

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

H03M 7/40 (2006.01)

H04N 7/50 (2006.01)



[12] 发明专利说明书

专利号 ZL 03812229.4

[45] 授权公告日 2009 年 7 月 29 日

[11] 授权公告号 CN 100521550C

[22] 申请日 2003.4.2 [21] 申请号 03812229.4

[30] 优先权

[32] 2002. 4. 2 [33] US [31] 60/369,500

[86] 国际申请 PCT/FI2003/000253 2003.4.2

[87] 国际公布 WO2003/084076 英 2003.10.9

[85] 进入国家阶段日期 2004.11.29

[73] 专利权人 诺基亚有限公司

地址 芬兰埃斯波

[72] 发明人 R·库尔切伦 M·卡策维茨

[56] 参考文献

US6166664A 2000.12.26

US6023296A 2000.2.8

New Results on Improved CABAC. Detlev Marpe. Joint Video Team (JVT) of ISO/IEC MPEG & ITU. T VCEG 2nd Meeting. 2002

审查员 王 艳

[74] 专利代理机构 北京市金杜律师事务所

代理人 冯 谱

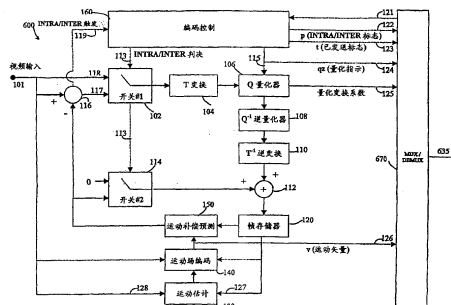
权利要求书 7 页 说明书 32 页 附图 7 页

[54] 发明名称

用于数字视频的编码方法、解码方法、编码器和解码器

[57] 摘要

本发明涉及用于将包括零值数据符号和非零值数据符号的数据符号集编码的方法，所述方法包括通过执行映射操作指配码字以表示非零值数据符号及相关联的若干在前或在后的零值数据符号。根据本发明，将一个码字指配给非零值数据符号及其相关联的若干在前或在后的零值数据符号的映射操作依赖于所述数据符号集中零值数据符号的最大可能数量。本发明还描述了对应的解码方法及编码和解码装置。



1. 一种将包括零值数据符号和非零值数据符号的数据符号集编码的方法，它包括：

- 通过映射操作指配码字以表示非零值数据符号及相关联的若干在前或在后的零值符号，其特征在于：将一个码字指配给非零值数据符号及其相关联的若干在前或在后的零值数据符号的所述映射操作依赖于所述数据符号集中零值数据符号的最大可能数量，并且其特征在于：结合将非零值数据符号及其相关联的若干在前或在后的零值数据符号指配给一个码字，通过从零值数据符号的所述最大可能数量中减去所述相关联的若干在前或在后的零值数据符号的数量以得到零值数据符号的更新的所述最大可能数量，从而更新所述数据符号集中零值数据符号的所述最大可能数量，其中，借以将后续码字指配给非零值数据符号及其相关联的若干在前或在后的零值数据符号的所述映射操作依赖于零值数据符号的所述更新的所述最大可能数量。

2. 如权利要求 1 所述的编码方法，其特征在于：通过从所述数据符号集中数据符号的总数中减去所述数据符号集中非零值数据符号的数量，确定所述数据符号集中零值数据符号数量的所述最大可能数量。

3. 如权利要求 1 或 2 所述的编码方法，其特征在于：将所述数据符号集中非零值数据符号的所述数量指示发送到对应的解码装置。

4. 如权利要求 1 或 2 所述的编码方法，其特征在于：根据所述数据符号集中零值数据符号的所述最大可能数量从映射表集选择特定的映射表来执行所述映射操作，所述映射表集的每个映射表定义数据符号到码字的指配关系。

5. 如权利要求 4 所述的编码方法，其特征在于：每个所述映射表定义非零值数据符号及其相关联的若干在前或在后的零值数据符号到码字的指配关系。

6. 如权利要求 4 所述的编码方法, 其特征在于: 每个所述映射表定义非零值数据符号及其相关联的若干在前或在后的零值数据符号到码号的指配关系, 每个码号表示码字集中的一个码字, 并可用作所述码字集的引用, 由此定义非零值数据符号及其相关联的若干在前或在后的零值数据符号到码字的指配关系。

7. 如权利要求 1 或 2 所述的编码方法, 其特征在于: 所述数据符号集表示变换系数值。

8. 如权利要求 1 或 2 所述的编码方法, 其特征在于: 先前编码的非零值数据符号的值用于从包含码字的码本集中指定一个码本, 以将当前非零值数据符号编码。

9. 如权利要求 1 或 2 所述的编码方法, 其特征在于: 所述方法用于视频编码器中。

10. 一种将表示数据符号集的码字集解码的方法, 所述数据符号集包括零值数据符号和非零值数据符号, 所述方法包括:

- 接收所述数据符号集中表示若干非零值数据符号的指示;
- 接收表示非零值数据符号及其相关联的若干在前或在后的零值数据符号的码字;
- 通过执行逆映射操作, 将所述码字解码以恢复所述非零值数据符号及其相关联的若干在前或在后的零值数据符号;

其特征在于: 所述逆映射操作依赖于所述数据符号集中零值数据符号的最大可能数量; 通过所述逆映射操作, 可从所述码字将非零值数据符号及其相关联的若干在前或在后的零值数据符号解码, 并且其特征在于: 结合将接收的码字解码以恢复由所述码字表示的所述非零值数据符号及其相关联的若干在前或在后的零值数据符号, 通过从零值数据符号的所述最大可能数量中减去所述相关联的若干在前或在后的零值数据符号的数量以得到零值数据符号的更新的所述最大可能数量, 从而更新所述数据符号集中零值数据符号的所述最大可能数量, 所述逆映射操作依赖于零值数据符号的所述更新的所述最大可能数量, 通

过所述逆映射操作，将后续码字解码以恢复所述后续码字表示的所述非零值数据符号及其相关联的若干在前或在后的零值数据符号。

11. 如权利要求 10 所述的解码方法，其特征在于：通过从所述数据符号集的数据符号的总数中减去所述数据符号集中非零值数据符号的数量，确定所述数据符号集中零值数据符号的所述最大可能数量。

12. 如权利要求 10 或 11 所述的解码方法，其特征在于：所述数据符号集中非零值数据符号的数量指示是从对应的编码装置接收的。

13. 如权利要求 10 或 11 所述的解码方法，其特征在于：根据所述数据符号集中零值数据符号的所述最大可能数量从逆映射表集中选择特定的逆映射表，从而执行所述逆映射操作，所述逆映射表集的每个逆映射表定义码字到数据符号的指配关系。

14. 如权利要求 13 所述的解码方法，其特征在于：每个所述逆映射表定义码字到非零值数据符号及其相关联的若干在前或在后的零值数据符号的指配关系。

15. 如权利要求 13 所述的解码方法，其特征在于：对所述接收的码字进行解码以恢复码号，所述码号用作根据所述数据符号集中零值数据符号的所述最大可能数量选择的特定逆映射表的引用，由此恢复由所述码字表示的所述非零值数据符号及其相关联的若干在前或在后的零值数据符号。

16. 如权利要求 10 或 11 所述的解码方法，其特征在于：所述数据符号集表示变换系数值。

17. 如权利要求 10 或 11 所述的解码方法，其特征在于：先前解码的非零值数据符号的值用于从包含表示非零值数据符号的码字的码本集中选择一个码本，以将当前码字解码。

18. 如权利要求 10 或 11 所述的解码方法，其特征在于：所述方法用于视频解码器中。

19. 一种用于将包括零值数据符号和非零值数据符号的数据符号

集编码的编码器，它包括：

- 通过执行映射操作指配码字，以表示非零值数据符号及其相关联的若干在前或在后的零值符号的部件，其特征在于：所述部件用于指配码字的所述部件设置来调整所述映射操作，通过所述映射操作，根据所述数据符号集中零值数据符号的最大可能数量来指配码字，及用于结合将非零值数据符号及其相关联的若干在前或在后的零值数据符号指配给一个码字，通过从零值数据符号的所述最大可能数量中减去所述相关联的若干在前或在后的零值数据符号的数量以得到零值数据符号的更新的所述最大可能数量，从而更新所述数据符号集中零值数据符号的所述最大可能数量的部件，且该部件用于指配码字的所述部件设置来调整所述映射操作，通过所述映射操作，根据零值数据符号的所述更新的所述最大可能数量来将后续码字指配给非零值数据符号及其相关联的若干在前或在后的零值数据符号。

20. 如权利要求 19 所述的编码器，其特征在于：它设置来通过从所述数据符号集的数据符号总数中减去所述数据符号集中非零值数据符号的数量，从而确定所述数据符号集中零值数据符号的所述最大可能数量。

21. 如权利要求 19 或 20 所述的编码器，其特征在于：它包括用于将所述数据符号集中非零值数据符号数量指示发送到对应解码装置的部件。

22. 如权利要求 19 或 20 所述的编码器，其特征在于：用于指配码字的所述部件设置来根据所述数据符号集中零值数据符号的所述最大可能数量从映射表集中选择特定的映射表，从而执行所述映射操作，所述映射表集的每个映射表定义数据符号到码字的指配关系。

23. 如权利要求 22 所述的编码器，其特征在于：每个所述映射表定义非零值数据符号及其相关联的若干在前或在后的零值数据符号到码字的指配关系。

24. 如权利要求 22 所述的编码器，其特征在于：每个所述映射

表定义非零值数据符号及其相关联的若干在前或在后的零值数据符号到码号的指配关系，每个码号表示码字集中的一个码字，其中：指配码字的所述部件用于将所述码号用作到所述码字集的引用，由此定义非零值数据符号及其相关联的若干在前或在后的零值数据符号到码字的指配关系。

25. 如权利要求 19 或 20 所述的编码器，其特征在于：所述数据符号集表示变换系数值。

26. 如权利要求 19 或 20 所述的编码器，其特征在于：先前编码的非零值数据符号的值用于从包含码字的码本集中指定一个码本，以将当前非零值数据符号编码，其中：指配码字的所述部件用于将所述先前编码的非零值数据符号的所述值用作到所述码本集的引用。

27. 如权利要求 19 或 20 所述的视频编码器。

28. 一种用于将表示数据符号集的码字集解码的解码器，所述数据符号集包括零值数据符号和非零值数据符号，所述解码器包括：

- 用于接收所述数据符号集中表示非零值数据符号数量指示的部件；
- 用于接收表示非零值数据符号及其相关联的若干在前或在后的零值数据符号的码字的部件；
- 用于通过执行逆映射操作，将码字解码以恢复所述非零值数据符号及其相关联的若干在前或在后的零值数据符号的部件；

其特征在于：所述部件用于将所述码字解码的所述部件设置来调整所述逆映射操作，通过所述逆映射操作，根据所述数据符号集中零值数据符号的最大可能数量，将非零值数据符号及其相关联的若干在前或在后的零值数据符号解码，及用于结合将接收的码字解码以恢复所述码字所表示的所述非零值数据符号及其相关联的若干在前或在后的零值数据符号，通过从零值数据符号的所述最大可能数量中减去所述相关联的若干在前或在后的零值数据符号的数量以得到零值数据符号的更新的所述最大可能数量，从而更新所述数据符号集中零值数据

符号的所述最大可能数量的部件；该部件用于将码字解码的所述部件设置来调整所述逆映射操作，通过所述逆映射操作，根据零值数据符号的所述更新的所述最大可能数量，将后续码字解码以恢复所述后续码字表示的所述非零值数据符号及其相关联的若干在前或在后的零值数据符号。

29. 如权利要求 28 所述的解码器，其特征在于：它设置来通过从所述数据符号集的数据符号的总数中减去所述数据符号集中非零值数据符号的数量，确定所述数据符号集中零值数据符号的所述最大可能数量。

30. 如权利要求 28 或 29 所述的解码器，其特征在于：它包括用于从对应的编码装置接收所述数据符号集中非零值数据符号的所述数量指示的部件。

31. 如权利要求 28 或 29 所述的解码器，其特征在于：用于将码字解码的所述部件设置来根据所述数据符号集中零值数据符号的所述最大可能数量从逆映射表集中选择特定的逆映射表，执行所述逆映射操作，所述逆映射表集的每个逆映射表定义码字到数据符号的指配关系。

32. 如权利要求 31 所述的解码器，其特征在于：每个所述逆映射表定义码字到非零值数据符号及其相关联的若干在前或在后的零值数据符号的指配关系。

33. 如权利要求 31 所述的解码器，其特征在于：用于将码字解码的所述部件设置来将接收的码字解码，以恢复码号并将所述码号用作根据所述数据符号集中零值数据符号的所述最大可能数量而选择的所述特定逆映射表的引用，由此恢复所述码字表示的所述非零值数据符号及其相关联的若干在前或在后的零值数据符号。

