置，以便与整幅图像中的常规位置垂直对齐；隔行场编码的底场中色度样点的垂直采样位置，相对于场采样网格下移1/4个亮度样点位置，以便与整幅图像中的常规位置垂直对齐。色度样点的水平采样位置不受隔行场编码的影响。垂直采样位置及其对应的时域采样位置如图W.2所示。



T1608800-00

图 W.2/H.263—隔行场编码中的色度样点垂直和时域对齐

W.6.3.12 图像编号

如果使用了附件U，则应该使用该消息。该消息包含2字节的数据，其中携带10比特的图像编号。因此DSIZE应该为3，CONT应该为0，EBIT应该为6。对于每个新编码传输的I帧或P帧或PB帧，图像编号应该递增1，并按10比特数取模。对于EI和EP图像，图像编号应该在同一扩展增强层范围内递增。B图像的编号应该在其后最近的一帧非B图像（B图像的参考图像，在比特流中的顺序先于该B图像）编号的基础上递增。如果同一增强层中的相邻图像具有相同的时域参考，并且如果使用了参考图像选择模式（附件N），那么译码器应该认为同一图像内容的冗余拷贝，所有这些图像共用相同的图像编号。如果在同一增强层内收到的两个相邻非B图像的图像编号之差（模1024）不为1，并且它们不是上述相同图像场景内容的冗余拷贝，那么译码器应该认为发生了图像丢失或数据损坏。

W.6.3.13 备用参考图像

编码器可以使用此消息通知译码器哪一帧与当前的运动补偿参考图像更接近，以便从中选出备用参考图像，在真正的参考图像丢失时使用。当缺少参考图像但是可以获取一个备用的参考图像时，译码器应该请求INTRA图像更新。这取决于编码器对备用参考图像的选择。消息数据包含备用参考图像的图像编号（按照选作参考的顺序，优先选用的先出现。）图像编号是指按照附件U或W.6.3.12规定的数值。此消息可以用于P、B、PB、改进的PB以及EP等类型的图像。但是，如果使用了附件N和附件U，并且图像和多个参考图像有关，则应该使用此消息。对于EP图像，此消息只能用于前向预测中，但是向上预测通常来自临时的对应的参考层图像。对于B、PB和改进的PB图像，规定了一个用于前向运动预测参考图像。如果图像是I或EI图像，则不应使用此消息。

附 件 X

档次和级别的定义

X.1 范围

在使用本建议书提供的多个可选模式时，为使那些采用了可选模式的终端使用比“基线”更好的句法相互连接，应定义若干优先操作模式组合。本附件包含一系列优先特征的组合，即所支持的档次。此外，还定义了多个最大性能参数组合，即这些档次下所支持的级别。本附件的主要目标是：

- 1) 提供一种描述和协商译码器能力的方法（通过规定档次和级别的参数）；
- 2) 鼓励译码器支持公共的增强特性，以达到最大程度的互操作性；
- 3) 描述特征集以便确定一些关键应用。

下面的条款和表X.1、X.2对档次和级别做了定义。表X.2中规定的最小图像间隔为比特流中相继图像译码时刻之差的最小值。对级别45以外其他外级别的支持隐含对所有更低级别的支持。对级别45的支持隐含对级别10的支持。

X.2 优先模式的档次支持

所支持的档次是通过译码器支持该档次的特征集合定义的。译码器支持一个档次，隐含着对该档次中任意模式组合的合法子集的支持。这一规定减少了对编码器在模式组合选择时的限制。这是符合本附件的基本目标的，即描述译码器应该支持哪些可选模式，从而确定关键应用，而不是在编码器上强制使用一个小而特殊的模式组合。

X.2.1 基线档次（档次0）

基线档次，称为档次0，是指提供本建议书最小“基线”能力的档次。“基线”是指本建议书中没有可选操作模式的句法。此档次的支持仅由基线设计构成。

X.2.2 H.320编码效率版本2后向兼容档次H.320（档次1）

H.320编码效率版本2后向兼容档次，称为档次1，设计用于保证H.320电路交换终端系统中H.242能力交换机制的兼容性。在本建议书第二版（不包括附件U、V和W）的特性集范围内，提供了基本的编码效率增强，和简单的功能增强。本档次由基线加上下列模式组成：

