



(12) 发明专利

(10) 授权公告号 CN 101636761 B

(45) 授权公告日 2013. 07. 24

(21) 申请号 200780052303. 6

(22) 申请日 2007. 11. 02

(30) 优先权数据

60/919, 702 2007. 03. 23 US

60/934, 634 2007. 06. 14 US

(85) PCT申请进入国家阶段日

2009. 09. 23

(86) PCT申请的申请数据

PCT/US2007/023172 2007. 11. 02

(87) PCT申请的公布数据

W02008/118145 EN 2008. 10. 02

(73) 专利权人 汤姆森特许公司

地址 法国布洛涅 - 比扬库尔

(72) 发明人 邹德坤 杰弗里·A·布卢姆 尹澎

奥斯卡·迪沃拉埃斯科达

(74) 专利代理机构 北京市柳沈律师事务所

11105

代理人 吕晓章

(51) Int. Cl.

H04N 7/26 (2006. 01)

H04N 7/50 (2006. 01)

H04N 21/2389 (2011. 01)

H04N 21/8358 (2011. 01)

G06T 1/00 (2006. 01)

(56) 对比文件

US 2002/0015509 A1, 2002. 02. 07, 全文.

US 2003/0185417 A1, 2003. 10. 02, 全文.

CN 1473438 A, 2004. 02. 04, 全文.

CN 1659855 A, 2005. 08. 24, 全文.

US 2006/0020809 A1, 2006. 01. 26, 全文.

审查员 侯冠华

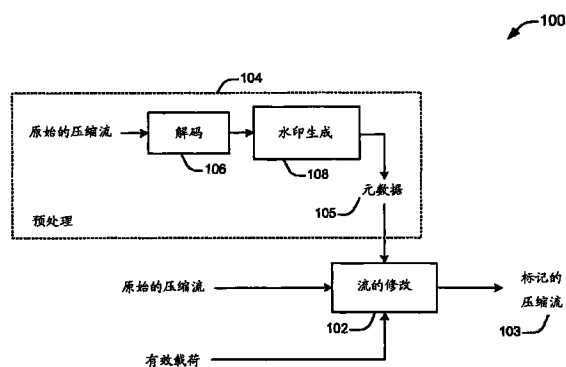
权利要求书3页 说明书16页 附图10页

(54) 发明名称

修改变码的比特流

(57) 摘要

实施方案可以涉及修改或使用编码的比特流、或从数据恢复水印信息的各个方面。在一个实施方案中,修改变码的比特流以允许对编码的比特流施加水印,而不改变比特流中的任何其他语法元素的编码。这是通过标识改变编码的语法元素而不破坏随后的编码的元素的解译所必须的条件来执行的。如果修改的值生成与原始值将生成的解码变量相同的解码变量,则利用另一编码值替换编码的语法元素。特定的实施方案集中在使用熵编码来编码语法元素,包括例如基于上下文的自适应二进制算术编码(CABAC)。无论编码或水印的类型,各种实施方案均可以提供恢复水印信息。特定的实施方案使用检测数据和特征信息来恢复水印信息。



1. 一种用于确定替换数据的方法,其包括:

访问 (252) 算术编码的数据,所述编码的数据至少包括第一部分和第二部分,其中所述第二部分是可解码的以产生基于所述第一部分以及所述第二部分的结果;以及

确定 (254) 修改的算术编码的第一部分,以使所述第二部分是可解码的以产生基于所述第二部分和所述修改的算术编码的第一部分的结果;

其中上述确定还包括确定用于生成至少一个 CABAC 状态值的修改的算术编码的第一部分,所述至少一个 CABAC 状态值与所述第一部分的至少一个状态值相同。

2. 根据权利要求 1 的方法,其进一步包括利用所述修改的算术编码的第一部分替换 (256) 所述第一部分。

3. 根据权利要求 1 的方法,其中所述修改的算术编码的第一部分包括具有与所述第一部分的比特长度相等的比特长度的值。

4. 根据权利要求 1 的方法,其中对所述第一部分的解码产生第一结果,对所述修改的算术编码的第一部分的解码产生与所述第一结果不同的修改的第一结果。

5. 根据权利要求 4 的方法,其中确定所述修改的算术编码的第一部分的步骤基于提供具有与所述第一结果的不同的修改的第一结果,所述不同 (1) 从观众的角度是感觉不到的,并且 (2) 是可由处理设备检测的。

6. 根据权利要求 1 的方法,其中所述修改的算术编码的第一部分适于通过利用所述修改的算术编码的第一部分替换所述第一部分以对编码的数据加水印。

7. 根据权利要求 1 的方法,其中所述结果是根据对所述第一部分的编码产生的状态基于第一部分而得到的。

8. 根据权利要求 1 的方法,其中所述第一部分包括 CABAC 编码的比特流中的语法元素。

9. 根据权利要求 8 的方法,其中确定步骤进一步包括:

确定 (302-304, 402-404) 所述语法元素的所有可能的替代的替换值;以及

对于每个可能的替代的替换值,确定 (308, 408) 该替换值是否生成所述语法元素的相同的状态变量值;以及

保存生成所述语法元素的相同的状态变量值的替代的替换值的列表。

10. 根据权利要求 9 的方法,其进一步包括:在确定所述替换值是否生成相同的状态变量之前对替代的替换值编码 (306)。

11. 根据权利要求 1 的方法,其中所述第一部分包括编码的语法元素,并且其中所述修改的算术编码的第一部分包括具有与所述第一部分的比特长度相等的比特长度的值,所述修改的算术编码的第一部分生成 CABAC 编码器中的 L 值,所述 L 值与由第一部分将生成的 L 值相等。

12. 根据权利要求 1 的方法,其中所述第二部分包括多于一个的编码的语法元素,每个编码的语法元素具有对应的状态变量值,其中所述修改的算术编码的第一部分维持该状态变量值以用于所有随后的第二部分语法元素的所有解码。

13. 根据权利要求 12 的方法,其中所述修改的算术编码的第一部分不影响数据集合中任何其他编码的语法元素的值。

14. 根据权利要求 1 的方法,其中:

所述编码的数据包括熵编码的数据;

所述第一部分包括语法元素 ;以及

所述修改的算术编码的第一部分包括语法元素的修改。

15. 一种用于确定替换数据的装置,所述装置包括 :

用于访问算术编码的数据的部件 (106),所述编码的数据至少包括两个部分,其中第二部分是可解码的以产生结果,该结果基于第一部分以及所述第二部分 ;以及

用于确定修改的算术编码的第一部分的部件 (108),以使所述第二部分是可解码的以产生基于所述第二部分和所述修改的算术编码的第一部分的结果 ;

其中用于确定的部件 (108) 包括用于确定用以生成至少一个 CABAC 状态值的修改的算术编码的第一部分的部件,所述至少一个 CABAC 状态值与所述第一部分的至少一个状态值相同。

16. 一种用于替换编码数据的方法,其包括 :