34. 如权利要求 28 或 29 所述的解码器，其特征在于：所述数据符号集表示变换系数值。

35. 如权利要求 28 或 29 所述的解码器，其特征在于：用于将码

字解码的所述部件设置来利用先前解码的非零值数据符号的所述值，从包含表示非零值数据符号的码字的码本集中选择一个码本，以将当前码字解码。

36. 如权利要求 28 或 29 所述的解码器。

37. 一种多媒体终端，它包括如权利要求 27 所述的视频编码器。

38. 一种多媒体终端，它包括如权利要求 36 所述的视频解码器。

39. 如权利要求 37 或 38 所述的多媒体终端，其特征在于：它包括用于形成到无线电通信网的无线链路的部件。

用于数字视频的编码方法、解码方法、编码器和解码器

技术领域

本发明涉及用于数字视频编码的方法、编码器、解码器和装置。更具体地说，本发明涉及使用变长编码（VLC），对在视频/图像编码器/解码器中由块变换编码产生的量化变换系数编码。

背景技术

诸如在胶片上录制的普通动画等数字视频序列包括一序列静态图像，运动错觉是通过以相对快的速率，一般为每秒 15 到 30 帧的速率，连续显示图像而产生的。由于显示速率相对较快，因此，连续帧中的图像往往很相似，因此包含了大量冗余信息。例如，一个典型的场面可包括一些静止元素如背景景物和一些移动区域，这些区域可能有许多不同形式，例如，新闻播报员的面孔、移动的车流等等。或者，例如，录制场景的摄像机本身可能移动，在这种情况下，图像的所有元素具有同类的运动。在许多情况下，这表示一个视频帧与下一视频帧之间的整体变化相当小。

未压缩数字视频序列的每个帧包括图像像素阵列。例如，在通称为四分之一公共交换格式（QCIF）的数字视频格式中，一个帧包括 176×144 个像素的阵列，在这种情况下，每个帧具有 25344 个像素。每个像素又由一定数量的比特表示，这些比特携带与像素对应的图像区域的亮度和/或色度有关的信息。通常，所谓的 YUV 彩色模型用于表示图像的亮度和色度内容。亮度即 Y 分量表示图像的强度（亮度），而图像的颜色内容由标记为 U 和 V 的两种色度或色差分量表示。

与基于与表示有关的基色（即红、绿和蓝 RGB）的彩色模型相比，基于图像内容亮度/色度表示的彩色模型具有某些优点。人的视觉系统

对强度变化比对颜色变化更敏感,因此,YUV 彩色模型利用了此属性,对色度分量(U、V)使用了比对亮度分量(Y)更低的空间分辨率。这样,可以减少将图像中颜色信息编码所需的信息量,同时具有可接受的图像质量下降。

色度分量的更低空间分辨率通常通过二次抽样实现。一般情况下,视频序列的每个帧划分成所谓的“宏块”,它包括空间上二次抽样的亮度(Y)信息和相关联的色度(U、V)信息。图3显示了可以形成宏块的一种方式。图3a显示了使用YUV彩色模型表示的视频序列的一个帧,每个分量具有相同的空间分辨率。宏块是通过将原始图像(图3b)中的 16×16 个图像像素区域表示为4个亮度信息块而形成的,每个亮度块包括亮度(Y)值和两个空间对应的色度分量(U和V)的 8×8 阵列,两个色度值在x和y方向上根据因数2进行二次抽样,以产生对应的 8×8 色度(U、V)值阵列(参见图3c)。根据某些视频编码建议,如国际电信联盟(ITU-T)建议H.26L,在宏块内使用的数据块大小可以不同于 8×8 ,例如,可以是 4×8 或 4×4 (参阅Joint Model (JM) 2.0 工作草案编号2[Joint Model (JM) 2.0 Working Draft Number 2, Doc. JVT-B118, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG Pattaya, 2nd JVT Meeting, Geneva, CH, Jan. 29 – Feb. 1, 2002, Section 3.2])。

QCIF 图像包括 11×9 个宏块。如果亮度块和色度块用8比特分辨率(即,由0到255之间的数字)表示,则每个宏块所需的总比特数为 $(16 \times 16 \times 8) + 2 \times (8 \times 8 \times 8) = 3072$ 比特。因此,表示QCIF格式的一个视频帧所需的比特数量为 $99 \times 3072 = 304128$ 个比特。这意味着要以每秒30帧的速率发送/录制/显示以YUV彩色模型表示的QCIF格式的未压缩视频序列,则所需数据量超过9Mbps(每秒百万比特)。这是极高的数据率,由于需要非常大的存储容量、传输信道容量和极佳的硬件性能而实际上不适用于视频录制、传输和显示应用。

如果要通过诸如ISDN(综合业务数字网络)或常规PSTN(公共

交换电话网)等固定线路网络实时传送视频数据,则可用的数据传输带宽一般约为每秒 64 千比特。在移动视频电话中,至少部分传输要通过无线电通信链路进行,可用的带宽可能低到每秒 20 千比特。这意味着必须使得表示视频数据所需的信息量大量减少才可以通过低带宽通信网络传输数字视频序列。为此,开发了视频压缩技术以减少发送的信息量,同时保持可接受的图像质量。

视频压缩方法基于减少视频序列的冗余的且感觉上不相关的部分。视频序列的冗余可分为空间、时间和频谱冗余。“空间冗余”是用于描述帧内相邻像素之间相关性(相似度)的术语。术语“时间冗余”表示在序列的一个帧中出现的对象可能在随后的帧中出现的事实;而“频谱冗余”指同一图像不同颜色分量之间的相关性。

简通过单地降低给定图像序列中不同形式的冗余,通常无法实现充分有效的压缩。因此,大多数当前的视频编码器还降低了主观上最不重要的那些视频序列部分的质量。另外,通过有效的无损编码,降低了压缩视频经特流本身的冗余。通常,这通过使用称为熵编码的技术实现。

在组成每个数字视频序列帧的像素之间,经常有大量空间冗余。换言之,序列的帧内任何像素的值与其紧密相邻的其它像素的值大致是相同的。一般情况下,视频编码系统使用称为“块变换编码”技术降低空间冗余,在该技术中,将诸如二维离散余弦变换(DCT)的数学变换应用于图像像素块上。这将图像数据从包括像素值的表示变换为包括一组表示空间频率分量的系数值的形式,极大地降低了空间冗余,由此产生了更简洁的图像数据表示。

使用块变换编码压缩,而未引用序列内其它任何帧的视频序列帧称为 INTRA 编码帧或 I 帧。此外,并且在可能时,从同一帧内的先前的编码块预测 INTRA 编码帧块。这种称为 INTRA 预测的技术可进一步降低表示 INTRA 编码帧所需的数据量。

通常,视频编码系统不仅降低了视频序列各个帧内的空间冗余,

而且利用称为“运动补偿预测”的技术来降低序列中的时间冗余。通过使用运动补偿预测，数字视频序列中一些（经常是许多）帧的图像内容是从序列中称为“参考”帧的一个或多个其它帧“预测”的。通过使用“运动矢量”在要编码（压缩）的帧与参考帧之间跟踪图像对象或区域的运动，可实现对图像内容的预测。通常，参考帧可在视频序列中要编码的帧之前，或者可在它之后。与在 INTRA 编码中的情况一样，视频帧的运动补偿预测一般是逐宏块执行的。

使用运动补偿预测压缩的视频序列帧通常称为 INTER 编码帧或 P 帧。单独的运动补偿预测很少提供足够精确的视频帧图像内容表示，因此，通常有必要为每个 INTER 编码帧提供称为“预测误差”（PE）帧。预测误差帧表示 INTER 编码帧的解码版本与要编码的帧图像内容之间的差异。更具体地说，预测误差帧包括表示要编码的帧中像素值与基于所述帧的预测版本形成的对应重建像素值之间的差异的值。因此，预测误差帧具有类似于静态图像的特性，并且可以应用块变换编码以降低其空间冗余，并因此降低表示它所需的数据量（比特数量）。

为更详细地说明视频编码系统的操作，现在参照图 1 和图 2。图 1 是通用视频编码器的示意图，它采用 INTRA 和 INTER 编码组合以产生压缩（编码）的视频比特流。稍后将描述图 2 显示的对应解码器。

视频编码器 100 包括用于从摄像机或其它视频源（未显示）接收数字视频信号的输入 101。它还包括用于执行基于块的离散余弦变换（DCT）的变换单元 104、量化器 106、逆量化器 108、用于执行基于块的逆向离散余弦变换的逆变换单元 110、合并器 112 和 116 及帧存储器 120。编码器还包括运动估计器 130、运动场编码器（coder）140 和运动补偿预测器 150。开关 102 和 114 由控制管理器 160 操作共同用于使编码器在视频编码的 INTRA 模式与视频编码的 INTER 模式之间切换。编码器 100 还包括视频复用编码器 170，该编码器根据编码器 100 产生的不同类型信息形成一个比特流，以便进一步传输到远程接收端，或者例如，以便存储在诸如计算机硬盘驱动器（未显示）等

海量存储介质上。

编码器 100 的操作如下。接收从视频源提供给输入 101 的每个未压缩视频帧并逐宏块进行处理，最好是以光栅扫描顺序处理。在新的视频序列编码开始时，将要编码的第一个帧编码为 INTRA 编码帧。随后，除非满足以下条件之一，否则编码器设计为以 INTER 编码格式将每个帧编码：1) 断定正在被编码的帧的当前宏块与在其预测中使用的参考帧中的像素值如此不同，以致产生过量的预测误差信息，在这种情况下，当前宏块以 INTRA 编码格式编码；2) 预定的 INTRA 帧重复间隔已满期；或者 3) 从接收终端收到表示以 INTRA 编码格式提供帧的请求的反馈。

通过监视合并器 116 的输出可检测条件 1) 的出现。合并器 116 形成在编码的帧的当前宏块与其在运动补偿预测功能块 150 中产生的预测之间的差异。如果此差异的测量值（例如，像素值的绝对差之和）超过预定阈值，则合并器 116 经控制线 119 通知控制管理器 160，并且控制管理器 160 经控制线 113 操作开关 102 和 114，以便将编码器 100 切换到 INTRA 编码模式。这样，本来要以 INTER 编码格式编码的帧可包括 INTRA 编码宏块。条件 2) 的出现通过实现于控制管理器 160 中的计时器或帧计数器来监视，使得如果计时器到期，或者帧计数器达到预定数量的帧，则控制管理器 160 经控制线 113 操作开关 102 和 114，将编码器 100 切换到 INTRA 编码模式。如果控制管理器 160 例如经控制线 121 从接收终端收到指示接收终端需要 INTRA 帧刷新的反馈信号，就触发条件 3)。例如，如果先前发送的帧因其传输期间的干扰而被严重损坏，使得无法在接收机上将其解码，则此类条件可发生。在这种情况下，接收解码器发出请求，请求以 INTRA 编码格式对下一帧编码，从而重新初始化编码序列。

现在将描述 INTRA 编码模式下编码器 100 的操作。在 INTRA 编码模式，控制管理器 160 操作开关 102 以接受来自输入线 108 的视频输入。通过输入线 118 从输入 101 逐宏块接收视频信号输入。在接收

它们时，组成宏块的亮度和色度值块传递到 DCT 变换块 104，DCT 变换块 104 对每个值块执行二维离散余弦变换，每个数据块产生一个二维 DCT 系数阵列。DCT 变换块 104 为每个数据块产生系数值阵列，系数值的数量取决于组成宏块的块的性质。例如，如果在宏块中使用的基本块大小为 4×4 ，则 DCT 变换块 104 为每个数据块产生 4×4 的 DCT 系数阵列。如果数据块大小为 8×8 ，则产生 8×8 的 DCT 系数阵列。

每个块的 DCT 系数传递到量化器 108，在该量化器中使用量化参数 QP 对它们进行量化。量化参数 QP 的选择由控制管理器 160 经控制线 115 控制。由于量化的系数比 DCT 变换块 104 原来生成的系数具有更低的数值精度，因此，量化引入了信息损伤。这提供了又一种机制，通过该机制可减少表示视频序列的每个图像所需的数据量。然而，不同于基本上无损的 DCT 变换，量化引起的信息损伤使图像质量不可逆地降低。对 DCT 系数施加的量化度越大，则图像质量损伤越大。