- 1) **先进帧内编码（附件I）**— 此模式提高了INTRA宏块（无论是在INTRA图像中还是在预测编码图像中）的编码效率。在编码器和译码器上由此模式引起的额外计算量是很小的（译码过程中每个 8×8 块最多8个加/减操作，加上编码过程中为了提高编码效率所使用的不同但差别很小的VLC表）。基于这些原因，先进帧内编码被包含在基本的工具包之中。
- 2) **去块效应滤波器（附件J）**— 去块滤波器可以明显改善视频图像的主观质量，因此被广泛用于视频通信终端的后处理过程中。因为去块滤波器被置于编码环路之内，附件J代表了去块效应滤波器的首选操作模式。这样的安排使滤波器更容易实现，也在一定程度上提高了编码性能。与先进预测模式类似，此模式也允许每宏块有4个运动矢量，并支持运动补偿时的边界外推，这两个特性都可以进一步提高编码性能。去块滤波器的计算量为每宏块几百个操作，但是存储访问和计算的依赖并不复杂。正因为后面这一点，去块滤波器对于某些实现而言比先进预测模式更好。并且，先进预测模式获得的益处同时在采用了去块滤波器的情况下并不显著。因此，去块滤波器被纳入基本工具包。
- 3) **全图像冻结附加增强信息（附件L，条款L.4）**— 全图像冻结实现起来非常简单，只需要译码器能够停止从输出缓存区向播放器转移新的图像即可。这个能力在当编码器正在建立一个更高保真度图像过程重，需要阻止低保真度图像显示时是很有用的。
- 4) **修正的量化（附件T）**— 此模式包括一个扩展的DCT系数范围，修改了的DQUANT句法，以及经过修改的色度量化步长。前面2个特性允许编码器有更大的灵活性，并且能切实减轻编码器的计算负荷（发生系数级饱和时不需要重新对宏块进行编码）。第3个特性明显提高了色度分量的保真度，代价是比特率有微小的增加，但并不增加计算量。在译码器上，增加的主要计算负荷仅是解析一些新的比特流符号。

X.2.3 版本1后向兼容档次（档次2）

版本1后向兼容档次，称为档次1，在这里定义为采用ITU-T H.263建议书第一版（不包括附加增强信息或任何使用了PLUSPTYPE字段的可选特性）中的特性集来提供增强的编码效率。本档次由基线加上下面的模式组成：

- 1) **先进预测（附件F）** — 从编码效率的角度来看，此模式是本建议书第一版中最为重要的模式。它包括了重叠块运动补偿和每宏块4运动矢量两个特性，并且允许运动矢量指向图像边界之外。先进预测模式的使用极大地提高了图像的客观质量和主观质量。然而，它需要更多的计算量，并且在译码器上引入了复杂的数据处理顺序上的依赖关系。但是，由于本建议书的实现已经先于其他模式的采用而实现了高级预测模式，为了能够保证与版本1译码器兼容的最佳兼容，推荐采用高级预测模式。

X.2.4 版本2交互和流式无线档次（档次3）

版本2交互和流式无线档次，称为档次3，在这里被定义为采用本建议书版本2（不包括附件U、V和W）中的特性集提供增强的编码效率和增强的容错性，用于向无线设备传输视频的应用之中。本档次由基线和下列模式组成：

- 1) **高级INTRA编码（附件I）** — 参见X.2.2条款1。
- 2) **去块滤波器（附件J）** — 参见X.2.2条款2。
- 3) **截面结构模式（附件K）** — 将截面结构模式包括在内是因为它在比特流中提供了重同步点，用于从差错或数据丢失中恢复。任意截面顺序（ASO）和矩形截面（RS）两个子模式不包括在本档次内，以限制译码器的复杂度。截面结构模式所引入的额外计算负担主要来自比特流的产生和解析，已经达到最小。
- 4) **改进量化（附件T）** — 参见X.2.2条目4。

X.2.5 版本3交互和流式无线档次（档次4）

版本3交互和流式无线档次，称为档次4，在这里被定义为采用本建议书版本3（不包括附件U、V和W）中的特性集提供增强的编码效率和增强的容错性，用于向无线设备传输视频的应用之中。本档次由基线和下列模式组成：

- 1) **档次3** — 此特性集提供了若干对于无线视频传输很有用的增强。
- 2) **数据分裂的截面模式（档次V）** — 此特性将截面中的运动矢量数据从DCT系数数据中分离出来，从而加强了容错性能，并且使用可逆变长编码保护运动矢量数据（宏块细节最重要的数据）。任意截面顺序（ASO）和矩形截面（RS）两个子模式不包括在本档次内，以限制译码器的复杂度。
- 3) **先前图像头部重复附加信息（附件W，条款W.6.3.8）** — 此特性允许译码器在数据丢失或损坏情况下从先前图像中接收和恢复头部信息。

X.2.6 会话高压缩档次（档次5）

会话高压缩档次，称为档次5，在这里被定义为不引入与B帧相关的时延，并且不添加提高容错性的特性的条件下提高编码效率。本档次由基线加上下列特性组成：

- 1) **档次1** — 此特性集提供了若干提高编码效率的增强。
- 2) **档次2** — 此档次增加了先进预测模式（附件F），它进一步提高了编码效率，并且提供了与本建议书第一版的后向兼容能力。
- 3) **无限制运动矢量并且UUI=“1”（附件D）** — 附件包含2个基本特性：
 - a) 图像边界外推；和
 - b) 支持更长的运动矢量。