访问 (262) 算术编码的比特流,所述编码的比特流至少包括第一部分和第二部分,第二部分是可解码的以产生基于第一部分以及所述第二部分的结果 ;以及

利用算术编码的替换值替换 (268) 所述第一部分以产生修改的编码的比特流,对于所述修改的编码的比特流,对所述第二部分的解码将仍然产生所述结果 ;

其中所述算术编码的比特流为 CABAC 编码的比特流,并且所述算术编码的替换值和所述第一部分产生至少一个相同的 CABAC 状态值。

17. 根据权利要求 16 的方法,其进一步包括 :

访问 (264) 第一部分的位置的指示符 ;以及

访问 (266) 所述第一部分的算术编码的替换值,使得利用所述算术编码的替换值替换所述第一部分产生修改的编码的比特流,对于所述修改的编码的比特流,对所述第二部分的解码将仍然产生所述结果。

18. 根据权利要求 16 的方法,其进一步包括访问信息段,并且其中替换所述第一部分的步骤包括基于所访问的信息段来替换所述第一部分。

19. 根据权利要求 18 的方法,其中替换所述第一部分是对编码的比特流加水印的一部分,并且所访问的信息段包括有效载荷信息。

20. 根据权利要求 19 的方法,其中所述有效载荷信息的值指示所述第一部分是否要被替换。

21. 根据权利要求 19 的方法,其中所述有效载荷信息的值指示包括所述算术编码的替换值的多个潜在的替换值中的哪一个将替换所述第一部分。

22. 根据权利要求 16 的方法,其中所访问的编码的比特流包括 CABAC 编码的比特流,并且所述第一部分包括 CABAC 编码的比特流中至少一个编码的语法元素。

23. 根据权利要求 17 的方法,其中所访问的指示符包括元数据。

24. 根据权利要求 16 的方法,其中 :

所述第二部分包括多于一个的语法元素,并且所述多于一个的语法元素中的每一个语法元素具有在对所述修改的编码的比特流的解码期间所产生的对应的状态变量值,并且

所述第一部分的算术编码的替换值将所述多于一个的语法元素中的每一个语法元素的状态变量值维持为与如果在所述修改的编码的比特流中使用所述第一部分而不是使用所述算术编码的替换值时的情况相同。

25. 一种用于替换编码数据的装置,所述装置包括:

用于访问 (262) 算术编码的比特流的部件,所述编码的比特流至少包括第一部分和第二部分,第二部分是可解码的以产生基于第一部分以及所述第二部分的结果;

用于利用算术编码的替换值替换 (268) 所述第一部分以产生修改的编码的比特流的部件,对于所述修改的编码的比特流,对所述第二部分的解码将仍然产生所述结果;以及

存储器,用于存储 (i) 所述算术编码的替换值以及 (ii) 标识解码的图像中用于恢复由所述算术编码的替换值指示的一个或者多个加水印比特的位置的位置信息;

其中所述算术编码的比特流为 CABAC 编码的比特流,并且所述算术编码的替换值和所述第一部分产生至少一个相同的 CABAC 状态值。

26. 一种用于解码数据的方法,其包括:

访问 (710) 修改的算术编码的数据集合,所述修改的算术编码的数据集合包括修改的第一部分和第二部分,所述修改的第一部分由对算术编码的第一部分的修改产生,其中所述第二部分是可解码的以产生结果,并且该结果是基于所述修改的第一部分以及所述第二部分的;

对所述第二部分解码 (730) 以产生所述结果,其中所述结果与如果基于所述第二部分和所述算术编码的第一部分而不是基于所述第二部分和所述修改的第一部分来执行对所述第二部分的解码时的情况相同;以及

对所述修改的第一部分解码 (720);

其中所访问的修改的数据集合包括 CABAC 编码的比特流,并且所述第一部分包括至少一个编码的语法元素;以及

修改的第一部分包括修改的语法元素,并且修改的语法元素产生与所述算术编码的第一部分所产生的 CABAC 编码的比特流的至少一个相同的解码变量。

27. 根据权利要求 26 的方法,其中所述修改的第一部分反映基于信息段的值进行的修改,并且该方法进一步包括基于对所述修改的第一部分的解码来确定所述信息段的值。

28. 根据权利要求 26 的方法,其中所述修改的第一部分是在所述修改的数据集合中包括的水印的至少一部分。

29. 根据权利要求 26 的方法,其中所述修改的第一部分包括修改的语法元素,所述修改的语法元素具有与所述第一部分的比特长度相等的比特长度,并且所述修改的第一部分导致与由所述算术编码的第一部分将生成的 L 值相等的 CABAC 编码器中的 L 值。

修改编码的比特流

[0001] 相关申请的交叉引用

[0002] 本申请要求 (1) 在 2007 年 3 月 23 日提交的、标题为“修改编码的比特流”的美国临时申请第 60/919,702 号 (代理明细号 PU070063) 和 (2) 在 2007 年 6 月 14 日提交的、标题为“修改编码的比特流”的美国临时申请第 60/934,634 号 (代理明细号 PU070137) 两者的权益。通过引用在此以其整个并入这两个优先申请。

技术领域

[0003] 本公开提供了涉及例如编码的实施方案。

背景技术

[0004] 加水印 (watermarking) 典型地涉及以这样的方式来修改数据的部分,即,使得可以在以后检测到该水印的方式。可以对包括编码的数据在内的各种不同类型的数据加水印。然而,编码方案 (scheme) 在改进,而现存的修改编码的数据的机制不能提供用于这种改进的编码方案所期望的加水印能力。

发明内容

[0005] 根据一般的方面,访问至少包括第一部分和第二部分的编码的数据。第二部分是可解码的以产生基于第一部分以及第二部分的结果。确定修改的第一部分以使第二部分是可解码的以产生基于第二部分和修改的第一部分的结果。

[0006] 根据另一一般的方面,信息标识编码的数据集合的第一部分的替换值。该编码的数据集合还包括第二部分,并且对第二部分的解码基于第二部分以及基于第一部分。该替换值具有属性,以使:(1) 如果在包括第一部分的编码的数据集合上执行解码,则对第二部分的解码产生特定的解码结果,并且 (2) 如果在包括替换值而不是第一部分的编码的数据集合上执行解码,则对第二部分的解码产生该特定的解码结果。

[0007] 根据另一一般的方面,访问至少包括第一部分和第二部分的编码的比特流,第二部分是可解码的以产生基于第一部分以及第二部分的结果。用替换值来替换第一部分以产生修改的编码的比特流,对于该修改的编码的比特流,对第二部分的解码将仍然产生该结果。

[0008] 根据另一一般的方面,编码的数据包括修改的第一部分和第二部分,修改的第一部分为第一部分的替换。第二部分是可解码的以产生基于修改的第一部分以及第二部分的结果。该结果与如果基于第二部分和第一部分而不是基于第二部分和修改的第一部分来执行对第二部分的解码时的情况是相同的。