如图 1 线 125 所示，每个块的量化 DCT 系数从量化器 106 传递到视频复用编码器 170。视频复用编码器 170 使用之字形扫描过程将每个块的量化变换系数排序。此操作将量化变换系数的二维阵列变换成一维阵列。诸如用于图 4 所示 4×4 阵列的典型的之字形扫描顺序近似按空间频率的升序将系数排序。这往往还根据系数的值将系数排序，使得位于一维阵列更前面的系数更可能比位于该阵列后面的系数具有更大的绝对值。这是因为较低的空间频率往往在图像块内具有更高的幅度。因此，量化变换系数一维阵列中最后的值通常为零。

一般情况下，视频复用编码器 170 用两个值表示一维阵列中的每个非零量化系数，称为 level（级别）和 run（游程）。Level 是量化系数的值，run 是在所述系数前连续零值系数的数量。将给定系数的 run 和 level 值排序，使得 level 值在相关联的 run 值前面。等于零的 level 值用于指示数据块中不再有其它非零系数值。此 0 级别值称为 EOB（块

结束)符号。

或者,一维阵列中每个非零值量化系数可由三个值(run、level、last)表示。Level和run的意义与前一方案相同,而last指示在一维阵列中当前系数后是否有其它任何非零值系数。因此,无需单独的EOB符号。

使用熵编码对Run和level值(及适用情况下的last)作进一步的压缩。熵编码是无损运算,它利用了要编码的数据集内的符号一般具有不同的出现概率的事实。因此,可不使用固定数量的比特表示每个符号,而是分配可变数量的比特,以使用具有更少比特的码字表示更可能出现的符号。为此,熵编码经常称为变长编码(VLC)。由于level和run的某些值比其它值更可能出现,因此,熵编码技术可有效地用于减少表示run和level值所需的比特数量。

一旦run和level值已经过熵编码,视频复用编码器170就将其与控制信息组合在一起,以形成编码图像信息135的单个压缩比特流,其中所述控制信息也采用适合所述信息种类的变长编码方法进行了熵编码。从编码器发送的正是此比特流,它包括了表示(run、level)对的变长码字。

虽然结合视频复用编码器170执行的操作对熵编码进行了描述,但应注意的是,在替代实施中可提供单独的熵编码单元。

编码器100中还形成了本地解码的宏块版本。这是通过使量化器106输出的每个块的量化变换系数通过逆量化器108,并在逆变换块110中施加逆DCT变换而实现的。这样,构造了对应宏块的每个块的重建的像素值阵列。所得的解码图像数据输入合并器112。在INTRA编码模式下,设置开关114以使经开关114到合并器112的输入为零。这样,合并器112执行的操作等同于不加变化地传递解码图像数据。

随着当前帧的后续宏块被接收,并在块104、106、108、110和112中经历如前所述的编码和本地解码步骤,于是在帧存储器120中建立了INTRA编码帧的解码版本。在当前帧的最后一个宏块已经过

INTRA 编码, 随后又经解码后, 帧存储器 120 包含了完整的解码帧, 该帧可以在将随后接收的视频帧以 INTER 编码格式编码的过程中用作预测参考帧。

现在将描述 INTER 编码模式下编码器 100 的操作。在 INTER 编码模式, 控制管理器 160 操作开关 102 以从线路 117 接收其输入, 它包括合并器 116 的输出。合并器 116 逐宏块从输入 101 接收视频输入信号。合并器 116 收到组成宏块的亮度和色度值块时, 就形成对应的预测误差信息块。预测误差信息表示所述块及在运动补偿预测功能块 150 中产生的其预测之间的差异。更具体地说, 宏块的每个块的预测误差信息包括值的二维阵列, 每个值表示在编码的亮度或色度信息块中的像素值与根据下述过程通过形成数据块的运动补偿预测所获得的解码像素值之间的差异。因此, 在每个宏块包括例如 4×4 块集合 (包括亮度和色度值) 的情况下, 宏块的每个块的预测误差信息同样包括 4×4 的预测误差值阵列。

宏块的每个块的预测误差信息传递到 DCT 变换块 104, DCT 变换块 104 对每个预测误差值块执行二维离散余弦变换, 为每个块产生 DCT 变换系数的二维阵列。DCT 变换块 104 为每个预测误差块产生系数值阵列, 系数值的数量取决于组成宏块的数据块的性质。例如, 如果在宏块中使用的基本数据块大小为 4×4 , 则 DCT 变换块 104 为每个预测误差块生成的 4×4 DCT 系数阵列。如果数据块大小为 8×8 , 则产生 8×8 的 DCT 系数阵列。

每个预测误差块的变换系数传递到量化器 106, 在该量化器中, 使用量化参数 QP, 以类似于以上结合 INTRA 编码模式下的编码器操作所述的方式, 对它们进行量化。同样地, 量化参数 QP 的选择由控制管理器 160 经控制线 115 控制。

如图 1 中线路 125 所示, 表示宏块的每个块的预测误差信息的量化 DCT 系数从量化器 106 传递到视频复用编码器 170。与在 INTRA 编码模式下一样, 视频复用编码器 170 使用上述之字形扫描过程 (参

见图 4) 将每个预测误差块的变换系数排序, 随后将每个非零量化系数表示为 level 和 run 值。它还以类似于以上结合 INTRA 编码模式所述的方式, 采用熵编码对 run 和 level 值进行压缩。视频复用编码器 170 还经线路 126 从运动场编码块 140 接收运动矢量信息 (在下面描述) 并从控制管理器 160 接收控制信息。它对运动矢量信息和控制信息进行熵编码, 形成编码图像信息的单个比特流 135, 其中包括经熵编码的运动矢量、预测误差和控制信息。

表示宏块的每个块的预测误差信息的量化 DCT 系数还从量化器 106 传递到逆量化器 108。在该处, 它们被逆量化, 所得的逆量化 DCT 系数块应用到逆 DCT 变换块 110 上, 在逆 DCT 变换块 110 中, 它们经受逆 DCT 变换以产生本地解码的预测误差值块。本地解码的预测误差值块随后输入合并器 112。在 INTER 编码模式下, 设置开关 114, 以便合并器 112 还为宏块的每个块接收由运动补偿预测功能块 150 生成的预测像素值。合并器 112 将每个本地解码的预测误差值块与对应的预测像素值块合并, 以得到重建的图像块, 并将它们存储在帧存储器 120 中。

在功能块 104、106、108、110 和 112 中, 从视频源接收视频信号的后续宏块并对其执行上述的编码和本地解码步骤时, 于是在帧存储器 120 中建立了帧的解码版本。在已对帧的最后一个宏块作了处理之后, 帧存储器 120 包含完全解码的帧, 该帧可以在以 INTER 编码格式对后续收到的视频帧进行编码的过程中用作预测参考帧。

现在将描述当前帧的宏块的预测的形成。以 INTER 编码格式编码的任意帧需要参考帧以进行运动补偿预测。这必然意味着在对进视频序列行编码时, 要编码的第一帧无论是序列中的第一帧还是某个其它帧, 必须以 INTRA 编码格式对其进行编码。这又意味着在视频编码器 100 由控制管理器 160 切换到 INTER 编码模式时, 通过在本地对先前编码的帧进行解码而形成的完整参考帧已经在编码器的帧存储器 120 中可用。通常, 参考帧是通过在本地将 INTRA 编码帧或 INTER

编码帧解码而形成的。

形成当前帧宏块的预测的第一个步骤由运动估计功能块 130 执行。运动估计功能块 130 经线路 128 接收亮度和色度值块，这些数据块组成了要编码的帧的当前宏块。它随后执行数据块匹配操作以便识别参考帧中大致与当前宏块对应的区域。为执行数据块匹配操作，运动估计功能块经线路 127 访问存储在帧存储器 120 中的参考帧数据。更具体地说，运动估计功能块 130 通过计算表示被检查的宏块与帧存储器 120 中存储的参考帧的像素的候选最佳匹配区域之间像素值差异的差值（例如，绝对差值之和），执行数据块匹配。对于在参考帧的预定义搜索区域内所有可能偏移上的候选区域，产生相应的差值，并由运动估计功能块 130 确定最小的计算差值。当前帧中宏块与产生最小差值的参考帧中候选像素值块之间的偏差定义了所述宏块的运动矢量。

一旦运动估计功能块 130 已产生了宏块的运动矢量，则它将运动矢量输出到运动场编码功能块 140。运动场编码功能块 140 使用包括一组基本函数和运动系数的运动模型，逼近从运动估计功能块 130 接收的运动矢量。更具体地说，运动场编码功能块 140 将运动矢量表示为运动系数值集，这些系数值与基本函数相乘时形成运动矢量的近似值。通常，采用只具有两个运动系数和基本函数的平移运动模型（translational motion model），但也可以使用更复杂的运动模型。

运动系数从运动场编码功能块 140 传递到运动补偿预测功能块 150。运动补偿预测功能块 150 还从帧存储器 120 接收由运动估计功能块 130 识别的最佳匹配候选像素值区域。利用由运动场编码功能块 140 生成的运动矢量的近似表示和参考帧的最佳匹配候选区域的像素值，运动补偿预测功能块 150 为宏块的每个块生成预测像素值阵列。每个预测像素值阵列传递到合并器 116，在合并器中，从当前宏块的对应块中的实际（输入）像素值中减去预测像素值。这样，获得宏块的测误差块集。

现在将描述图 2 所示的视频解码器 200 的操作。解码器 200 包括视频复用解码器 270、逆量化器 210、逆 DCT 变换器 220、运动补偿预测功能块 240、帧存储器 250、合并器 230、控制管理器 260 及输出 280，其中视频复用解码器 270 从编码器 100 接收编码的视频比特流 135，并将其去复用成各组成部分。

控制管理器 260 响应于是否在对 INTRA 或 INTER 编码帧解码而对解码器 200 的操作进行控制。使解码器在解码模式之间切换的 INTRA/INTER 触发控制信号例如可从与从编码器接收的每个压缩视频帧相关联的图像类型信息中获得。INTRA/INTER 触发控制信号由视频复用解码器 270 从编码的视频比特流中提取，并经控制线 215 传递到控制管理器 260。

对 INTRA 编码帧的解码是基于宏块来执行的，每个宏块实际上是在视频比特流 135 中一收到与其相关的编码信息后就被解码。视频复用解码器 270 将宏块的块的编码信息与可能的涉及所述宏块的控制信息分开。INTRA 编码宏块的每个块的编码信息包括变长码字，该变长码字表示该块的非零量化 DCT 系数的熵编码 level 和 run 值。视频复用解码器 270 使用与编码器 100 中所用编码方法对应的变长解码方法将变长码字解码，从而恢复 level 和 run 值。随后，它重建宏块的每个块的量化变换系数值阵列，并将它们传递到逆量化器 210。与宏块相关的任何控制信息也在视频复用解码器中采用适当的解码方法解码，并传递到控制管理器 260。具体地说，与应用到变换系数的量化级相关的信息由视频复用解码器 270 从编码比特流中提取，并经控制线 217 提供给控制管理器 260。该控制管理器又将此信息经控制线 218 传送到逆量化器 210。逆量化器 210 根据控制信息将宏块的每个块的量化 DCT 系数逆量化，并将现在的逆量化 DCT 系数提供给逆 DCT 变换器 220。

逆 DCT 变换器 220 对宏块的每个块的逆量化 DCT 系数执行逆 DCT 变换，以形成包括重建像素值的图像信息的解码数据块，宏块的

每个块的重建像素值经合并器 230 传递到解码器的视频输出 280，例如，它们可从该处提供给显示装置（未显示）。宏块的每个块的重建像素值还存储在帧存储器 250 中。由于在对 INTRA 编码宏块的编码/解码中未使用运动补偿预测，因此，控制管理器 260 控制合并器 230 将这样的每个像素值块传递到视频输出 280 和帧存储器 250。随着后续 INTRA 编码帧宏块被解码和存储，解码帧逐渐汇集在帧存储器 250 中，并因而可用作参考帧，以便在对后续接收的 INTER 编码帧解码时用于运动补偿预测。

INTER 编码帧也按宏块进行解码，每个 INTER 编码宏块实际上是在视频比特流 135 中一收到与其相关的编码信息就被解码。视频复用解码器 270 将 INTER 编码宏块的每个块的编码预测误差信息与编码运动矢量信息和与所述宏块相关的可能控制信息分开。如上所述，宏块的每个块的编码预测误差信息包括变长码字，该变长码字表示所述预测误差块的非零量化变换系数的熵编码的 level 和 run 值。视频复用解码器 270 使用与编码器 100 中所用编码方法对应的变长解码方法将变长码字解码，由此恢复 level 和 run 值。随后，它重建每个预测误差块的量化变换系数值阵列，并将它们传递到逆量化器 210。与 INTER 编码宏块相关的控制信息也在视频复用解码器 270 中采用适当的解码方法解码，并传递到控制管理器 260。与应用到预测误差块变换系数的量化级相关的信息从编码比特流中提取，并经控制线 217 提供给控制管理器 260。该控制管理器又将此信息经控制线 218 传送到逆量化器 210。逆量化器 210 根据控制信息将表示宏块的每个块的预测误差信息的量化 DCT 系数逆量化，并将现在的逆量化 DCT 系数提供给逆 DCT 变换器 220。随后在逆 DCT 变换器 220 中将表示每个数据块的预测误差信息的逆量化 DCT 系数逆变换，以产得到宏块的每个块的重建预测误差值阵列。