档次1中包含附件J，因此其中第一个特性通过档次被支持。特别是对于大画面、快速运动、摄像机移动和较低的图像速率而言，支持更长的运动矢量可以获得编码效率的显著提高。当出现PLUSPTYPE时，此模式还允许更长的运动矢量之差，它可以显著简化编码操作。更长的运动矢量展现了译码器上存储访问中的潜在问题，但是由于运动矢量的最大值是受限于图像大小的，因此这个问题不会造成实现上的障碍。

- 4) **增强的参考图像选择（附件U）** — 此模式从先前图像中选择多个图像作为宏块级运动预测的参考，显著提高了压缩效率。其子模式子图像移除（附件U，条款U.4.3）不包括在档次5之中。

X.2.7 会话因特网档次（档次6）

会话因特网档次，称为档次6，在这里被定义为在不引入与B帧相关的时延情况下提高编码效率，并添加了一些适合在IP网（IP网络采用了基于包的数据协议，传输相对较大的数据分组，数据丢失多于数据差错）上使用的容错机制。本档次由基线加上下列特性组成：

- 1) **档次5** — 此特性集给出了提高编码效率的若干增强。
- 2) **附带任意截面顺序子模式的截面结构模式（附件K）** — 这里使用截面结构模式是为了利用其提供比特流中的再同步点功能，以便于从数据包丢失恢复。同时还包含了任意截面顺序（ASO）子模式，以便于对运动补偿误差采用交错打包的方法进行差错隐藏，以及乱序数据的接收。为了限制对译码器复杂度的需求，本档次没有包含矩形截面子模式。截面结构模式所引入的额外计算负担达到最小，主要来自比特流的产生和解析。

X.2.8 会话隔行扫描档次（档次7）

会话交织档次，称为档次7，在这里被定义为低时延应用条件下提高编码效率，并支持隔行扫描的视频源。本档次由基线加上下列附加特性组成：

- 1) **档次5** — 此特性集给出了提高编码效率的若干增强，并且没有引入额外时延。
- 2) **用于240线和288线图像的隔行扫描场标识（附件W，条款W.6.3.11）** — 此特性允许视频按照与现有摄像机兼容的隔行扫描图像格式发送。

X.2.9 高延迟档次（档次8）

高延迟档次，称为档次8，在这里被定义为在对延迟没有严格限制的应用中提高编码效率。本档次由基线加上下列附加特性组成：

- 1) **档次6** — 此特性集给出了提高编码效率和数据鲁棒性的若干增强。
- 2) **参考图像重采样（只含显式4因子模式）** — 参考图像重采样中的显式4因子模式在图像头部指出的情况下，允许仅在新帧大小变化时自动对参考图像进行重采样。此操作模式没有比特开销。预测式的动态分辨率调整使得编码器可以灵活地在时域分辨率和空域分辨率之间进行折中。另外，由于在4因子情况下使用简单的有限冲击响应（FIR）滤波器（每像素大约需要4个操作）即可，因此附件P的这一最简单模式（只包含显式4因子上采样或下采样）在编码器和译码器上没有增加太多的计算量。
- 3) **B帧（时域可扩展性，附件O，条款O.1.1）** — 此特性允许使用B帧，即在时域进行双向预测。B帧提高了编码效率，但同时也需要更大的处理能力，并且引入了额外的编码和译码时延。增强的参考图像选择模式中B帧的2帧后向预测子模式（附件U，条款U.3.1.5.5）不包括在档次8之中。

X.3 图像格式和图像时钟频率

为保证良好的互操作性，支持较大图像格式（QCIF, CIF, 4CIF, 16CIF）的编码器和译码器同时也应该支持所有更小的图像格式。所有符合本附件所规定之档次和级别的译码器都必须满足这一要求。例如，若一个译码器符合本附件规定的一个档次和级别，能够译码4CIF格式的图像，那么也必须能够译码CIF图像。

译码器对于较小的图像格式所能支持的最大帧率应该不低于较大的标准图像格式下的最大帧率。所有符合本附件所规定之档次和级别的译码器都必须满足这一要求。例如，若一个译码器符合本附件规定的一个档次和级别，能够译码25帧/秒的4CIF格式的图像，那么也必须能够译码25帧/秒的CIF、QCIF和SQCIF图像。