[0009] 根据另一一般的方面,访问包括修改的第一部分和第二部分的修改的数据集合。修改的第一部分由第一部分的修改中产生。第二部分是可解码的以产生结果,该结果基于修改的第一部分以及第二部分。第二部分是可解码的以产生该结果,其中该结果与如果基于第二部分和第一部分而不是基于第二部分和修改的第一部分来执行对第二部分的解码

时的情况相同。对修改的第一部分解码。

[0010] 在附图和下面的描述中阐述了一个或多个实施方案的细节。即使在一种特定的方式中描述,应当清楚,可以按各种方式配置或实施该实施方案。例如,实施方案可以被作为方法来执行、或者被实施为配置成执行一组操作的装置、或者被实施为存储用于执行一组操作的指令的装置,或者在信号中实施。结合考虑附图和权利要求书,其他方面和特征将从下面的详细描述中变得明显。

附图说明

- [0011] 图 1 为对熵编码的流加水印的实施方案的框图;
- [0012] 图 2a 为示出在修改编码的比特流中所使用的操作的顺序的框图;
- [0013] 图 2b 为示出在修改编码的比特流中所使用的操作的另一顺序的框图;
- [0014] 图 2c 为用于修改编码的比特流的方法的框图;
- [0015] 图 2d 为用于修改编码的比特流的另一方法的框图;
- [0016] 图 3 为用于修改目标语法元素 (target syntax element) 的方法的流程图;
- [0017] 图 4 为用于修改被按照旁路模式 (bypass mode) 编码的目标语法元素的方法的流程图;
- [0018] 图 5 为用于修改目标语法元素的另一方法的流程图;
- [0019] 图 6 为用于替换编码的比特块的方法的流程图;
- [0020] 图 7 为用于对修改的比特流解码的方法的流程图;
- [0021] 图 8 为用于检测水印信息的方法的流程图;
- [0022] 图 9 为用于检测水印信息的另一方法的流程图;

具体实施方式

[0023] 至少一个实施方案直接修改熵编码的比特流,而不要求熵解码和重新编码的步骤。进一步地,提供了实施方案,在所述实施方案中熵编码为算术编码,并且更具体地,在所述实施方案中使用被称作基于上下文 (context) 的自适应二进制算术编码 (CABAC) 的算术编码方案。CABAC 的特征可以表现为随后的编码序列依赖于前面的编码序列的、具有“记忆”的编码方案。因而,在实施方案中,以这样的方式修改给定的编码序列,以使随后的编码序列仍然可以被正确地解码。另外,提供了直接对熵编码的比特流施加水印而不要求熵解码和重新编码 (re-encode) 的步骤的实施方案。至少一个加水印实施方案修改 CABAC 编码的比特流。CABAC 在 H. 264/AVC 标准中被广泛地使用。

[0024] 在许多应用中,存在对压缩的视频流加水印的增长的市场需求。早期的方法对流进行解压缩、在像素域中施加水印,然后重新压缩。早期的改进为使用来自原始的 (original) 压缩流的信息来简化重新压缩。这是利用通过施加熵解码和解析 (parse) 编码的比特流来部分地解压缩流的技术而得到了改善。在熵解码之后,加水印算法直接在诸如系数和运动矢量之类的语法元素上操作。为表示水印数据,在对流进行修改之后,将施加熵编码。我们在本公开中提出了实施方案,该实施方案直接在熵编码的比特流上施加水印而无需熵解码和重新编码的步骤。

[0025] 对 CABAC 流加水印涉及改变 CABAC 流中的编码的语法元素。但是,如上所述,期望

通过修改 CABAC 比特流的编码的语法元素而不改变任何其他语法元素的编码来施加水印。这是困难的,是因为 CABAC 流中的每个元素的值影响后面的值的解译 (interpretation)。通常,改变编码的语法元素将导致所有未来编码的元素被解码器误解译或不可解码。

[0026] 一些应用具有对编码的比特流所进行的改变不可以改变比特流的长度的附加约束 (constraint)。在极限情况,要求只在替代值的比特长度与原始值的比特长度相同的情况下,才用替代值替换编码的元素。

[0027] 首先,我们讨论一种情形,在其中可以利用不同的比特串 (string of bits) 来替换 CABAC 编码的 H. 264/AVC 流中的比特串,以使修改只影响目标语法元素,而不干涉任何其他编码的语法元素的正确解译。在这样的情形中,不要求替换比特串与原始比特串的长度相同。H. 264/AVC 比特流中不是所有语法元素都是 CABAC 编码的,并且修改这些非 CABAC 元素或许是可能的。但是,在这里所描述的至少一个实施方案具体地集中在利用 CABAC 编码的那些语法元素上。

[0028] 第二,我们描述并示出在不要求替换比特串与原始比特串的长度相同时可以使用的方法。

[0029] 第三,我们描述并示出怎样如上所述的在 CABAC 编码的比特流中修改语法元素以便将信息嵌入 (embed) 比特流。进一步地,可以通过分析修改的比特流或通过把比特流解码为像素域图像的序列并分析这些图像,来在以后恢复该信息。

[0030] 最后,我们描述并示出可以怎样利用旁路模式来改善编码的比特流的水印嵌入处理的效率。

[0031] 提供 CABAC、旁路模式和加水印的简要讨论来辅助对多种实施方案的理解。这些讨论经常是特定的。但是,这些讨论不是旨在是完全的、并且进一步地,这些讨论中的每个细节可以不一定应用于本申请中所描述的所有实施方案。另外,这些讨论包括本领域技术人员所知的大量材料。但是,这些讨论将总是包括新颖的材料或材料的组织,尽管可能没有具体地指出这些例子。这些讨论不是旨在限制本申请的范围 (breadth)。相反,这些讨论提供许多可能的上下文中的一个或多个的具体的上下文,来辅助读者理解该实施方案。

[0032] CABAC 讨论

[0033] 许多视频压缩方法的最后步骤为压缩数据的无损熵编码。在 H. 264/AVC 中,CABAC 普遍地被用于熵编码。CABAC 是通过维持每个语法元素类型的各自的上下文以及通过使这些上下文与每个编码元素相适应、来实现改善的压缩性能的算术编码方案。这表示了典型的算术编码方案的许多扩展 (extension)。首先 CABAC 被设计用于算术地编码二进制数据。换言之,每个码元为 0 或 1。在算术编码之前首先将本来 (inherently) 不是二进制的语法元素的值进行二进制化。这意味着只有 0 的可能性和 1 的可能性。二进制化的具体方法旨在优化随后的算术编码的效率。每个二进制化的语法元素被称作二进制串 (bin string)。第二个扩展为编码是自适应的 (adaptive)。当处理二进制串的每个比特时,允许改变控制编码的解码变量。最后,允许该自适应是基于上下文 (context-based) 的。与每个比特相关联的是由许多变量组成的上下文。一些上下文变量在许多上下文之间被共享,而一些专用于单个上下文。一个语法元素的编码导致关联的上下文进行适应。H. 264/AVC 定义了编码和解码期间维持的 460 个分开 (separate) 的上下文。