与宏块相关联的编码运动矢量信息由视频复用解码器 270 从编码视频比特流 135 中提取。由此获得的解码运动矢量信息经控制线 225

传递到运动补偿预测功能块 240，运动补偿预测功能块 240 使用与编码器 100 中用于将 INTER 编码宏块编码的运动模型相同的运动模型，重建宏块的运动矢量。重建的运动矢量接近编码器的运动估计功能块 130 原来确定的运动矢量。解码器的运动补偿预测功能块 240 使用重建的运动矢量识别帧存储器 250 中存储的预测参考帧中重建像素区域的位置。参考帧例如可以是先前解码的 INTRA 编码帧，或者先前解码的 INTER 编码帧。在任何一种情况下，将重建运动矢量指示的像素区域用于形成所述宏块的预测。更具体地说，运动补偿预测功能块 240 通过从在参考帧中识别的像素区域复制对应的像素值，形成宏块的每个块的像素值阵列。预测，即从参考帧导出的像素值块，从运动补偿预测功能块 240 传递到合并器 230，在合并器 230 中，将其与解码的预测误差信息组合在一起。实际上，将每个预测块的像素值添加到由逆 DCT 变换器 220 输出的对应重建预测误差值中。这样，获得了宏块的每个块的重建像素值阵列。重建像素值传递到解码器的视频输出 280，并且还存储在帧存储器 250 中。

随着后续 INTER 编码帧宏块被解码和存储，解码帧逐渐汇集在帧存储器 250 中，并因而可用其它 INTER 编码帧的运动补偿预测的参考帧。

现在将更详细地分析使用变长编码（VLC）技术，对与量化变换系数相关联的 run 和 level 值进行熵编码。

一般情况下，将 run-level 对作为单个符号编码。这通过在每个可能的 run-level 对与定义的码号（code number）之间定义映射关系来实现。表 3 中显示了一个 run-level 对到码号的查找表形式的映射示例。使用查找表中的映射，为每个块的对(run, level)指定一个码号，并通过 VLC 编码技术对所得的码号进行编码。例如，在 Joint Model (JM) 2.0 工作草案编号 2（Joint Model (JM) 2.0 Working Draft Number 2, Doc. JVT-B118, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG Pattaya, 2nd JVT Meeting, Geneva, CH, Jan. 29 – Feb. 1, 2002）（H.26L

视频编码标准)中,定义了将(run, level)对映射到码号上的两个不同查找表。选择使用哪个查找表基于数据块的 QP 值和编码模式。更具体地说,如果编码模式是 INTRA 且 QP 小于 24,则使用不同的表。

JVT H.26L 的 Joint Model JM 2.0 利用了称为 UVLC 的所谓的“通用”变长编码方案。因此,根据 H.26L JM 2.0 实现的视频编码器使用单个变长码字集对要在比特流中从编码器发送的所有信息(语法元素)进行熵编码。虽然使用相同的码字,但为不同类型信息的编码定义了预定数量的数据符号到码字的不同的映射。例如,为包含亮度信息的块提供了两个数据符号到码字的映射,而映射的选择取决于用于将量化 DCT 变换系数排序的之字形扫描操作的类型(单扫描或双重扫描)。有关根据 H.26L 定义的单扫描和双重扫描之字形扫描方案,请参阅工作草案编号 2(Working Draft Number 2, Doc. JVT-B118, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG Pattaya, 2nd JVT Meeting, Geneva, CH, Jan. 29 – Feb. 1, 2002)。除其它映射之外,还为宏块类型(MB_Type)信息、运动矢量数据(MVD)和 INTRA 预测模式信息提供不同的映射(参见下表 3)。

H.26L JM 2.0 中定义的 UVLC 码字可以下面表 1 所示的如下压缩形式编写,其中, x_n 项可取 0 或 1 值。

```

1
0 1  $x_0$ 
0 0 1  $x_1 x_0$ 
0 0 0 1  $x_2 x_1 x_0$ 
0 0 0 0 1  $x_3 x_2 x_1 x_0$ 
.....

```

表 1: 根据 H.26L JM 2.0 生成 UVLC 码字的方案

表 2 显示了根据表 1 所示方案生成的前 16 个 UVLC 码字,而表 3 显示了根据 H.26L 提供的一些不同数据符号到码字映射。要了解完

整的映射说明，读者应再次参阅工作草案编号 2（Working Draft Number 2, Doc. JVT-B118, Joint Video Team (JVT) of ISO/IEC MPEG and ITU-T VCEG Pattaya, 2nd JVT Meeting, Geneva, CH, Jan. 29 – Feb. 1, 2002, Section 3.2）。

码字索引	VLC 码字
0	1
1	010
2	011
3	00100
4	00101
5	00110
6	00111
7	0001000
8	0001001
9	0001010
10	000010000
11	000010001
12	000010010
13	000010011
14	000010100
15	000010101
...	...

表 2: 根据表 1 所示方案生成的前 16 个 H.26L JM 2.0 UVLC 码字

Code_ number	MB_Type		INTRA_ prediction_ mode		MVD	Tcoeff_chroma_AC		Tcoeff_luma	
	INTRA	INTER	Prob0	Prob1		Tcoeff_luma 简单扫描		双重扫描	
						Level	Run	Level	Run
0	Intra4x4	Skip	0	0	0	EOB	-	EOB	-
1	0,0,0 ³	16x16	1	0	1	1	0	1	0
2	1,0,0	8x8	0	1	-1	-1	0	-1	0
3	2,0,0	16x8	0	2	2	1	1	1	1
4	3,0,0	8x16	1	1	-2	-1	1	-1	1
5	0,1,0	8x4	2	0	3	1	2	2	0
6	1,1,0	4x8	3	0	-3	-1	2	-2	0
7	2,1,0	4x4	2	1	4	2	0	1	2
8	3,1,0	Intra4x4	1	2	-4	-2	0	-1	2
9	0,2,0	0,0,0 ³	0	3	5	1	3	3	0
10	1,2,0	1,0,0	0	4	-5	-1	3	-3	0
11	2,2,0	2,0,0	1	3	6	1	4	4	0
12	3,2,0	3,0,0	2	2	-6	-1	4	-4	0
13	0,0,1	0,1,0	3	1	7	1	5	5	0
14	1,0,1	1,1,0	4	0	-7	-1	5	-5	0
15	2,0,1	2,1,0	5	0	8	3	0	1	3
16	3,0,1	3,1,0	4	1	-8	-3	0	-1	3
17	0,1,1	0,2,0	3	2	9	2	1	1	4
18	1,1,1	1,2,0	2	3	-9	-2	1	-1	4
19	2,1,1	2,2,0	1	4	10	2	2	2	1
20	3,1,1	3,2,0	0	5	-10	-2	2	-2	1
21	0,2,1	0,0,1	1	5	11	1	6	3	1
22	1,2,1	1,0,1	2	4	-11	-1	6	-3	1
23	2,2,1	2,0,1	3	3	12	1	7	6	0
24	3,2,1	3,0,1	4	2	-12	-1	7	-6	0
25		0,1,1	5	1	13	1	8	7	0
26		1,1,1	5	2	-13	-1	8	-7	0
27		2,1,1	4	3	14	1	9	8	0
28		3,1,1	3	4	-14	-1	9	-8	0
29		0,2,1	2	5	15	4	0	9	0
30		1,2,1	3	5	-15	-4	0	-9	0
31		2,2,1	4	4	16	5	0	10	0
32		3,2,1	5	3	-16	-5	0	-10	0
33			5	4	17	3	1	4	1
34			4	5	-17	-3	1	-4	1
35			5	5	18	3	2	2	2
36					-18	-3	2	-2	2
37					19	2	3	2	3
38					-19	-2	3	-2	3
39					20	2	4	2	4
40					-20	-2	4	-2	4
41					21	2	5	2	5
42					-21	-2	5	-2	5
43					22	2	6	2	6
44					-22	-2	6	-2	6
45					23	2	7	2	7
46					-23	-2	7	-2	7
47					24	2	8	11	0
..					..	..	..	..	..

表格 3: H.26L TML-5 中定义的示例数据符号到 UVLC 码字映射

H.26L 中采用的通用变长编码方案具有多种技术优点。具体地说, 使用可根据诸如表 1 所示的简单规则构造的单个 VLC 码字集允许逐

比特创建码字。这消除了在编码器和解码器中存储码字表的需要，因而降低了对编码器和解码器的存储器要求。当然，应注意，在替代实施中，可在编码器和解码器中生成并存储码字表。不同的数据符号到码字映射至少允许对 UVLC 编码方案进行有限的调整，以适应经过熵编码的各种类型的图像相关数据和控制信息的不同统计属性。不同数据符号到码字映射可存储在编码器和解码器中，因而无需在比特流中从编码器向解码器发送映射信息。这有助于保持抗差错能力。

然而，UVLC 熵编码方案也有一些缺陷。具体地说，不同数据符号到码字映射提供的固定码字和有限的适应性不可避免地得到并非最佳的数据压缩效果。这归因于如下事实：在图像编码中，不同变换系数的出现频率（即概率）及相应的不同(run, level)对的概率会根据在编码的图像内容和图像类型改变。因此，如果使用单个变长码字集，并且在要编码/解码的数据符号与 VLC 之间只提供了单个映射，则通常无法取得最佳的编码效能。

为此，已建议对基本 UVLC 编码方案加以改进。更具体地说，建议包括不止一个 VLC 码字集。Gisle Bjontegaard 在 Q.15/SG16 “在用于 INTRA Luma 系数的两个 VLC 之间使用自适应切换”（Use of Adaptive Switching Between Two VLC for INTRA Luma Coefficients, Doc. Q15-K-30, August 2000）中提出了这样的提议。此提议建议对某些类型的信息熵编码使用第二 VLC 码字集。根据文档 Q15-K-30，所提议的第二个 VLC 码字集称为 VLC2，它是根据下面表 4 中所示的方案构造的。具体地说，它用于对 run 和 level 值进行编码，这些值与使用根据 H.26L 定义的双重之字形扫描方案扫描的 4×4 像素 INTRA 编码亮度块的量化变换系数相关联。表 5 以明确的形式显示了该 VLC2 的前 16 个码字。

1 1 0
1 1 1
0 1 $x_1 x_0$
0 0 1 $x_2 x_1 x_0$
0 0 0 1 $x_3 x_2 x_1 x_0$
.....