建议支持定制图像格式和/或图像频率的编码器和译码器遵循本段下面的规则。这些规则是对所有符合本附件所定义之档次和级别的译码器的共同要求：

- 1) 支持这里所规定的任何档次和级别的译码器，必须能够译码标准的宽度和高度小于或等于所能译码最大图像格式的图像。例如，一个支持 720×288 定制格式的译码器，也必须能够译码CIF、QCIF和sub-QCIF图像。
- 2) 这里所定义的任何档次和级别的译码器，如果支持定制图像格式，则其必须支持所有长和宽小于所支持最大图像格式的图像。
- 3) 这里所定义的任何档次和级别的译码器，其支持的最小图像间隔是基于标准帧频率的，即 $(30\ 000/1001)$ 个单元每秒，那么对于所有支持的，高和宽等于和小于规定了最小图像间隔的最大图像格式的图像，也必须支持相同大小或更小图像间隔。
- 4) 这里所定义的任何档次和级别的译码器，支持最小图像间隔和定制图像频率，对于所有支持的，规定了最小图像间隔，等于和小于最大尺寸的图像格式，应该支持相同大小和更小图像的任何图像时钟频率的使用。

X.4 性能级别

定义了8个性能级别用于译码器实现。档次0到4的最小的虚拟参考译码器在表X.1中说明。对于档次5到8，虚拟参考译码器更大，并支持多参考图像下的增强的参考图像选择。表X.2定义了这些级别详细的性能参数：

- 1) **级别10** — 支持QCIF和sub-QCIF分辨率的译码，能够在 $64\ 000$ 比特每秒的速率上运行，每秒钟可译码 $(15\ 000)/1001$ 帧。
- 2) **级别20** — 支持CIF、QCIF和sub-QCIF分辨率的译码，能够在 $2 \times (64\ 000) = 128\ 000$ 比特每秒的速率上运行，对于CIF格式每秒钟可译码 $(15\ 000)/1001$ 帧，对于QCIF和sub-QCIF格式每秒钟能够译码 $(30\ 000)/1001$ 帧。
- 3) **级别30** — 支持CIF、QCIF和sub-QCIF分辨率的译码，能够在 $6 \times (64\ 000) = 384\ 000$ 比特每秒的速率上运行，每秒钟能够译码 $(30\ 000)/1001$ 帧。
- 4) **级别40** — 支持CIF、QCIF和sub-QCIF分辨率的译码，能够在 $32 \times (64\ 000) = 2\ 048\ 000$ 比特每秒的速率上运行，每秒钟能够译码 $(30\ 000)/1001$ 帧。
- 4.5) **级别45** — 支持QCIF和sub-QCIF分辨率的译码，能够在 $2 \times (64\ 000) = 128\ 000$ 比特每秒的速率上运行，每秒钟可译码 $(15\ 000)/1001$ 帧。另外，在除0和2之外的其他档次中，支持QCIF和更小的定制图像格式。
- 5) **级别50** — 支持定制图像格式和CIF及以下分辨率的译码，能够在 $64 \times (64\ 000) = 4\ 096\ 000$ 比特每秒的速率上运行，对于CIF及以下格式每秒钟可译码50帧，对于 353×240 和更小格式每秒钟能够译码 $(60\ 000)/1001$ 帧。

- 6) **级别60** — 支持720×288和更小分辨率的译码，能够在 $128 \times (64\ 000) = 8\ 182\ 000$ 比特每秒的速率上运行，对于720×288及以下分辨率每秒钟能够译码50帧，对于720×240及以下分辨率每秒钟能够译码(60 000)/1001帧。
- 7) **级别70** — 支持定制的和720×576及以下标准格式图像的译码，能够在 $256 \times (64\ 000) = 16\ 384\ 000$ 比特每秒的速率上运行，对于720×576及以下分辨率每秒钟能够译码50帧，对于720×480及以下分辨率每秒钟能够译码(60 000)/1001帧。

对于特定的档次和级别，实际使用时不应超过本附件所规定的比特率限制。但是，特定的系统也可以使用其他手段传递比特率信息。在特定系统中使用时，其他有关档次和级别方面的问题可能也取决于那些对能力的额外限制，但是译码特定档次和级别下的任何比特流所需的能力不应超过本附件的规定。