[0034] 上面使用的术语“解码变量”指两个变量集合。第一集合被称作“上下文变量”,

其包含表示可能性 0 或 1 的两个变量（由对这些码元的哪一个的标识是最小可能 LPS、以及指示该 LPS 下次 (next) 在比特流中出现的可能性的、可能性表格中的可能性状态指数 (index) I 来表示）。每个上下文与维持上下文变量的集合的上下文指数相关联。第二集合被称作“状态变量”，其包括两个变量以便定义内部 CABAC 范围值（由范围的底部 (bottom) L 和范围的宽度 R 来表示）。在所有语法元素之间共享 L 和 R 变量，同时对于每个上下文维持分开的 LPS 和 I 值。在这里可互换地使用术语“状态变量”、“（多个）状态值”和“状态变量值”。

[0035] 在二进制化之后，取得 (retrieve) 恰当的解码变量并用于对二进制串算术编码。在每个比特被编码（或解码）后，将基于编码（或解码）的比特和前面的上下文值来更新对应的上下文。

[0036] 对于特定的语法元素，预期要被编码的下一码元 (symbol) 为 MPS (最可能的码元是那些不是 LPS 的二进制码元)。如果发生了这种情况，它将导致指数 I 改变以反映 LPS 下次出现的预期或可能性的进一步减小。如果 LPS 出现，则指数将改变以反映 LPS 的可能性的增加。LPS 的可能性不可以超过 50%，因为这将意味着码元不再是最小可能 (least probable) 的。在此情况中，LPS 将改变为另一码元，并且指数将再次表示小于 50% 的可能性。

[0037] 以与所有算术编码处理相同的方式来更新 L 和 R 值。依赖于哪个码元被编码，L 将是当前间隔 (interval) 的下限 (lower boundary)，而 R 将为当前间隔的宽度。

[0038] 通常，不存在导致状态变量（具体地为 L 和 R）相同的改变的两个二进制串，而这将暗示不可能修改算术编码的比特流而不造成解码器误解译所有随后编码的元素。但是 CABAC 要求以固定的长度二进制数来表示 L 和 R。这是通过被称作重新归一化 (renormalization) 的处理来完成的。L 和 R 的重新归一化的具体细节可以在例如 H. 264 标准中找到。

[0039] 该重新归一化处理的一个结果是，多个不同的二进制串引起 L 和 R 的相同的改变变得可能了。至少一个实施方案利用该事实以标识可以用不同的值来替换的语法元素值、同时维持 CABAC 解码器中原始的 L 和 R 状态变量值。

[0040] 旁路模式的讨论

[0041] 在两个码元具有相等的可能性的语法元素的特殊情况中，将不断地更新 LPS 和状态指数，而不能在编码效率方面引入任何获益 (gain)。由指数表示的可能性本质上将是 50%，其意味着范围 R 将经历从一个比特到下一个比特的相同的可预测的改变。

[0042] 为了加速编码/解码处理而不损害 (compromise) 编码效率，标识那些码元被预期为具有相等的可能性的语法元素，并以被称作旁路模式的特殊方式来对待。这里，不使用上下文，并且码元的可能性均被固定为 0.5。结果是无论范围 R 的值是多少，范围 R 在每个解码的二进制后将为其值的一半。重新归一化将使 R 加倍，所以在旁路模式解码中，R 的最终固定长度表示将保持不变。随着新码元被编码，只有范围的底部 L 进行适应。至少一个实施方案利用旁路模式以简化搜索被不同的值可以替换的语法元素的值，同时维持 CABAC 解码器中的原始状态变量值。

[0043] H. 264 标准规定了必须利用旁路模式 CABAC 来编码的三个语法元素。这些语法元素为系数 (coefficient) 的符号、运动矢量差的后缀 (suffix) 和系数数据的后缀。

[0044] 加水印的讨论

[0045] 我们现在提出加水印的一般领域以及在该领域内应用一个或多个所描述的实施方案的更具体的范围的简短讨论。加水印领域的普通技术人员将熟悉这里提出的众多信息。

[0046] 如这里所使用的短语“数字加水印”典型地指根据一些有效载荷 (payload) 数据来修改艺术作品 (典型地为图像、运动图像序列或音频片段 (clip)), 以使下列条件得到满足:

[0047] 1. 作品的修改的版本与作品的原始版本在感觉上不可分辨, 并且

[0048] 2. 以后可以从作品的修改的版本中恢复有效载荷数据。

[0049] 被称作“鲁棒 (robust) 性的数字加水印”的数字加水印的方法的子类 (subclass) 引入了第三条件:

[0050] 3. 可以从作品的修改的版本的失真 (distort) 版本恢复有效载荷, 其中该失真可以是由被修改的作品可能已经受到的通常的信号处理和信号操作 (例如压缩、降噪滤波、色彩增强等) 引入的, 或该失真可以是由敌意 (adversary) 故意 (intentionally) 引入的、以企图使有效载荷数据不可恢复。

[0051] 鲁棒性的数字加水印有许多应用, 其包括但不限于下列应用:

[0052] 1. 所有者的标识: 水印有效载荷标识作品的所有者。

[0053] 2. 复制控制: 水印有效载荷指示与作品相关联的版权。用于浏览、复制、记录、打印、分发或者可以恢复有效载荷数据并将行为限制于版权所允许的那些行为的任何其他行为的设备。

[0054] 3. 交易追踪: 水印有效载荷标识合法地向其分发作品拷贝的接收者。在合法的接收者未接收到进一步分发作品的权利时, 这是有用的。如果发现未授权的作品, 原始内容的所有者可以从未授权的拷贝恢复有效载荷并标识对未授权使用负责的接收者。

[0055] 本领域普通技术人员将看出, 鲁棒性的数字加水印有许多其他应用。

[0056] 可以在“基带”中或在压缩作品上执行加水印。基带图像或运动图像序列例如为像素域表示。基带音频作品例如为音频采样。一些应用要求对压缩作品加水印。在此情况中, 输出也是压缩的作品。压缩的最后步骤典型地为熵编码, 并且解压缩的第一步典型地为熵解码。熵编码/解码处理典型地是无损的。对压缩作品加水印的一个方法为: 首先施加熵解码, 然后施加解压缩以获得基带表示 (represent)。对基带表示加水印, 并且将产生的加水印后的作品进行压缩和熵编码。该方法可能是耗时的, 并且由于重新压缩, 可以产生感觉 (perceptual) 质量的降级。

[0057] 为了改善感觉质量并且减少所需的计算, 可以在解压缩期间节省诸如运动矢量、模式确定、质量因子 (factor) 和其他信息的之类的来自原始压缩作品的信息, 并且在重新压缩期间使用该信息。这意味着重新压缩不需要执行任何运动估计 (因而节省计算和/或时间), 并且原始量化因子和模式确定的使用可以导致改善的感觉质量。