表 4: 根据 Q.15/SG 16 文档 Q15-K-30 生成 VLC2 码字的方案

码字索引	VLC2 码字
0	10
1	110
2	111
3	0100
4	0101
5	0110
6	0111
7	001000
8	001001
9	001010
10	001011
11	001100
12	001101
13	001110
14	001111
15	00010000

表 5: 根据表 4 所示方案生成的 Q.15/SG 16 文档 Q15-K-30 前 16 个 VLC2 码字

通过将表 4、表 5 分别与表 1、表 2 比较可以看出，VLC2 的最短码字有两个比特，而为原来的 UVLC 码字集中的最短码字分配的为单比特。由于通用的可解码性要求任何 VLC 码字不可以是其它任何码字的前缀，因此，这种变更对其它 VLC2 码字的结构有重要的影响。

具体地说，为最短码字分配两个比特使得可以用更少的比特来表示具有具有更高索引值的码字。例如，参照表 2，可以看到索引 7 对应的 UVLC 码字有 7 个比特，而索引 7 对应的 VLC2 码字只有 6 个比特。

同样也可观察到，一个块内非零系数数量与同一块中可能的(run, level)对之间有很强的相关性。应注意，数据块中非零系数的数量受变换块大小的限制，例如，对于 4×4 变换，该数量上限为 16。同样地，run 的值受变换块大小的限制，例如，对于 4×4 变换，该值上限为 15。非零系数的数量很大，如接近阵列的总大小时，可观察到具有小 run 值和高 level 值的(run, level)对具有较高的概率。例如，对于具有 15 个非零系数的 4×4 变换，run 只可以在范围[0, 1]内取值。

基于这些观察，Karczewicz 所提的提议“高比特率的 VLC 系数编码”（“VLC Coefficients Coding for High Bit-Rate”，Doc. JVT-B072, 2nd JVT Meeting, Geneva, CH, Jan. 29 – Feb. 1, 2002）中建议了一种变长编码方法，在该方法中，提供了单独的查找表，以对应于不同数量的非零值量化变换系数。根据此提议，码字及(run, level)对与每个表的码字之间的映射设计为：在应用到从具有指定数量的非零值系数的量化变换系数值阵列导出的(run, level)对集时，提供最佳的数据压缩。有关(run, level)对统计属性相对于非零值量化变换系数数量的变化的信息是以经验方式获得的，例如，通过预先对若干测试（“训练”）图像或视频序列进行编码而获得。随后可设计 VLC 码字及码字之间的映射并作为编码器中的一个或多个查找表存储。

Karczewicz 建议的算法细节如下：在图像像素块或预测误差值数据块已进行变换编码以形成二维变换系数值阵列，且每个系数值已量化后，则阵列中非零量化系值的数量被确定。将称为 N_c 的值设为该数量并用于明确表示阵列中非零系数值的数量。因此，不再需要 EOB 符号（例如，等于零的 level 值）。随后，通过使用定义(run, level)对与码字之间映射的查找表，将所述块中的每个(run, level)对映射到码字上，其中，查找表是根据 N_c 值来选择的。

上述两种 VLC 编码方法中，通过在不止一个码字集之间进行基

于上下文的切换来选择码字；这两种方法与使用单个 VLC 码字集的方法相比，均提高了数据压缩效率。然而，在许多数据压缩应用中，特别是在与视频编码相关的应用中，人们不断期望提高数据压缩效率。因此，对于开发新的变长编码方法，以提高对要编码的数据符号类型和统计属性的适应性，同时保持低计算复杂性、低存储要求和良好的抗差错能力，人们仍然有着相当高的兴趣。正是在这样的背景下，提出了根据本发明的方法。

发明内容

本发明一般涉及表示为(run, level)对的数据符号的变长编码。它涉及使用数据符号与 VLC 码字之间的至少两个映射，并定义映射之间的切换规则，这些规则考虑了要进行变长编码的数据符号的统计属性。这样，与在数据符号与码字之间使用固定映射的 VLC 编码方法相比，取得了提高的数据压缩效率。切换规则设计为使得无需在比特流中包括与映射选择相关的任何信息，便可将包括按照本发明方法形成的 VLC 码字的比特流解码。换言之，无需在比特流中提供用于生成每个码字的映射的明确指示。此属性增强了数据压缩效率。

具体地说，本发明提出用于对量化变换系数进行编码以提高编码效率的方法。更具体地说，本发明的具体目的是提供一种用于对表示非零值量化变换系数的(run, level)对进行编码的新的更有效的方案。象 Karczewicz 在文档“高比特率的 VLC 系数编码”(“VLC Coefficients Coding for High Bit-Rate”, Doc. JVT-B072, 2nd JVT Meeting, Geneva, CH, Jan. 29 – Feb. 1, 2002)中建议的方法一样，本发明基于如下观察：变换系数块中非零值系数的数量以及 run 的值受变换块大小限制。此观察及对现实图像操作的编码器的进一步分析使本发明的发明者意识到，在块中非零系数数量与所述块中 run 值之间存在相关性。具体地说，他们意识到随着从块将越来越多的(run, level)对编码，run-level 组合的可能数量开始减少。正是此后一认识促成了本发明。

因此，根据本发明的 VLC 编码方法将如下事实纳入考虑：随着

为连续的(run, level)对指配了码字,与要编码的下一(run, level)对相关联的 run 值的可能范围变得更加有限。

Run 的可能值随每次(run, level)对的编码而变化的事实可以在(run, level)对到码号的索引分配期间加以利用。

更具体地说,根据本发明的实施例,至少提供了两个映射集以将(run, level)对与码号相关。利用根据 max_run 值从至少两个映射集中之一选择的映射表,首先将每个(run, level)对映射到一个码号, max_run 值定义了 run 可以取的最大值。通过考虑块内非零值系数数量将 Max_run 初始化,并在将每个(run, level)对编码后对 Max_run 进行更新。以下示例阐明了这一点:

假设一个量化变换系数块具有 5 个非零系数,带有(run, level)对(2,1)、(3,2)、(4,3)、(2,3)和(1,2)。此处应注意的是,通常给定系数的 run 和 level 值是有序的,以便 level 值在相关联的 run 值之前。因此,在上述(run, level)对中,level 值是每对的第一个数字(2,3,4,2,1),并且 run 值是每对的后一个数字(1,2,3,3,2)。在编码过程开始时,由于有 5 个非零系数,并且最多可以有 16 个这种系数,因此,run 可以取的最大值为 11。将此最大可能的 run 值表示为 max_run,可以看到,在将第一个(run, level)对,即(2,1)编码后, max_run 将减少,准确地说,通过减去当前 run 值而更新先前的 max_run 值,可发现它将变成 10。更具体地说, max_run 的更新方式为:

$$\text{max_run} = \text{max_run} - \text{run}$$

其中, max_run 在块编码开始时,初始化为 $16 - N_c$ 。

根据本发明的第一方面,提供了一种将包括零值数据符号和非零值数据符号的数据符号集编码的方法,所述方法包括通过映射操作指配码字以表示非零值数据符号及相关联的若干在前或在后的零值符号。根据所述方法,所述映射操作依赖于所述数据符号集中零值数据符号的最大可能数量;通过所述映射操作,将一个码字指配给非零值数据符号及其相关联的若干在前或在后的零值数据符号。

根据本发明的第二方面,提供了一种将表示数据符号集的码字集

解码的方法，所述数据符号集包括零值数据符号和非零值数据符号，所述方法包括接收所述数据符号集中非零值数据符号的数量指示，接收表示非零值数据符号及相关联的若干在前或在后的零值数据符号的码字，以及通过执行逆映射操作，将所述码字解码以恢复所述非零值数据符号及其相关联的若干在前或在后的零值数据符号。根据所述方法，所述逆映射操作依赖于所述数据符号集中零值数据符号的最大可能数量；通过所述逆映射操作，可从所述码字将所述非零值数据符号及其相关联的若干在前或在后的零值数据符号解码。

根据本发明的第三方面，提供了一种将包括零值数据符号和非零值数据符号的数据符号集编码的编码器；所述编码器包括用于通过映射操作指配码字，以表示非零值数据符号及相关联的若干在前或在后的零值符号的部件。根据本发明，指定编码器中提供的码字的部件设置来根据所述数据符号集中零值数据符号的最大可能数量来调整指配码字的映射操作。

根据本发明的第四方面，提供了一种将表示数据符号集的码字集解码的解码器；所述数据符号集包括零值数据符号和非零值数据符号；所述解码器包括：用于接收表示所述数据符号集中非零值数据符号数量指示的部件；用于接收表示非零值数据符号及相关联的若干在前或在后的零值数据符号的码字的部件；以及用于通过执行逆映射操作，将码字解码以恢复所述非零值数据符号及其相关联的若干在前或在后的零值数据符号的部件。根据本发明，用于将码字解码的部件设置来根据所述数据符号集中零值数据符号的最大可能数量来调整逆映射操作，通过逆映射操作，可将非零值数据符号及其相关联的若干在前或在后的零值数据符号解码。

根据本发明的第五方面，提供了一种多媒体终端，它包括根据本发明第三方面的编码器。

根据本发明的第六方面，提供了一种多媒体终端，它包括根据本发明第四方面的解码器。

根据本发明第五和/或第六方面的多媒体终端最好是移动多媒体

终端，用于通过无线电连接与移动通信网络通信。

附图说明

现在将参照附图，通过示例描述本发明的实施例，其中：

图 1 是根据现有技术的通用视频编码器的示意方框图；

图 2 是根据现有技术且与图 1 所示编码器对应的通用解码器的示意方框图；

图 3 显示了根据现有技术的宏块的形成；

图 4 显示了作为示例的之字形扫描顺序；

图 5 是根据本发明实施例的视频编码器的示意方框图；

图 6 是根据本发明实施例且与图 5 所示编码器对应的视频解码器的示意方框图；以及

图 7 是可实施根据本发明的方法的多媒体通信终端的示意方框图。

具体实施方式

在下面的详细说明中，给出了本发明的示范实施例。这些实施例涉及视频编码器中(run, level)对的变长编码及其在对应视频解码器中的后续解码，(run, level)对表示作为块变换编码结果产生的非零值变换系数。然而，本领域的技术人员会理解，根据本发明的方法可以一般地适用于表示为(run, level)对的数据符号或要编码的数据符号集的任何等效表示的 VLC 编码。

图 5 是可应用根据本发明的 VLC 编码方法的视频编码器 600 的示意方框图。图 5 所示视频编码器的结构大致与图 1 所示现有技术视频编码器的结构相同，但对执行变长编码操作的那些编码器部件进行了适当的修改。在某种意义上功能实现和操作与前述现有技术视频编码器相同的所有视频编码器部件以相同的标记标识。在此说明书中，假定所有变长编码操作在视频复用编码器 670 中执行。然而，应理解，在本发明的替代实施例中，可提供单独的变长编码单元。还应注意的

是,根据本发明的方法可应用于视频编码器生成的某些数据符号(例如,与非零值变换系数相关联的(run, level)对),并且其它 VLC 编码方法可用于将其它数据符号编码。

现在将详细地介绍视频编码器 600 的操作。在将数字视频帧编码时,编码器 600 在某种意义上以类似于前面结合图 1 所述的方式操作,以生成 INTRA 编码和 INTER 编码的压缩视频帧。如本文前面所述的那样,在 INTRA 编码模式下,对每个图像数据(像素值)块应用离散余弦变换(DCT),以生成对应的变换系数值二维阵列。DCT 操作在变换块 104 中执行,由此生成的系数随后传递到量化器 106 中进行量化。在 INTER 编码模式下,对预测误差值块应用功能块 104 执行的 DCT 变换。由此产生的变换系数也传递到量化器 106 并在其中量化。

应注意的是,INTER 编码帧可包含 INTRA 编码图像块;此外,在一些情况下,不对特定图像块应用变换编码。例如,如果 INTRA 预测在 INTRA 编码模式下使用,则在编码器中从一个或多个先前编码的图像块中预测一些图像块。这种情况下,编码器为解码器提供要在预测中使用的先前数据块的指示,并且不输出任何变换系数数据。此外,在 INTER 编码模式下,某个块的预测与该块本身的图像数据之间的差异可能很小,以致从数据压缩率角度上最好不发送任何预测误差信息。因此,根据本发明的 VLC 编码方法适用于经过变换编码及随后的变换系数量化的那些图像块。

根据本发明,在视频复用编码器 670 收到量化变换系数块(二维阵列)时,它确定阵列中非零值系数的数量 N_c ,并将此数量的指示在比特流 635 中传送给解码器。在本发明的优选实施例中, N_c 是在传输前作了变长编码。在替代实施例中,可以如上所述传送或可以应用另一种编码方法。

视频复用编码器 670 接着将非零值系数表示为(run, level)对。更具体地说,如图 4 中所示那样,先使用预定的扫描顺序扫描量化变换系数的二维阵列,以生成有序的一维阵列。有序的一维阵列中的每个非零值系数值随后用 run 值和 level 值表示,其中,level 值表示系数

值, run 值表示在非零值系数前连续的零值系数的数量。

随后, 视频复用编码器 670 对 run 和 level 值应用变长编码, 以便为每个(run, level)对指配一个 VLC 码字。根据本发明的优选实施例, 在将符号编码前考虑可能的最大 run(表示为 max_run), 通过选择(run, level)对到码号的不同映射, 将(run, level)对编码。在对量化变换系数块中第一(run, level)对进行编码时, 将 max_run 设为等于块中可存在的非零值系数的最大可能数量减去先前确定的 N_c 值。因此, 对于 4×4 块, max_run 初始化为:

$$\text{max_run} = 16 - N_c \quad (1)$$

以下示范伪码提供了根据本发明将码字指配给(run, level)对的一种方法:

$$\text{Index_RL} = \text{RL_index}[\text{max_run}, \text{level}, \text{run}]; \quad (2)$$

其中, RL_index 根据 max_run 值将 run_level 值的整数值映射到不同的码号。max_run 在使用训练材料运行模拟后初始化。最后, 用表 2 或表 5 所列 VLC 码字之一将 Index_RL 编码。

更具体地说, 对于每个可能的 max_run 值, 通过分析不同(run, level)对的出现概率, 以经验方式从一个或多个视频数据测试序列中构造映射函数 RL_index。随后将应用映射方法生成的码号(即, Index_RL 值)用于从码字表中选择表示(run, level)对的码字。本发明的优选实施例使用以上表 2 中所示的码字表或表 5 中所示的码字表。