表 X.1/H.263—档次总结

| 与右边所列档次相关的附件/条款 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|--|---|---|---|---|---|---|---|---|---|
| 5.1.5: 定制图像格式 (CPFMT) | L | L | L | L | L | L | L | L | L |
| 5.1.7: 定制图像时钟频率 (CPCFC) | L | L | L | L | L | L | L | L | L |
| C: 连续出现多点和视频复用 | | | | | | | | | |
| D.1: 图像边界上的运动矢量 | | X | X | X | X | X | X | X | X |
| D.2: UII = '1 或 UII 不出现: 运动矢量范围扩展 | | | | | | X | X | X | X |
| D.2 UII = '01': 无限制运动矢量范围扩展 | | | | | | | | | |
| E: 基于语法的算术编码 | | | | | | | | | |
| F.2: 每宏块 4 运动矢量 | | X | X | X | X | X | X | X | X |
| F.3: 重叠块运动补偿 | | | X | | | X | X | X | X |
| G: PB 帧 | | | | | | | | | |
| H: 前向纠错 (ITU-T H.320 建议书系统层引用) | | | | | | | | | |
| I: 高级帧内编码 | | X | | X | X | X | X | X | X |
| J: 去块效应滤波器 | | X | | X | X | X | X | X | X |
| K (不含子模式): 截面结构 — 不含子模式 | | | | X | X | | X | | X |
| K (带 ASO 子模式): 截面结构 — 附带任意截面顺序子模式 | | | | | | | X | | X |
| K (带 RS 子模式): 截面结构 — 附带矩形截面子模式 | | | | | | | | | |
| L.4: 附加全图像冻结增强 | | X | | | | X | X | X | X |
| L: 附加增强 — 其他附加增强信息特性 | | | | | | | | | |
| M: 改进的 PB-帧 | | | | | | | | | |
| N: 参考图像选择 (含子模式) | | | | | | | | | |
| O.1.1 时域 (B 帧): 时域、信噪比和空域可分级—B 帧用于时域可分级 | | | | | | | | | X |

表 X.1/H.263—档次总结 (续完)

| 与右边所列档次相关的附件/条款 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|---|
| O 信噪比和空域: 时域、信噪比和空域可分级— <i>EI</i> 和 <i>EP</i> 帧用于信噪比和空域可分级 | | | | | | | | | |
| P.5: 参考图像重采样—显式 4 因子 | | | | | | | | | X |
| P: 参考图像重采样—通用重采样 | | | | | | | | | |
| Q: 降低分辨率更新 | | | | | | | | | |
| R: 独立片断译码 | | | | | | | | | |
| S: 可选择的帧内变长编码 | | | | | | | | | |
| T: 修正的量化 | | X | | X | X | X | X | X | X |
| U 不含子模式: 增强的参考图像选择 (不含子模式) | | | | | | X | X | X | X |
| U 带子图像删除 (SPR): 增强的参考图像选择—带子图像删除子模式 | | | | | | | | | |
| U 带 B 帧 2 参考图像子模式 (BTSPM): 增强的参考图像选择—带 B 帧 2 参考图像子模式 | | | | | | | | | |
| V: 数据分割截面 | | | | | X | | | | |
| W.6.3.8: 附加的增补增强信息规范—先前图像头部重复 | | | | | X | | | | |
| W.6.3.11: 附加的增补增强信息规范—隔行场标识 | | | | | | | | X | |
| W: 附加的增补增强信息规范—其他 SEI 特性 | | | | | | | | | |
| <p>"X" 表示支持某个特性时某个档次的组成部分。</p> <p>"L" 表示包含某个特性取决于某档次下的级别。</p> | | | | | | | | | |