[0058] 出现了一类被称作“压缩域加水印”的加水印算法。这些方法可以被描述为在加水印前执行“部分解码”, 而不是前面所描述的“完全解码”。这里首先将压缩作品熵解码以暴露 (expose) 压缩作品的语法元素。这些语法元素可以包括系数 (例如块 DCT 或小波 (wavelet) 系数)、运动矢量、图片类型、预测模式和许多其他语法元素。然后加水印的算法

直接修改这些语法元素中的一些。最后,将修改的语法元素熵编码以获得修改的压缩作品。

[0059] 在本申请中所描述的至少一个实施方案落入可以被称作“熵编码的流的加水印”的加水印算法的新类。这些方法如图 1 所图示的直接修改熵编码的流。

[0060] 参考图 1,根据有效载荷,通过流修改 102 来直接修改原始压缩流,并且输出是作为结果的标记的压缩流 103。有效载荷例如可以是播放器的序列号、播放器的型号、或是本质上期望用于以后的标识的任何其他信息。向流修改处理告知由元数据 (metadata) 要进行的所有改变的位置以及实际进行改变的位置,所述元数据可以在预处理阶段 104 期间产生。通过将原始压缩流进行解码 106 来产生元数据 105,并产生水印 108。元数据标识流内要进行改变(即用于水印生成)的位置并指示流应当怎样响应不同的有效载荷码元来改变。图 1 示出了特殊情况,在其中通过对压缩作品的解码的语法元素的分析来产生元数据。加水印的方法的一个挑战性的方面是元数据的生成。下面解释怎样对于一个或多个的实施方案完成元数据的生成。

[0061] 前面和图 1 暗示 (suggest) 可以存在预处理阶段 104。指出在其中有三个重要的时间的情形是有用的。第一时间为在何处/何时分析压缩的比特流以生成一些元数据。第二时间为在何处/何时与被称作有效载荷的特定的码元序列一起使用一些或者所有在第一时间处生成的元数据来修改比特流。第三时间为在何处/何时使用在一些或者所有的在第一时间处生成的元数据,或者根本不使用在第一时间处生成的元数据来进行以下操作:分析修改的比特流、或分析通过解压缩修改的比特流而获得的运动图像序列、或分析通过解压缩修改的比特流而获得的运动图像序列的失真版本。在第三时间中的该分析的目的是恢复有效载荷。

[0062] 为了理解上下文,考虑在作品分发之前出现的第一时间、在复制期间(其中每个拷贝使用的有效载荷唯一地标识该拷贝)出现的第二时间以及在已经发现作品的未授权拷贝(在该时间恢复有效载荷以揭示所分发的拷贝中的哪个是未授权拷贝的来源)之后出现的第三时间。该例子只是例示性的,并且不是旨在暗示对这些构思的应用的任何限制。另外,虽然可以存在特定应用的要求,但不存在对第一时间(预处理)和第二时间(嵌入)的不同的技术要求。

[0063] 作为附加的例子,我们再次参考图 1。例如可以在对电影的交付 (delivery) 的授权期间执行预处理阶段。在一种情形中,执行处理以标识对那些已经确定可接受的替换的、编码的元素的位置。将该处理的结果(例如,这样的编码元素的位置以及可接受的替换值)储存在元数据中,所述元数据被包括在编码的电影中。

[0064] 可以在电影播放 (playback) 期间执行流的修改,在其中例如播放器(例如软件播放器、机顶盒播放器或 DVD 播放器)的序列号被用作有效载荷。例如使用有效载荷来确定是否替换所标识的编码元素。例如,如果在元数据中提供单个可接受的替代,则有效载荷序列中的“0”可以对于相关联的编码序列指示“无替换”,而“1”可以指示“替换”。作为另一个例子,可以替换所有标识的位置,而有效载荷可以指示使用所提供的两个替换值(对于通过位置来标识的每个编码序列)中的哪一个。例如“0”可以指示使用第一替换,而“1”可以指示使用第二替换。

[0065] 可以在与发生流的修改的位置不相关的位置处执行有效载荷的恢复。关键是访问包含基于修改的流的数据的文档。例如,修改的流可以被电子地复制并置于可记录的 DVD

上,或可以呈现修改的流然后重新记录和编码,然后置于可记录的 DVD 上。如果获取了该可记录的 DVD,则可以分析该可记录的 DVD 以恢复 (recover) 有效载荷。

[0066] 在 AVC 编码的比特流中的基于 CABAC 的数据嵌入

[0067] 一种对压缩流加水印的方式为根据下列要求 / 条件来改变一个或多个的语法元素的值:

[0068] R1. 修改的压缩流仍然是有效流 (仍然符合于特定的压缩标准)。

[0069] R2. 通过解压缩修改的压缩流获得的运动图像序列与通过解压缩原始压缩流获得的运动图像序列在感觉上不可分辨。

[0070] R3. 对流的修改导致通过解压缩修改的压缩流获得的运动图像序列中的可测量的改变。

[0071] 在压缩技术使用算术编码熵编码技术时,要求 R1 尤其具有挑战性。这是利用 CABAC 的 H. 264/AVC 的情况。

[0072] 我们首先考虑要求 R1,并讨论可以怎样利用不同的比特串来替换 CABAC 编码的 H. 264/AVC 流中的比特串,以使修改的压缩流仍然是有效流。这要求修改只影响目标的语法元素,而不干涉任何其他编码的语法元素的正确解译。在本实施方案中,不要求替换比特串与原始比特串长度相同 (后面讨论该约束)。为了解释的目的,下列描述专注于对被称作“目标元素”的单个语法元素的修改。但是,本领域普通技术人员将看出,可以将相同的构思应用于包括多于一个目标元素的目标元素集合。

[0073] 目标元素具有被二进制化为“原始二进制串”的“原始值”。原始二进制串是 CABAC 编码的,与其他语法元素一起产生比特的“原始编码块”。在 CABAC 编码处理中,原始二进制串将状态变量和与语法相关联的上下文从它们的原始值修改为它们的修改值。

[0074] 在各种所描述的实施方案中,存在修改 CABAC 编码的比特流以符合要求 R1 的两个步骤,但可以以任一顺序执行该步骤。在对这些步骤的讨论中,这些步骤被称作“步骤 1”和“步骤 2”,但这些步骤的序号不是旨在暗示或要求同样的任何特别或特定的执行顺序。图 2a 和 2b 为示出满足要求 R1 的步骤的框图。

[0075] 给定 CABAC 编码的 H. 264/AVC 比特流,步骤 1 将建立从目标元素到原始编码块的映射 (202)。在一个实际实施方案中,这是通过解码比特流以及追踪哪些比特流比特产生哪些语法元素 (和对应的状态变量值) 来完成的。在第二实际实施方案中,CABAC 解码器被应用于压缩的比特流以暴露语法元素。然后由所实行 (instrument) 的 CABAC 编码器来处理这些语法元素以复原 (restore) 原始压缩的比特流。所复原的比特流在此不是所感兴趣的,而作为有用的 CABAC 编码器的实行则是所感兴趣的。所实行的 CABAC 编码器可以用于追踪语法元素和它们在压缩比特流中的最终位置之间的对应关系。