表 6 和表 7 显示了根据 max_run 值将(run, level)对映射到码号的示例。在所述两个表中, 行对应于 level 值, 列对应于 run 值, 而每个单元格中的条目表示该特定 run-level 对(即, run 和 level 值的组合)的 VLC 码字的码号。为最可能的对相配码号 1, 为第二最可能的对指定码号 2, 以此类推。表 6 和表 7 均列出了前 15 个最可能的对。表 6 显示了 max_run 为 8 时的情况, 而表 7 显示了 max_run 为 3 时的情况。

Run

1	2	3	4	5	6	7	8	11	12
---	---	---	---	---	---	---	---	----	----

Level	5	10	13	15						
	9									
	14									

表 6: max_run = 8。

		Run			
Level	1	3	5	9	
	2	7	12		
	4	10			
	6	14			
	8				
	11				
	13				
	15				

表 7: max_run = 3

使用以上映射方法，选择 VLC 码字以表示特定的(run, level)对后，视频复用编码器 670 在比特流 635 中将码字发送到解码器。如果当前数据块有更多的(run, level)对要编码，则它将通过减去当前 run 值来更新 max_run，以便用于将下一(run, level)对编码，更新方式即：

$$\text{max_run} = \text{max_run} - \text{run}$$

编码过程会继续，直至已将给定块的所有(run, level)对表示为码字为止，随之视频复用编码器 670 接收下一量化变换系数块以进行编码，并对下一数据块重复编码过程。

现在将参照图 6 描述根据本发明的视频解码器 700 的操作。图 6 所示视频解码器结构大致与图 2 所示的现有技术视频解码器的结构相同，但对执行变长解码操作的那些解码器部件进行了适当的修改。功能实现和操作在某种意义上与前述现有技术视频解码器相同的所有视频解码器部件用相同的标记标识。在下面的说明中，假定所有变长解码操作在视频复用解码器 770 中执行。然而，应理解，在本发明的

替代实施例中，可提供单独的变长解码单元。

现在将详细地介绍该视频解码器的操作。这里，假定图 6 所示视频解码器与结合图 5 所述的编码器相对应，因此能够接收编码器 600 发送的比特流 635 并将其解码。在该解码器中，比特流由视频复用解码器 770 接收并分成各组成部分。如先前有关现有技术的所述那样，对从比特流中提取的压缩视频数据按宏块进行处理。根据本发明，INTRA 编码宏块的压缩视频数据包括表示宏块的每个块的 VLC 编码 (run, level) 对的变长码字，以及每块中出现的非零值变换系数的数量 N_c 的指示和编码控制信息（例如，与量化参数 QP 相关）。INTER 编码宏块的压缩视频数据包括每块的 VLC 编码预测误差信息（包括每块的 VLC 编码的 (run, level) 对和 N_c 指示）、宏块的运动矢量信息和编码的控制信息。

以同样的方式处理 INTRA 编码宏块的每个块的 N_c 值和 (run, level) 对和与 INTRA 编码宏块相关联的每个预测误差数据块的 N_c 值和 (run, level) 对。

更具体地说，在对 INTRA 或 INTER 编码图像块的 VLC 编码 (run, level) 对解码时，视频复用解码器 770 先确定该块中非零值量化变换系数的数量。如上所述，与非零值量化变换系数数量 N_c 相关的信息由编码器 600 插入比特流 635 中。此信息由解码器 700 的视频复用解码器 770 从接收的比特流中抽取。

随后，视频复用解码器 770 开始对图像块的 VLC 编码的 (run, level) 对解码。根据本发明，解码操作的执行方式类似于如上所述编码器中执行的编码过程。更具体地说，在对与特定图像块相关联的第一变长码字解码时，视频复用解码器 770 通过将 max_run 的值设为等于块中可能存在的非零值系数的最大数量减去从接收的比特流中提取的与所述块相关的 N_c 值，将 max_run 值初始化。它随后确定与接收码字对应的码号，并根据 max_run 的值，选择将该码号与 run 和 level 值的特定组合相关的映射表，解码器中使用的映射表对应于编码器中用于

生成码字的那些映射表。

在恢复与数据块第一非零值量化变换系数对应的 level 和 run 值后, 视频复用解码器 770 通过减去刚确定的 run 值来更新 max_run 的值, 并开始对数据块的下一码字进行解码。此过程会继续, 直到已将表示特定图像块的所有码字解码为止, 视频复用解码器 770 随之从解码的(run, level)对重建图像块的量化变换系数, 并将它们传递到逆量化器 210。如结合现有技术视频解码器 200 所述的那样, 视频解频过程的剩余部分继续, 从而形成每个图像块的重建像素值。

图 7 显示了包括可适于按照本发明操作的视频编码和解码设备的终端装置。更具体地说, 该图显示了根据 ITU-T 建议 H.324 实现的多媒体终端 80。终端可视为多媒体收发装置。它包括捕获数据流并将其编码和复用以便经通信网传送的单元以及接收多媒体内容并将其去复用、解码和显示的单元。ITU-T 建议 H.324 定义了终端的全部操作, 并涉及控制其不同构成部分操作的其它建议。此种多媒体终端可在诸如常规移动视频电话等实时应用或诸如从例如因特网多媒体内容服务器检索和/或流式传送视频剪辑的非实时应用中使用。

在本发明范围中, 应理解图 7 所示 H.324 终端仅仅是适于应用本发明方法的多种替代多媒体终端实施例之一。还应注意到, 就终端设备的位置和实现而言, 存在多种替代方案。如图 7 所示, 多媒体终端可位于连接到诸如模拟 PSTN (公共交换电话网) 等固定线路电话网络的通信设备中。这种情况下, 多媒体终端配有符合 ITU-T 建议 V.8、V.34 和可选的 V.8bis 标准的调制解调器 91。或者, 多媒体终端可连接到外部调制解调器。调制解调器允许将多媒体终端生成的复用数字数据和控制信号转换成适合经 PSTN 传输的模拟形式。它还使多媒体终端能够从 PSTN 接收模拟形式的数据和控制信号, 并将它们转换成可由终端以适当方式去复用并加以处理的数字数据流。

H.324 多媒体终端也可以这样实现, 使它可直接连接到数字固定线路网络, 如 ISDN (综合业务数字网络)。这种情况下, 调制调解

器 91 可替换为 ISDN 用户网络接口。在图 7 中, 此 ISDN 用户网络接口由备选功能块 92 表示。

H.324 多媒体终端也可适于在移动通信应用中使用。如果配合无线通信链路使用, 则调制解调器 91 可替换为任何适当的无线接口, 如图 7 中备选功能块 93 所示。例如, H.324/M 多媒体终端可包括无线电收发信机, 其可实现到当前第二代 GSM 移动电话网, 或所提出的第三代 UMTS (通用移动电话系统) 的连接。

应注意的是, 在设计用于双向通信, 即用于收发视频数据的多媒体终端中, 同时提供按照本发明实施的视频编码器和视频解码器是有利的。此类编码器和解码器对经常实现为一个组合功能单元, 称为“编解码器”。

现在将参照图 7 进一步详细描述典型的 H.324 多媒体终端。

多媒体终端 80 包括称为“终端设备”的多个单元。这包括一般性地分别用标记 81、82 和 83 表示的视频、音频和远程信息处理装置。例如, 视频设备 81 可包括捕获视频图像的摄像机、显示接收视频内容的监视器和可选的视频处理设备。音频设备 82 通常包括例如用于捕获语音消息的麦克风以及再现接收音频内容的扬声器。音频设备还可包括其它音频处理单元。远程信息处理设备 83 可包括数据终端、键盘、电子白板或静态图像收发信机, 如传真机。

视频设备 81 连接到视频编解码器 85。视频编解码器 85 包括均根据本发明实现的视频编码器和相应的视频解码器。此类编码器和解码器将在下面描述。视频编解码器 85 负责以适当的形式将捕获的视频数据编码, 以便通过通信链路传输, 并且将从通信网络接收的压缩视频内容解码。在图 7 所示示例中, 视频编解码器是根据 ITU-T 建议 H.26L 实现的, 但进行了适当的修改以便在视频编解码器的编码器和解码器中实施根据本发明的自适应变长编码方法。

终端的音频设备连接到图 7 中用标号 88 表示的音频编解码器。类似于视频编解码器, 音频编解码器包括编码器/解码器对。它将终端

音频设备捕获的音频数据转换成适合经通信链路传输的形式，以及将从网络接收的音频数据变换成适合再现的形式，例如，适合在终端的扬声器上再现。音频编解码器的输出传递到延迟块 87，以补偿视频编码过程引入的延迟，从而确保音频和视频内容的同步。

多媒体终端的系统控制功能块 84 采用适当的控制协议（信令功能块 88）在发送和接收终端之间建立共同的操作模式，控制端到网络的信令。信令功能块 88 交换有关发送和接收终端的编码和解码功能的信息，并可用于实现视频编码器的不同编码模式。系统控制部功能块 84 还控制数据加密的使用。有关在数据传输中要使用的加密类型的信息从加密功能块 89 传递到复用器/去复用器（MUX/DMUX 单元）90。

在始自多媒体终端的数据传输期间，MUX/DMUX 单元 90 将编码的、同步的视频和音频流与从远程信息处理设备 83 输入的数据和可能的控制数据组合在一起，形成单个比特流。有关要应用到比特流的数据加密类型（如果有的话）的信息由加密功能块 89 提供，用于选择加密模式。对应地，正在接收复用的并且可能加密的多媒体比特流时，MUX/DMUX 单元负责将该比特流解密，将其分成各多媒体组分量，并将这些分量传递到适当的编解码器和/或终端设备，以进行解码和再现。

应注意的是，根据本发明的多媒体终端功能单元、视频编码器、解码器和视频编解码器可实现为软件或专用硬件，或两者的组合。根据本发明的变长编码和解码方法特别适合以计算机程序的形式实现，该计算机程序包括用于执行本发明功能步骤的机器可读指令。因此，根据本发明的变长编码器和解码器可实现为存储在存储媒体上并在诸如个人台式计算机的计算机中执行的软件代码。

如果多媒体终端 80 是移动终端，即，如果它配有无无线电收发信机 93，则本领域的技术人员可以理解，它还可以包括其它单元。在一个实施例中，它包括具有允许用户操作多媒体终端 80 的显示器和键

盘、控制负责多媒体终端不同功能的功能块的中央处理器如微处理器、随机存取存储器 RAM、只读存储器 ROM 和数字摄像机。微处理器的操作指令是对应于多媒体终端 80 基本功能的程序代码，存储在只读存储器 ROM 中，并可在需要时，例如在用户的控制下由微处理器执行。微处理器按照程序代码使用无线电收发信机 93 形成与移动通信网络的连接，从而使多媒体终端 80 可经无线电路径从移动通信网络收发信息。

微处理器监视用户接口的状态并控制数字摄像机。微处理器响应用户命令而指示摄像机将数字图像录制到 RAM 中。一旦捕获了图像，或者在捕获过程期间，微处理器将图像分割成图像段（例如宏块），并如上所述使用编码器对各段执行运动补偿编码以生成压缩图像序列。用户可命令多媒体终端 80 在其显示屏上显示捕获的图像，或者使用无线电收发信机 93 将压缩的图像序列发送到另一多媒体终端、连接到固定线路网络（PSTN）的视频电话或其它一些电信装置。在优选实施例中，图像数据的传输在对第一段编码时便开始，以便收信方可以最小的延迟开始对应的解码过程。

虽然在特定实施例的范围中进行了描述，但本领域的技术人员清楚，可以对所述内容进行多种修改并予以不同的变化。因此，虽然已针对一个或多个优选实施例对本发明作了示意和描述，但本领域的技术人员将理解，可以在不脱离如上所述本发明范围和精神的情况下，进行某些修改或改变。

本发明的一个替代实施例基于以下观察：对系数的之字形扫描往往根据其值将系数排序，这样，一维阵列中位置较靠前的系数比阵列中位置较靠后的系数具有更大的绝对值。因此，连续的 level 值表现出极大的相似性，即，在给定的变换系数块内，要编码的系数的 level 一般具有与先前编码系数大致类似的幅值。此替代实施例利用了这种相似性，按照如下关系将(run, level)对编码：