表 X.2/H.263—操作级别

| 右边所列级别的参数 | 10 | 20 | 30 | 40 | 45 | 50 | 60 | 70 |
|---|-----------------------|---|--|--|---|--|---|---|
| 最大图像格式 | QCIF (176 × 144) | CIF (352 × 288) | CIF (352 × 288) | CIF (352 × 288) | QCIF (176 × 144)在 0 和 2 之外的档次中支持 CPFMT | CIF (352 × 288) 支持 CPFMT | CPFMT: 720 × 288 支持 CPFMT | CPFMT: 720 × 576 支持 CPFMT |
| 最小图像间隔 | 2002/(30 000) 秒 | CIF: 2002/(30 000) 秒 QCIF 和 sub-QCIF : 1001/(30 000)秒 | 1001/(30 000) 秒 | 1001/(30 000) 秒 | 2002/(30 000) 秒 在 0 和 2 之外的档次中支持 CPCFC | CIF 或更小: 1/50 秒 352 × 240 或更小: 1001/(60 000) 秒 支持 CPCFC | 720 × 288 或更小: 1/50 秒 720 × 240 或更小: 1001/(60 000)秒 支持 CPCFC | 720 × 576 或更小: 1/50 秒 720 × 480 或更小: 1001/(60 000) 秒 support of CPCFC |
| 最大码率 (64 000 比特/秒) | 1 | 2 | 6 | 32 | 2 | 64 | 128 | 256 |
| 最大虚拟参考译码器缓存以16 384为单位 | 1: 档次 5-8 | 2: 档次 5-8 | 6: 档次 5-8 | 32: 档次 5-8 | 2: 档次 5-8 | 64: 档次 5-8 | 64: 档次 5-8 | 256: Prof. 5-8 |
| 最大BPPmaxKb (以1024比特为单位) | 128: 档次 5-8 | 512: 档次 5-8 | 512: 档次 5-8 | 512: 档次 5-8 | 128: 档次 5-8 | 512: 档次 5-8 | 1024: 档次 5-8 | 1024: 档次 5-8 |
| 最大ERPS参考图像 (附件U) | 5: 档次 5-7 10: 档次 8 | 5: 档次 5-7 10: 档次 8 对 QCIF 或 sub-QCIF,在档次 5-8 中乘以 2 | 5: 档次 5-7 10: 档次 8 对 QCIF 或 sub-QCIF,在档次 5-8 中乘以 2 | 5: 档次 5-7 10: 档次 8 对 QCIF 或 sub-QCIF,在档次 5-8 中乘以 2 | 5: 档次 5-7 10: 档次 8 | 5: 档次 5-7 10: 档次 8 对 QCIF 或更小格式,在档次 5-8 中乘以 2 | 5: 档次 5-7 10: 档次 8 multiplied by 2 for CIF or smaller, and by 4 for QCIF or smaller in 档次 5-8 | 5: 档次 5-7 10: 档次 8 档次 5-8 中, 对 CIF 或更小格式乘以 2,对 QCIF 或更小格式乘以 4 |
| 注 1 — 表 X.2 没有指定最大参考图像缓存的档次, 不要求支持多参考帧缓存。 | | | | | | | | |
| 注 2 — 表 X.2 没有指定 BPPmaxKb 和 HRD B 的档次, 则使用表 X.1 指定的最小值来确定最大比特率和分辨率。 | | | | | | | | |

X.5 使用ITU-T H.245建议书时的通用能力定义

表X.3定义了能力标识符，在使用ITU-T H.245建议书进行能力判决的系统中用来建立H.263能力。这些参数只能作为H.245中 **VideoCapability** 结构体中的 **genericVideoCapability**、**VideoMode** 结构体中的 **genericVideoMode** 出现。表X.4到X.14定义了相关的能力参数。

当包含在逻辑信道通告（Logical Channel Signalling）或模式请求（Mode Request）中时，参数标识符（Parameter identifier）的值在0-8之间的那个参数必须出现；即只能有一个档次被指定。

表 X.3/H.263—H.263能力的标识符

| | |
|-------------------|---|
| 能力名称: | H.263 |
| 能力类型: | 视频编解码器 |
| 能力标识符类型: | 标准 |
| 能力标识符的值: | itu-t (0) recommendation (0) h (8) 263 generic-capabilities (1) 0 |
| MaxBitRate: | 通常必须包含 maxBitRate 字段。 |
| NonCollapsingRaw: | 此字段必须包含。 |
| Transport: | 此字段必须包含。 |

表 X.4/H.263—基线档次（档次0）能力

| | |
|----------|---|
| 参数名称: | baselineProfile |
| 参数描述: | 这是一个可展开的 GenericParameter。 baselineProfile 指示基线档次在出现在能力交换（Capability Exchange）中支持的最大级别，在逻辑信道信令（Logical Channel Signalling）中的传输最大级别，以及在模式请求（Mode Request）中所希望的级别。 |
| 参数标识符的值: | 0 |
| 参数状态: | 强制 |
| 参数类型: | unsignedMin |
| 代替: | — |

表 X.5/H.263—H.320编码效率版本2后向兼容档次（档次1）能力

| | |
|----------|--|
| 参数名称: | h320Profile |
| 参数描述: | 这是一个可展开的 GenericParameter。 h320Profile 指示出 H.320 编码效率版本 2 后向兼容档次在能力交换（Capability Exchange）中支持的最大级别，出现在逻辑信道信令（Logical Channel Signalling）中传输最大级别，以及出现在模式请求（Mode Request）中所希望的级别。 |
| 参数标识符的值: | 1 |
| 参数状态: | 可选 |
| 参数类型: | unsignedMin |
| 代替: | — |

表 X.6/H.263—版本1后向兼容档次（档次2）能力

| | |
|----------|---|
| 参数名称: | backwardCompatibleProfile |
| 参数描述: | 这是一个可展开的 GenericParameter。 backwardCompatibleProfile 指示出版本 1 后向兼容档次在能力交换（Capability Exchange）中支持的最大级别，出现在逻辑信道信令（Logical Channel Signalling）中传输最大级别，以及出现在模式请求（Mode Request）中所期望的级别。 |
| 参数标识符的值: | 2 |
| 参数状态: | 可选 |
| 参数类型: | unsignedMin |
| 代替: | — |