[0076] 步骤 2 为搜索用于目标元素的一个或多个的可接受的替代值 (204)。搜索可接受的替代值的一种方式检查所有可能的替代值并确定哪些 (如果有) 是可接受的。每个可能的替代语法值将被二进制化为替代的二进制串。与其他语法元素一起,替代二进制串被 CABAC 编码以产生替代的编码的比特块。在 CABAC 编码处理中,替代二进制串将状态变量以及所影响的上下文从它们的原始值修改为它们的替代值。如果所有的替代状态变量值和所影响的上下文与所有修改的状态变量值和所影响的上下文的等同 (identical),则可能的替代语法值被称作“可接受的替代值”。换言之,如果状态变量和所影响的上下文具有与如

果不进行改变时的情况下它们将具有的值相同的值,则该改变将是可接受的。

[0077] 在图 2a 和 2b 中描绘这两个步骤 202、204 的两个不同的顺序,并在图 3 中图示步骤 204 的细节。注意,对于在步骤 2(204) 中发现的至少一个替代语法值的目标元素仅仅要求步骤 1(202) 中的映射(mapping)和分配(assignment)。在图 2b 中的步骤的第二顺序中,对于那些所标识的目标元素仅仅需要执行步骤 1 的映射(202)。

[0078] 图 2c 示出了用于修改编码的比特流的方法 250 的框图。首先,访问具有第一部分和第二部分的编码的比特流(252)。对于第一部分确定替换值(254),并且然后利用所确定的替换值来替换第一部分(256)。一旦替换值替换了第一部分,则发送修改的编码比特流(258)。“发送”可以是例如电子传送或以光盘或主机(master)形式的发送。

[0079] 在这里被称为第二部分的可以是下一邻近的(adjacent)语法元素(以及对应的状态变量值),或可以是可由两个或多个的语法元素和对应的状态变量值组成的比特流的剩余部分。因而,在对于第一部分确定替换值时,可以考虑所有随后的语法元素和他们相应的状态变量值。在第一部分的替代的或所确定的替换不改变或变更第二部分中的其他语法元素的状态变量值时,则第一替换值将提供期望的作为结果的(resulting)对第二部分的解码。

[0080] 确定替换值的操作 254 的特征也可以表现为确定修改的第一部分。在正在被讨论的实施方案中,在第一部分的修改之前,第二部分可首先(originally)解码(解码部分地依赖于未修改的第一部分)以产生特定的结果。进一步地,在第一部分的修改之后,第二部分是可解码的(解码部分地依赖于修改的第一部分)以产生相同的特定结果。

[0081] 与方法 250 相似的另一实施方案涉及两个操作。第一操作为访问至少包括第一部分和第二部分的编码的数据。第二部分是可解码的以产生基于第一部分以及第二部分的结果。第二操作为确定修改的第一部分,以使第二部分是可解码的以产生基于第二部分和修改的第一部分的结果。

[0082] 标识编码元素的位置和可能的替换值的所有数据或部分数据可以被存储在存储设备上,或被电子地发送。一个实施方案为诸如 DVD、硬盘或其它存储设备之类的装置。该装置包括具有存储在其上的信息的处理器可读介质。所存储的信息标识编码的数据集合的第一部分的替换值。编码的数据集合还包括第二部分,并且对第二部分的解码是基于第二部分以及第一部分的。替换值具有属性,以使:(1) 如果在包括第一部分的编码的数据集合上执行解码,则对第二部分的解码产生特定的解码结果,以及(2) 如果在包括替换值而不是第一部分的编码的数据集合上执行解码,则对第二部分的解码产生该特定的解码结果。另一实施方案提供了被格式化为包括存储在该装置的处理器可读介质上的信息的信号。

[0083] 图 2d 示出了用于修改编码的比特流的方法 260 的另一实施方案的框图。首先,访问(262)具有第一部分和第二部分的编码的比特流。然后访问(264)提供关于编码的比特流内第一部分的位置信息的元数据。然后访问(266)第一部分的替换值并且利用所访问的替换值来替换(268)第一部分。一旦替换值替换了第一部分,则发送(270)修改的编码的比特流。“发送”可以是例如电子传送或以光盘或主机(master)形式的发送。

[0084] 与方法 260 相似的另一实施方案涉及两个操作。第一操作是访问包括至少两个部分的编码的比特流。第二部分是可解码的以产生基于第一部分以及第二部分的结果。第二操作是利用替换值替换第一部分以产生修改的编码的比特流,对于所述修改的编码的比特

流,对第二部分的解码仍然产生该结果。

[0085] 附加的实施方案是专注于 (direct to) 诸如方法 260 之类的方法的结果。例如,一个实施方案为诸如 DVD、硬盘或其它包括修改的编码的数据的存储设备之类的装置,而另一实施方案是被格式化为包括这样的修改的编码的数据的信号。更具体地,一个实施方案为包括处理器可读介质的装置,所述处理器可读介质具有存储在其上的、包括修改的第一部分和第二部分的编码的数据。修改的第一部分为第一部分的替换者,其中 (1) 第二部分是可解码的以产生基于修改的第一部分以及第二部分的结果,并且 (2) 该结果与如果基于第二部分和第一部分而不是基于第二部分和修改的第一部分来执行对第二部分的解码时的情况相同。另一实施方案是格式化为包括存储在该装置的处理器可读介质上的编码的数据的信号。

[0086] 图 3 示出了表示用于发现目标元素的所有可接受的替代值的步骤的一个实施方案。首先,提供目标元素的所有潜在的替代 (替换) 值的列表 (302)。确定在列表中是否留下任何替代 (替换) 值 (304)。如果该结果为否,则处理结束 (316)。在列表中存在替代 (替换) 值时,获得下一替代 (替换) 的可能值并且将该语法值进行编码 (306)。然后确定替代 (替换) 解码变量值 (包括状态变量值和上下文) 是否等于修改的解码变量值 (308)。在它们相等时,将替代 (替换) 值保存至可接受的替代值的列表 (312),并且然后将所测试的值从列表中移除 (314)。然后该处理在步骤 304 处再次开始,直至用尽 (exhaust) 潜在的替代 (替换) 值的列表。

[0087] 我们现在考虑第二要求 R2。对于给定的目标元素,我们需要评估 (assess) 或预测利用满足要求 R1 的每个替代值来替换其原始值时对感觉的影响。调用 (recall) 该评估或预测能完全访问压缩流的所有语法元素,并且可以使用这些语法元素来预测根本的 (underlying) 图像的屏蔽效应 (masking effect) 以及人的视觉或听觉系统 (在视频的情况中,我们考虑人的视觉系统) 对该修改的敏感度。这种预测的细节对于感觉建模 (perceptual modeling) 领域的技术人员来说是众所周知的。感觉模型被广泛地引用于压缩文献和加水印的文献这两者中。