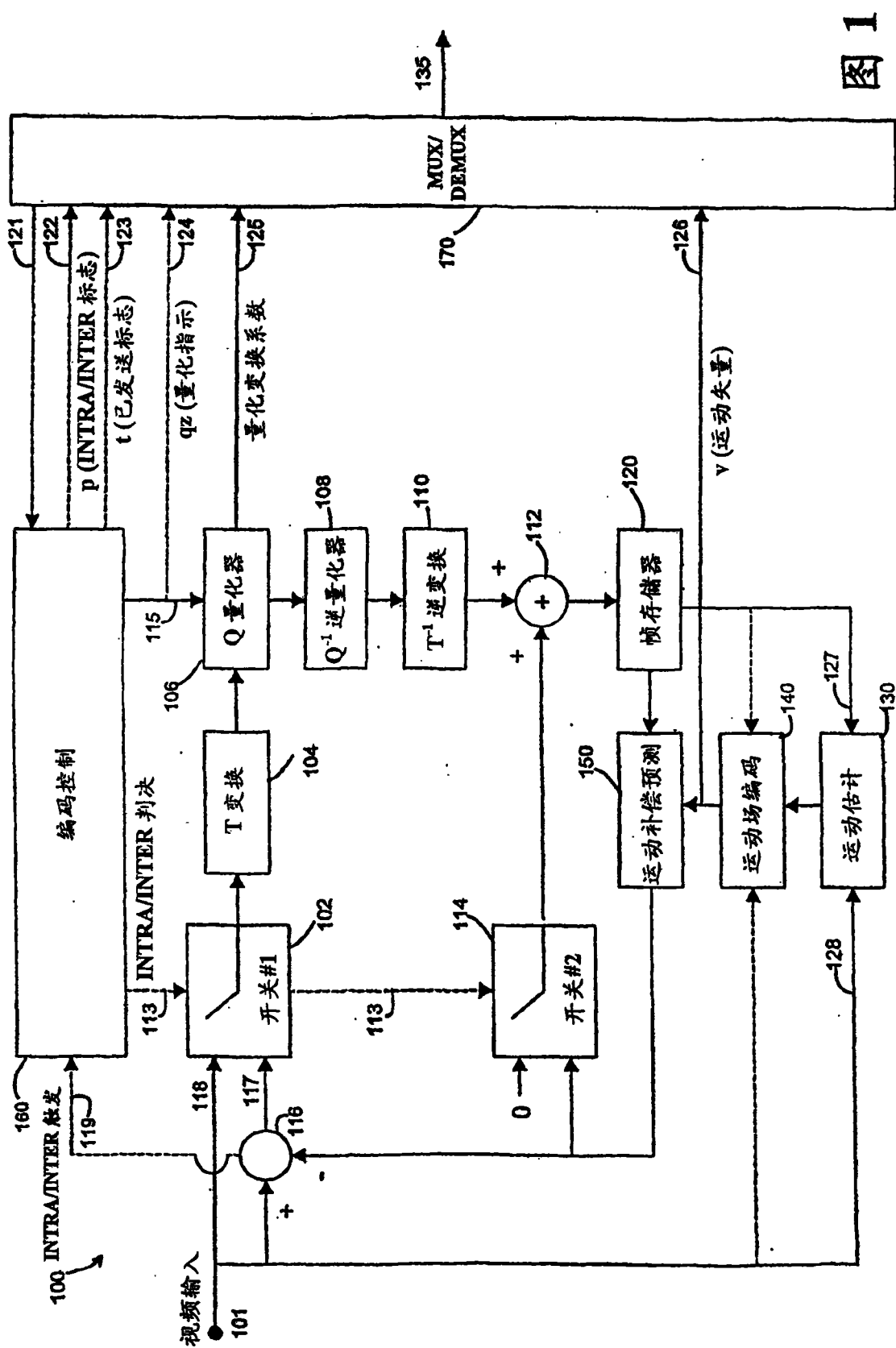
$$\text{Index_RL} = \text{RL_Index}\{\text{max_run, level, run, pre_level}\} \quad (3)$$

其中, `pre_level` 是块中先前编码的 `level` 值。与先前所述的实施例相比, 存在影响`(run, level)`的索引值分配的另一参数(即, `pre_level`)。

本发明的又一实施例依据的事实是: 连续的 `level` 值表现出很强的相关性。更具体地说, 先前编码的 `level` 值除了用于索引分配中, 还可用于指定不同的 VLC 码本以将当前符号编码。下面的示范伪码提供了一种方法, 用于将 VLC 码本指配给通过采用本发明前述实施例之一找到的`(run, level)`对码号:

$$\text{VLC_N} = \text{Nc_map}[\text{prev_level}] \quad (4)$$

其中, `prev_level` 是块中先前编码的 `level` 值, 并且 `Nc_map` 是将整数值映射到不同的可用 VLC 代码的阵列。



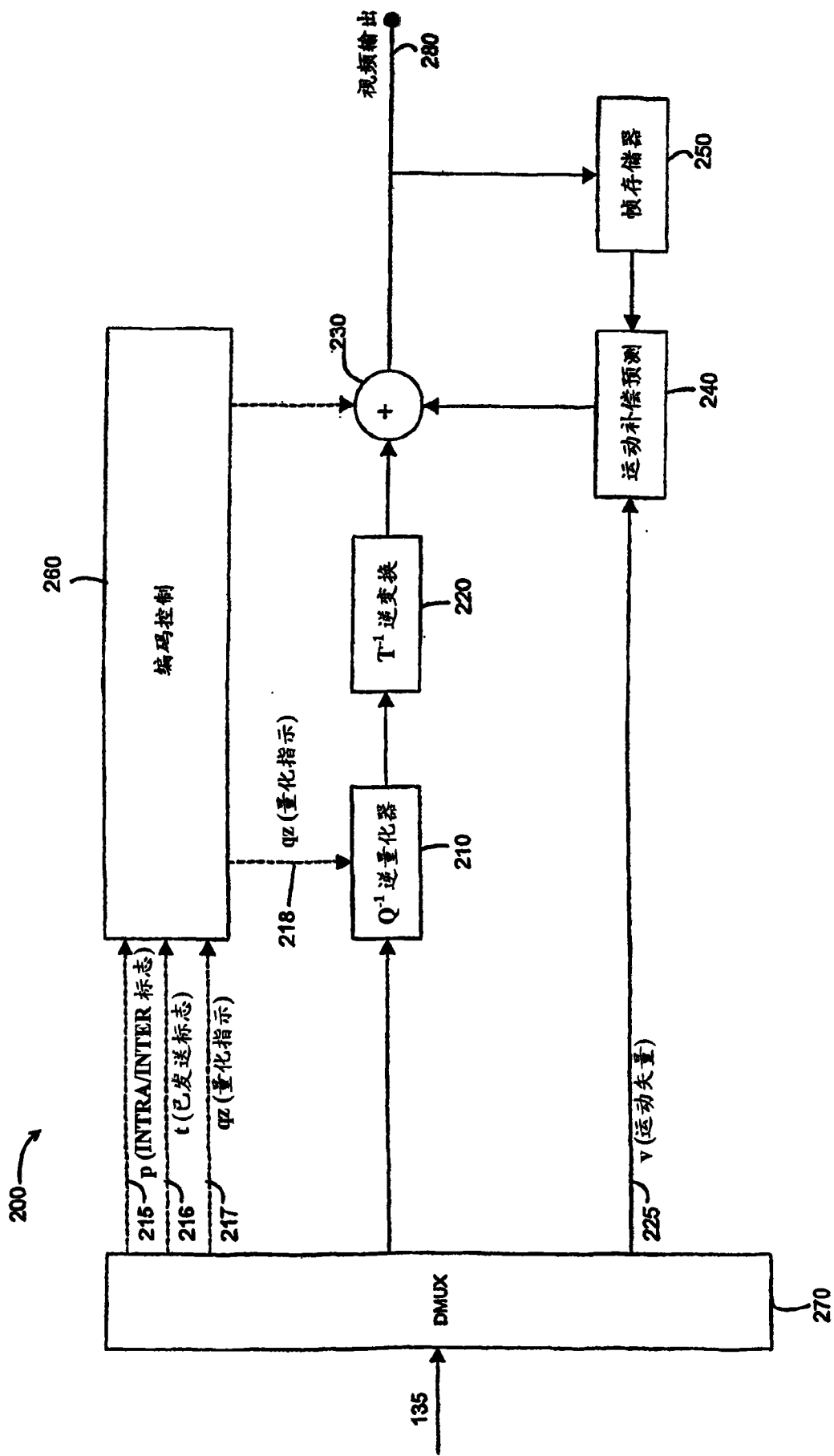


图 2

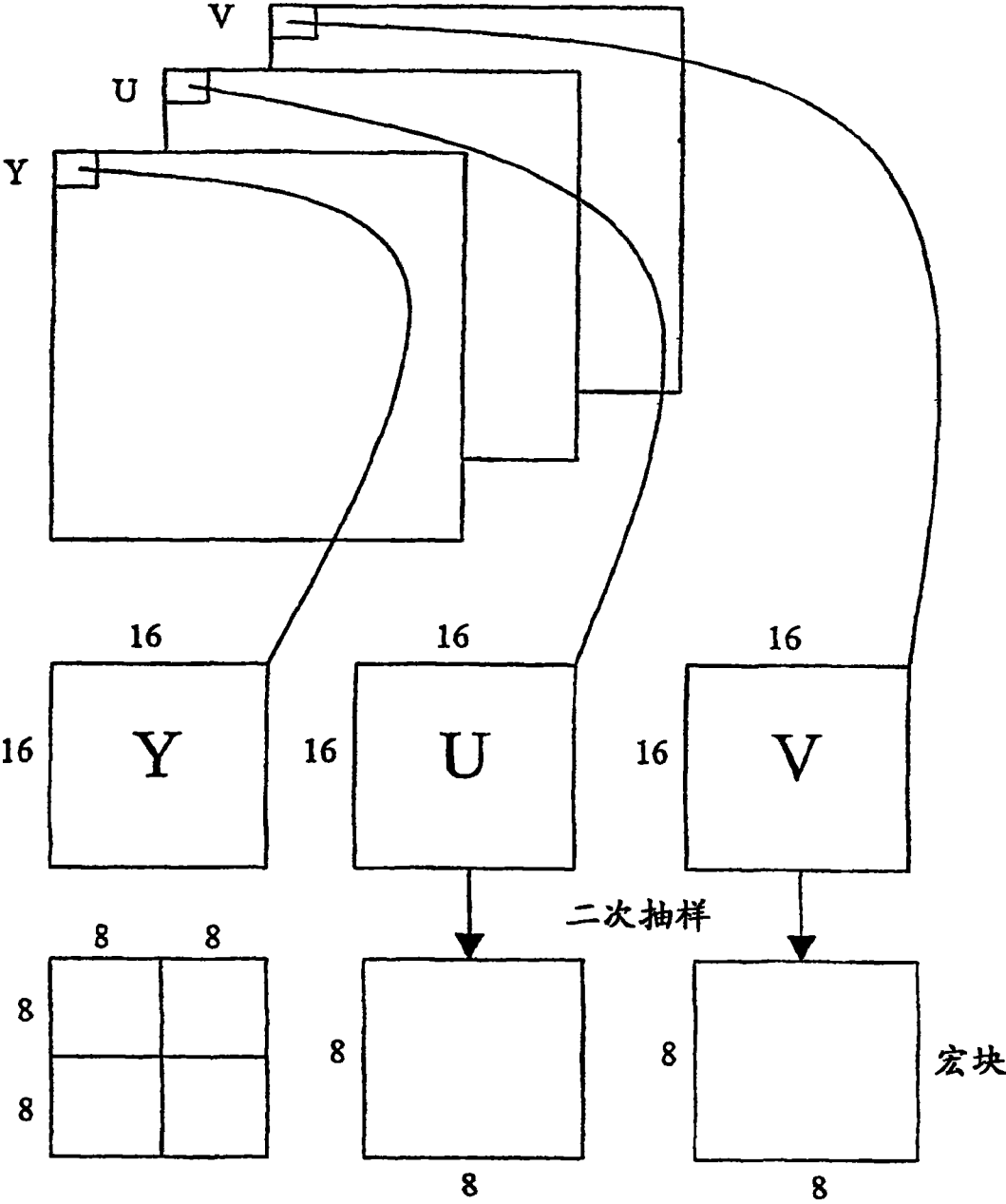


图 3

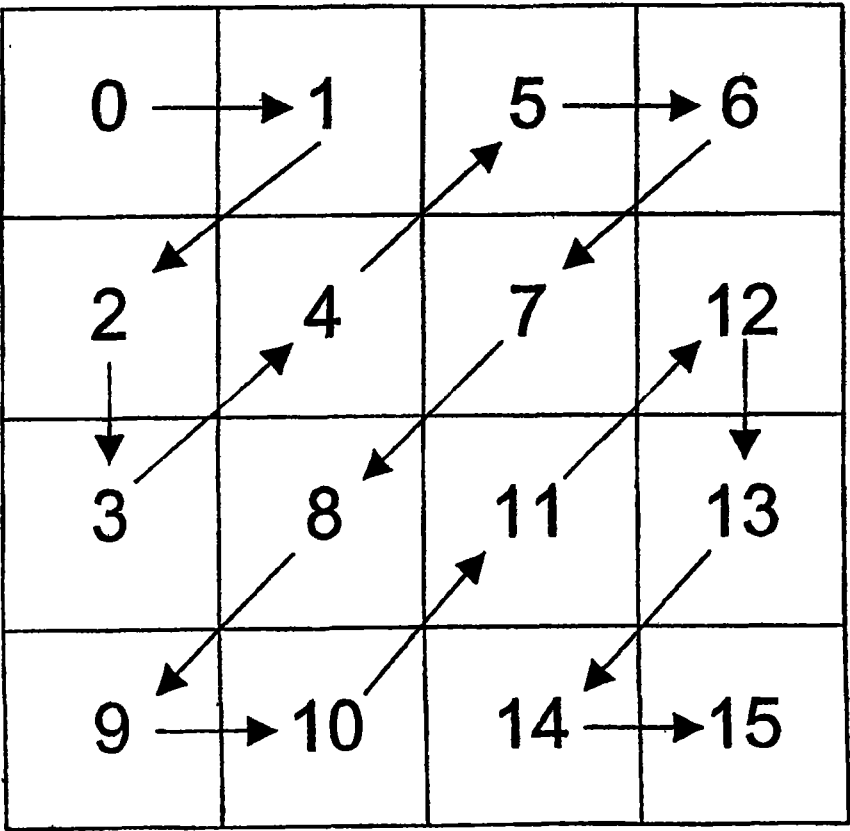
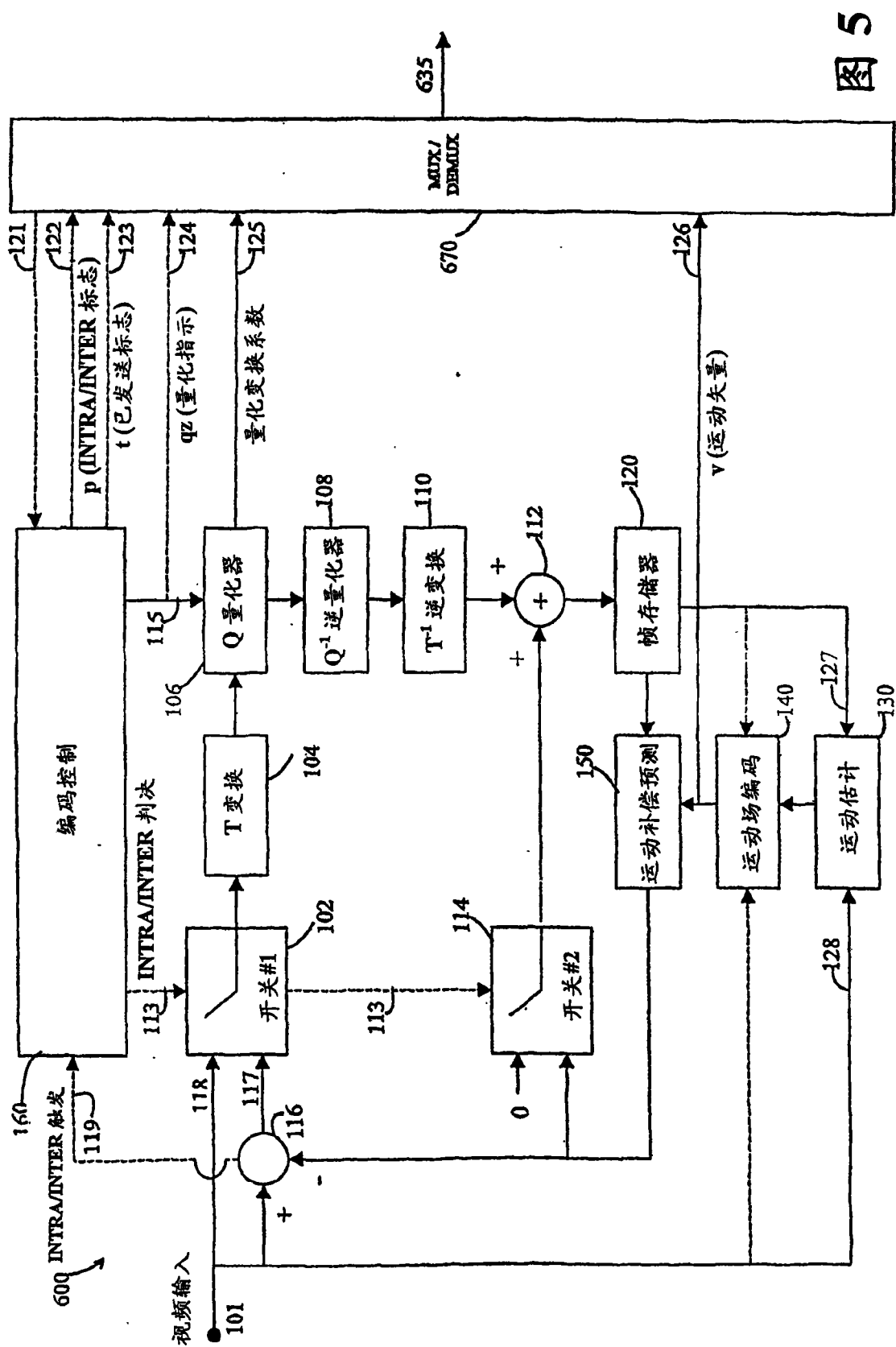


图 4



5
[Illegible]

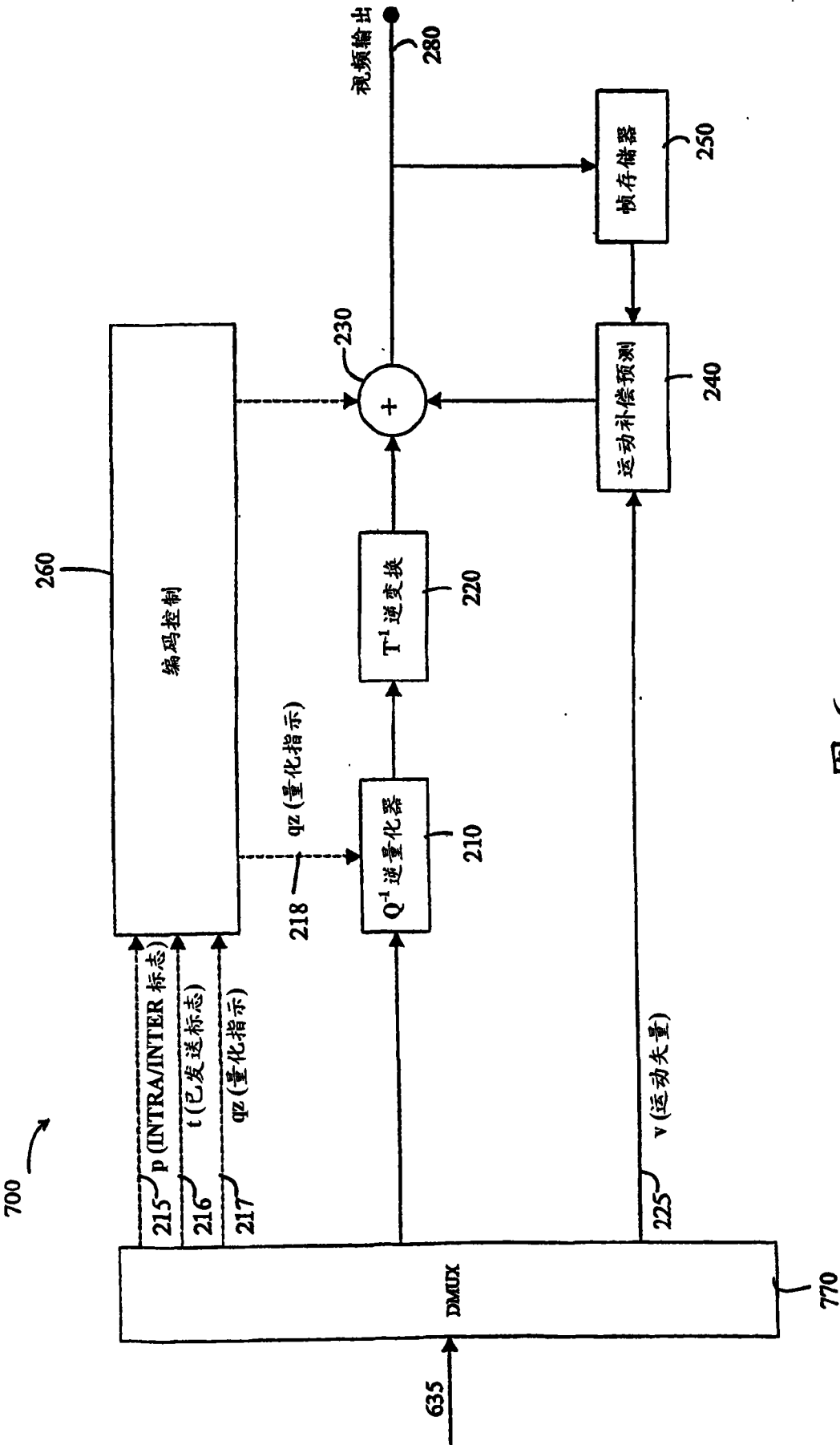


图 6

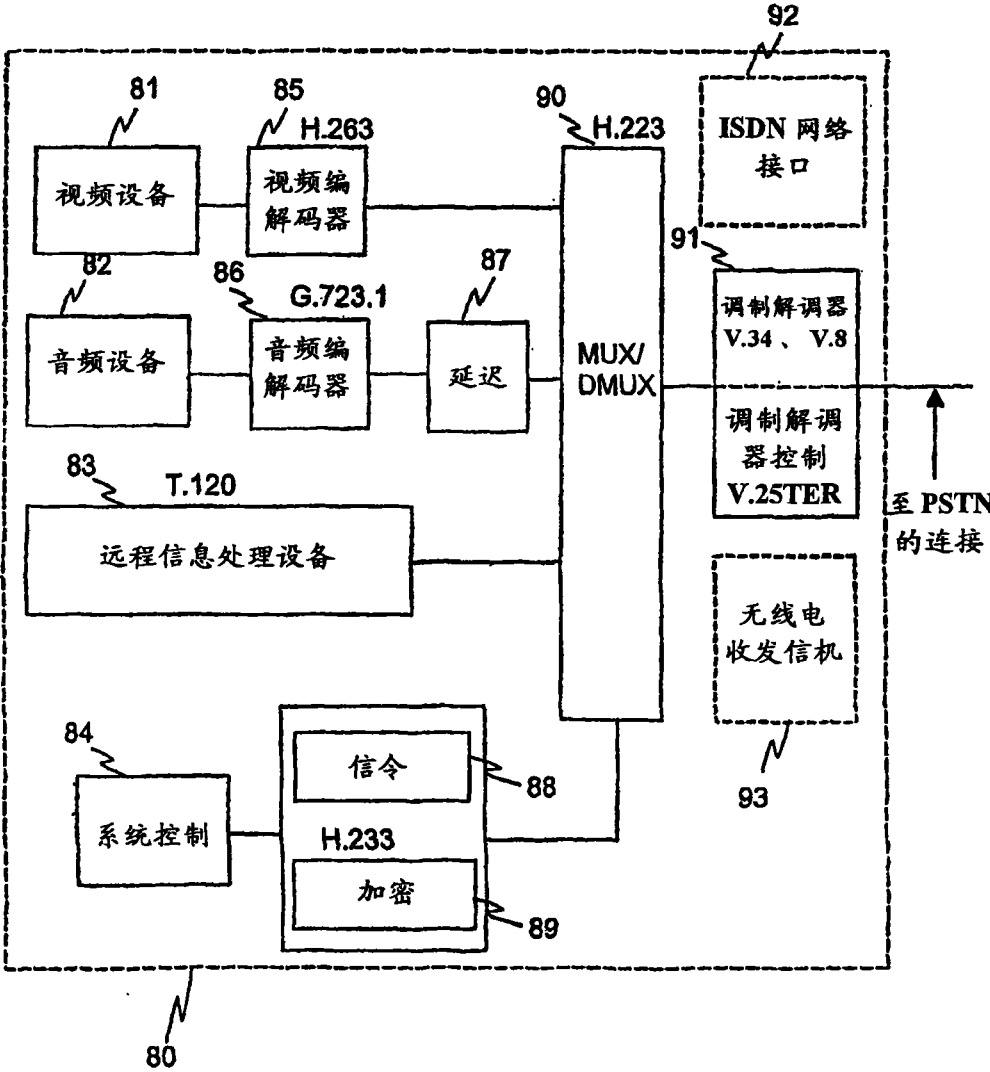


图 7