表 X.7/H.263—版本2交互和流式无线档次（档次3）能力

| | |
|----------|--|
| 参数名称: | v2WirelessProfile |
| 参数描述: | 这是一个可展开的 GenericParameter。 v2WirelessProfile 指示出版本 2 交互和流式无线档次在能力交换（Capability Exchange）中支持的最大级别，出现在逻辑信道信令（Logical Channel Signalling）中传输最大级别，以及出现在模式请求（Mode Request）中所期望的级别。 |
| 参数标识符的值: | 3 |
| 参数状态: | 可选 |
| 参数类型: | unsignedMin |
| 代替: | — |

表 X.8/H.263—版本3交互和流式无线档次（档次4）能力

| | |
|----------|--|
| 参数名称: | v3WirelessProfile |
| 参数描述: | 这是一个可展开的 GenericParameter。 v3WirelessProfile 指示出版本 3 交互和流式无线档次在能力交换（Capability Exchange）中支持的最大级别，出现在逻辑信道信令（Logical Channel Signalling）中传输最大级别，以及出现在模式请求（Mode Request）中所期望的级别。 |
| 参数标识符的值: | 4 |
| 参数状态: | 可选 |
| 参数类型: | unsignedMin |
| 代替: | — |

表 X.9/H.263—会话高压压缩档次（档次5）能力

| | |
|----------|---|
| 参数名称: | conversationalProfile |
| 参数描述: | 这是一个可展开的 GenericParameter。 conversationalProfile 指示会话高压压缩档次在能力交换（Capability Exchange）中支持的最大级别，出现在逻辑信道信令（Logical Channel Signalling）中传输最大级别，以及出现在模式请求（Mode Request）中所期望的级别。 |
| 参数标识符的值: | 5 |
| 参数状态: | 可选 |
| 参数类型: | unsignedMin |
| 代替: | — |

表 X.10/H.263—会话因特网档次（档次6）能力

| | |
|----------|--|
| 参数名称: | conversationalInternetProfile |
| 参数描述: | 这是一个可展开的 GenericParameter。 conversationalInternetProfile 指示会话因特网档次在能力交换（Capability Exchange）中支持的最大级别，出现在逻辑信道信令（Logical Channel Signalling）中传输最大级别，以及出现在模式请求（Mode Request）中所期望的级别。 |
| 参数标识符的值: | 6 |
| 参数状态: | 可选 |
| 参数类型: | unsignedMin |
| 代替: | — |

表 X.11/H.263—会话交织档次（档次7）能力

| | |
|----------|--|
| 参数名称: | conversationalInterlaceProfile |
| 参数描述: | 这是一个可展开的 GenericParameter。 conversationalInterlaceProfile 指示会话交织档次在能力交换（Capability Exchange）中支持的最大级别，出现在逻辑信道信令（Logical Channel Signalling）中传输最大级别，以及出现在模式请求（Mode Request）中所期望的级别。 |
| 参数标识符的值: | 7 |
| 参数状态: | 可选 |
| 参数类型: | unsignedMin |
| 代替: | — |

表 X.12/H.263—高延迟档次（档次8）能力

| | |
|----------|--|
| 参数名称: | highLatencyProfile |
| 参数描述: | 这是一个可展开的 GenericParameter。 highLatencyProfile 示高延迟档次（档次 8）在能力交换（Capability Exchange）中支持的最大级别，出现在逻辑信道信令（Logical Channel Signalling）中传输最大级别，以及出现在模式请求（Mode Request）中所期望的级别。 |
| 参数标识符的值: | 8 |
| 参数状态: | 可选 |
| 参数类型: | unsignedMin |
| 代替: | — |

表 X.13/H.263—时域空域折中能力

| | |
|----------|--|
| 参数名称: | temporalSpatialTradeOffCapability |
| 参数描述: | 这是一个可展开的 GenericParameter。 此参数的出现指示出编码器能够在远程终端命令控制之下改变其时域和空域分辨率之间的平衡。作为接收能力的一部分时没有意义。 |
| 参数标识符的值: | 9 |
| 参数状态: | 可选 |
| 参数类型: | 逻辑 |
| 代替: | — |

表 X.14/H.263—视频损坏宏块能力

| | |
|----------|---|
| 参数名称: | videoBadMBsCap |
| 参数描述: | 这是一个可展开的 GenericParameter。 此参数的出现指示出编码器（或译码器）具有接收（或发送）videoBadMBs 的能力。作为传送能力的一部分时，它表示编码器处理 videoBadMBs 命令并采取适当动作恢复视频质量的能力。当作为接收能力的一部分时，表示译码器发送适当 videoBadMBs 标识符的能力。 |
| 参数标识符的值: | 10 |
| 参数状态: | 可选 |
| 参数类型: | 逻辑 |
| 代替: | — |