[0088] 除使用感觉性 (perceptibility) 的压缩域计算预测之外,我们还可以有能够将流完全地解压缩为运动图像序列的选择 (luxury)。例如,这可以是按照预处理来执行分析的情况。在该情况中,计算预测可以检查实际的像素数据。这可能导致对感觉性的更准确的预测。另外,我们可以通过下列步骤来评估利用替代值替换原始语法元素的感觉性:解压缩原始流以获得参考运动图像序列,然后用替代者替换语法元素值并解压缩产生的流以获得第二运动图像序列。许多众所周知的技术可以用于评估两个运动图像序列之间的差别 (difference) 的感觉性。

[0089] 最后,我们可以有依靠 (resort) 主观评估的选择。观众可以评估参考运动图像序列和修改的运动图像序列之间的差别的感觉性。如果主体 (subject) 不能察觉 (perceive) 该差别,则修改满足要求 R2。

[0090] 这些只是一些可以用于确定所有改变中的哪一些改变满足要求 R1 也满足要求 R2 的方法。

[0091] 我们现在考虑第三要求 R3。使用目标语法元素的替代语法值必须引起 (induce) 一些在以后将修改的流解压缩为运动图像序列时可测量的改变。存在可以引起的两种可测

量的改变：“直接的”与“间接的”

[0092] 对于“直接改变”，语法元素直接对应于运动图像序列的可测量伪像 (artifact)。例如亮度 (luminance) 块 DC 系数的修改将直接导致对应的解压缩块的平均亮度的可测量改变。另一方面，对于“间接改变”，图像序列中所测量的伪像只间接地与流中的修改相关。例如，运动矢量的修改将导致错误的块被用作预测，并且将因此导致对应的解压缩块中的不正确的像素数据。确定使用哪个运动矢量可能是困难的，而使用不同的运动矢量可以影响其他可测量的伪像。运动矢量可以用于产生具有更高或更低的平均亮度的重构的 (reconstructed) 块。

[0093] 要求 R3 的可测量的改变将辅助标识位置，并且辅助观察进行了什么改变，以便恢复有效载荷。这在获得了盗版拷贝时尤其有用。通过“观察”进行了的改变，可以获得盗版拷贝的来源。

[0094] 在一个实施方案中，确定哪些运动矢量语法元素具有满足要求 R1 和 R2 的至少两个替代语法值，使得使用替代语法值中的一个将产生具有较高的平均亮度的重构的块，而使用不同的替代语法值将产生具有较低的平均亮度的重构的块。与这两个替代值中的每一个的 CABAC 编码对应的编码比特块，与 CABAC 流内的比特位置一起被作为元数据发送至嵌入器 (embedder)。基于对应的有效载荷比特的值，嵌入器将利用与一个替代比特块或另一个替代比特块对应的比特块来替换原始比特块。例如，如果有效载荷比特为“0”，则嵌入器选择或挑选与降低对应的块中的平均亮度的语法值相关联的比特块；如果有效载荷比特为“1”，则嵌入器选择与增加对应的块中的平均亮度的语法值相关联的比特块。明显地，对于每个要嵌入的有效载荷比特，我们需要至少一个满足所有要求 R1、R2 和 R3 的目标元素。

[0095] 元数据还应当包含其中将引入亮度变化的解压缩的运动图像序列中的帧数和块数。为了进一步的鲁棒性，预处理阶段还可以在元数据中存储该块的原始的平均亮度。在恢复（例如解码）时使用该信息以标识修改的块，并将其平均亮度与存储在元数据中的原始值比较。在给出的例子中，高于原始值的平均亮度将暗示 (imply) 对应的有效载荷比特为“1”，而低于原始值的平均亮度将暗示对应的有效载荷比特为“0”。

[0096] 比特长度约束

[0097] 当存在替代的编码的比特块与原始编码的比特块具有相同的长度的附加约束时，我们可以简单地将该约束添加至要求 R1。换言之，如果所有的状态变量和所影响的上下文保持与其将是相同的，并且如果产生的编码块长度与原始长度相同，则该改变将是可接受的。例如，再次参考图 3，可以修改判定块 308 以便不仅确定解码变量值是否保持相同，而且确定产生的编码块是否具有与原始长度相同的长度。作为另一例子，再次参考图 3，可以修改块 302 以提供导致编码的比特块具有与原始长度相同的长度的、目标元素的所有潜在替换值的列表。

[0098] 前面所描述的添加比特长度约束的方法是一个可能的方法。但是，在一些应用中，替代的方法可以是合适的。我们现在呈现旨在限制搜索可接受的替代值的不同方法。在图 5 中所描述的该实施方案中，第一步骤为建立从所有语法元素到比特流中它们对应的编码比特的映射 (502)。在第二步骤中，标识来自列表的一个目标元素，并且标识比特流中表示该元素的编码的比特块 (504)。

[0099] 因为 CABAC 可以分配比特片段 (fraction) 以表示语法元素，所以存在表示不同的

语法元素的比特块可以重叠 (overlap) 的可能性。第三步骤为标识编码的比特流中其表示与要被修改的元素的表示相重叠的任何其他语法元素 (506)。这些语法元素在这里被称作重叠语法元素。确定具有相同比特长度的所有替代可能的比特组合的列表 (508), 并且随后的剩余步骤 (即, 步骤 510-522) 分别与步骤 304-316 实质上相同, 所述随后的剩余步骤用于确定将使用所标识的具有相同比特长度的替代可能的比特组合中的哪一些, 所述步骤 304-316 被描述为确定将使用目标元素的潜在替代值中的哪一些。一个不同之处为块 514 包括确定替代的比特组合是否满足任何重叠语法元素的解码值保持不变的条件, 以及确定解码变量值是否保持不变。

[0100] 如果编码比特块具有长度 n , 则存在其可以表示的 $2n$ 个不同的可能值。我们搜索任何符合下列两个准则 (criteria) 的这样的值:

[0101] 1. 重叠语法元素的解码值保持不变; 以及

[0102] 2. 留下由编码比特块所改变的所有的状态变量和上下文变量与未修改的编码比特块的情况时它们的相同的值。

[0103] 这里我们已经将搜索限制为只是导致替代的编码比特块的尺寸与原始的编码比特块尺寸相同尺寸的那些替代值。

[0104] 实施方案不需要专注于重叠语法元素和比特长度约束这两者。相反, 某些实施方案可以只专注于重叠语法元素, 或只专注于比特长度约束, 而不是专注于另一考虑。

[0105] 进一步地, 各种实施方案可以一起在多个语法元素 (例如数据流中邻近的 (contiguous) 语法元素) 上运行。例如, 实施方案将多个邻近的语法元素 (重叠的或非重叠的) 视为单个组, 并且确定组的替换, 以使在该组的结尾处 (但不一定在组内的语法元素之间) 的解码变量值保持不变。一个这样的实施方案将两个邻近的语法元素考虑为目标元素, 其中第二语法元素为重叠语法元素, 并且允许修改这两个语法元素。