附录 I

错误跟踪

I.1 概述

本附录描述一种有效恢复的方法，传输后如果通过到译码器的反馈信道报告有错误的MB时，使用本附录定义的方法。发送和处理反馈信息的能力通过外部手段传递（例如，ITU-T H.245建议书）。此外，反馈信息的格式和内容也通过外部方式定义（例如，ITU-T H.245建议书）。

I.2 错误跟踪

由于INTRA编码可以阻止时域错误传播，它应该用到受传输错误严重影响的宏块上。这要求复制图像的位置和范围对译码器来说是可用的。如下算法提供一个基于编码器收到的反馈信息的错误分布估计。该算法考虑运动补偿预测引起的空间错误传播，以及接收反馈信息引起的延迟。该算法给出了估计空间一时间错误跟踪反馈信息的一个可能的方法。其他算法也是可行的。

假设有N个宏块，帧编号从左上到右下依次为 $mb = 1 \dots N$ 。 $\{n_{err}, mb_{first}, mb_{last}\}$ 为译码器收到的反馈信息，这里 $mb_{first} \leq mb \leq mb_{last}$ 是指帧 n_{err} 中的一系列错误宏块。

为了改进反馈消息，译码器必须持续记录每个帧编码的信息。首先，需要存储初始错误 $E_0(mb, n)$ ，该错误是由于帧n的宏块mb的丢失造成的。假设一个简单错误隐藏的情况，错误宏块作为未编码处理， $E_0(mb, n)$ 等于帧n和n-1中的宏块mb的总的绝对差值（SAD）。第二，从帧n-1中的宏块 mb_{source} 到帧n宏块 mb_{dest} 中传输的像素数存储在相关函数 $d(mb_{source}, mb_{dest}, n)$ 。这些相关函数从运动矢量取得。

假设由帧 n_{next} 取得的反馈信息已经编码，例如 $n_{next} > n_{err}$ 。然后，宏块 mb 中的估计错误 $E(mb, n_{err})$ 和帧 n_{err} 初始化如下：

$$E(mb, n_{err}) = \begin{cases} E_0(mb, n_{err}) & mb_{first} \leq mb \leq mb_{last} \\ 0 & \text{其他} \end{cases}$$

对于后续帧 n ， $n_{err} < n < n_{next}$ ，错误可以按照如下公式进行计算：

$$E(mb, n) = \sum_{i=1}^N E(i, n-1) \frac{d(i, mb, n)}{256}$$

这里假设每个宏块中都有同样分布的错误。

将估计错误 $E(mb, n_{next} - 1)$ 合并到下一帧的模式决定中。例如，如果 $E(mb, n_{next} - 1)$ 超出一个门限值，宏块 mb 以INTRA模式编码。

实际上，错误跟踪信息将只存在于最后一个M帧中。这样，如果 $n_{err} < n_{next} - M$ ，就不会有可用的错误信息，编码器必须采取特定操作。例如，下一帧可能以INTRA模式编码。然而，其他过程也是可以的，并且可能效率更高。

附录 II

推荐的可选增强

H.263附录II的内容于1998年2月份通过，在H.263附件X通过之后，附录II的内容已经变为了独立的一部分。由于该附录被很多ITU-T文本的用户用作参考，因此该附录现在主要作为用户参考，而与H.263附件X的内容无关。

ITU-T 系列建议书

| | |
|------------|-------------------------|
| A系列 | ITU-T工作的组织 |
| D系列 | 一般资费原则 |
| E系列 | 综合网络运行、电话业务、业务运行和人为因素 |
| F系列 | 非话电信业务 |
| G系列 | 传输系统和媒质、数字系统和网络 |
| H系列 | 视听和多媒体系统 |
| I系列 | 综合业务数字网 |
| J系列 | 有线网和电视、声音节目和其他多媒体信号的传输 |
| K系列 | 干扰的防护 |
| L系列 | 线缆的构成、安装和保护及外部设备的其他组件 |
| M系列 | 电信管理，包括TMN和网络维护 |
| N系列 | 维护：国际声音节目和电视传输电路 |
| O系列 | 测量设备技术规程 |
| P系列 | 电话传输质量、电话装置和本地线路网络 |
| Q系列 | 交换和信令 |
| R系列 | 电报传输 |
| S系列 | 电报业务终端设备 |
| T系列 | 远程信息处理业务的终端设备 |
| U系列 | 电报交换 |
| V系列 | 电话网上的数据通信 |
| X系列 | 数据网和开放系统通信及安全 |
| Y系列 | 全球信息基础设施、互联网的协议问题和下一代网络 |
| Z系列 | 用于电信系统的语言和一般软件问题 |