[0106] 信息嵌入

[0107] 上面的讨论描述了一种方法, 该方法用于分析目标语法元素以及标识所有 (如果有) 可以通过利用替代比特块以替换编码的比特流中的比特块来代替 (substitute) 的可接受的替代值。通过检查 H. 264/AVC 编码序列中的所有语法元素, 我们可以建立对其至少存在一个可接受的替代值的语法元素的列表。这是“可改变的语法元素”列表。对于每个可被改变的语法元素, 我们可以建立可接受的替代值的列表。

[0108] 已经呈现了可以怎样使用该方法在比特流中嵌入信息的例子。信息嵌入的目标是根据有效载荷来修改所有可改变的语法元素列表上的语法元素的子集, 将它们的值从它们的原始值改变为所列的可接受的替代值中的一个。在一个实际实施方案中, 这是按如图 6 所示的三个步骤来完成的。

[0109] 参考图 6, 存在处理 600。在处理 600 的第一步骤中, 选择可改变的语法元素的子集 (602)。该子集的选择可以是基于对该改变将在解码图像上具有的对感觉的影响的估计。该子集的选择可以是基于对通过信号处理或通过故意篡改 (tamper) 而修改图像之后, 对检测解码图像中的改变的难度的估计。该子集的选择可以受要求以下内容的实施方案的影响: 例如对改变后的块中的连续 (consecutive) 比特的数目的限制、或对必须在任意两个改变的块之间的未改变比特的最小数目的限制。在优选实施方案中, 在预处理期间执行该第一步骤。

[0110] 在第二步骤中,为了以后的使用而保存元数据(604)。该元数据包括“嵌入数据”和“检测数据”。嵌入数据是在压缩比特流中可以发现原始的编码比特块的位置、并且是两个比特块,如果对应的有效载荷比特为“0”则一个比特块被用于替换块,而如果对应的有效载荷比特为“1”则另一个比特块被用于替换块。在一个实施方案中,这些块中的一个块自身为原始的比特块,而另一个块为替代比特块。在该情况中,可改变的语法元素的列表可以包括对其只存在一个可接受的替代值的那些语法元素。在另一实施方案中,两个块都是对应于不同的语法值的替代比特块。在该情况中,可改变的语法元素的列表可以只包括对其存在至少两个可接受的替代值的那些语法元素。

[0111] 检测数据包括将用于嵌入的可改变的元素的集合以及这些元素的原始值。它还包括可以在解压缩的运动图像序列中检测比特流修改的位置。这可以被指定为帧数与块数或者宏块数或者像素位置或者恢复所需要的任何其他定位符(locator)。检测数据还可以包括如果不进行改变时将被测量的检测测量的值。换言之,这可以包括检测测量的原始值。在前面的例子中,检测测量为特定块的平均亮度,而检测数据可以包括该块的原始平均亮度。在一个实施方案中,在预处理期间执行该第二步骤。

[0112] 为了在亮度或对比度的全局或局域的改变上增加加水印的技术的鲁棒性,还可以对于不期望被水印处理修改的块保存检测数据。例如,检测数据可以包括不被水印改变的块的原始平均亮度。然后检测器可以作为参考使用这些检测数据,以确定整个图像或至少在参考区域的图像的部分是否经历了亮度的改变。如果参考块中的所测量的亮度不与检测数据中所记录的匹配,则可以在有效载荷的恢复之前进行补偿。

[0113] 在第三步骤中,使用嵌入数据和有效载荷来利用不同的比特块替换对应于可改变的元素的原始的编码比特块(606)。我们考虑两种情况。在第一情况中,如果要被表示的有效载荷比特具有一个码元值,例如“1”,则替换比特块为对应于可接受的替代值中的一个值的编码比特块。如果要被表示的有效载荷比特具有另一码元值,例如“0”,则替换的编码比特块为原始编码比特块。在第二情况中,子集的选择被限制为对其存在至少两个可接受的替代值的元素。在该情况中,如果要被表示的比特为“0”,则替换的编码比特块为对应于可接受的替代值中的一个值的编码比特块,而如果要被表示的比特为“1”,则它是对应于不同的可接受的替代值的编码比特块。这两个替代值与它们所表示的标签“0”或“1”一起被存储在检测数据中。在优选实施例中,在嵌入期间执行该第三步骤。

[0114] 可以借助检测数据从 H. 264/AVC 比特流恢复有效载荷。检测数据指示表示隐藏的有效载荷信息的特定的语法元素以及这些元素的原始值。如果嵌入处理使用原始值来表示“0”并使用替换值来表示“1”,则检测器可以将比特流中的值与存储在检测数据中的原始值比较。如果它们匹配,则检测器报告比特“0”。如果它们不匹配,则检测器报告比特“1”。如果嵌入处理利用两个替代值中的一个值来替换原始值,则从检测数据中恢复这两个替代值和它们对应的标签。然后检测器将比特流中的值与两个替代值中的每一个值比较,并且报告对应的标签。

[0115] 还可以从解码的像素域图像来恢复隐藏的数据。该处理特定于在子集选择处理中使用的特定的测量。在一个实际实施方案中,该测量为像素块的平均亮度。从检测数据恢复每个修改的块的原始平均亮度。检测器计算解码的图像序列中的指定的像素块的平均亮度,并将该值与存储在检测数据中的原始值比较。如果计算值高于原始值,则检测器报告比

特“1”。如果计算值低于原始值,则检测器报告比特“0”。

[0116] 如果在解码之后、但是在检测之前通过均匀(uniform)的亮度改变来修改解码图像,则该恢复方法可能有问题。为了针对该问题,可以在检测数据中存储多个参考块的平均亮度。这些是不被嵌入处理所改变的块。检测器可以刚好在恢复之前计算参考块的平均亮度,并且发现任何与隐藏的数据无关的亮度改变。可以通过相应地调整原始值来补偿这些亮度改变。

[0117] 应当清楚,信息嵌入可以包括各种应用。一个这样的应用是其中在数据流中嵌入有效载荷来加水印。其他应用包括嵌入实际上任何期望传送(communicate)的信息。另外,一些信息嵌入应用可以使用不同的要求。例如,可能期望对数据流进行改变以导致可由用户察觉的改变。

[0118] 旁路模式的利用

[0119] H. 264/AVC 指定了三个在旁路模式中被编码的语法元素。对于这些语法元素,在编码期间只修改一个变量。随着比特被编码,只有表示范围(range)底部(bottom)的L值改变。因而,对于旁路模式的语法元素来说,发现产生与原始的相同的解码变量值的替代语法值的可能性较高。在一个实际实施方案中,只考虑运动矢量差的语法元素的后缀。对于这些中的每一个,我们搜索这样的替代值:(1) 导致编码比特块中的相同数目的比特(2) 不影响任何其他语法元素的值并且(3) 在其对应的二进制串被处理之后产生CABAC编码器中的相同的L值。从这些中,我们按照上述的选择用于嵌入的子集(例如参见图6)。通过图4中所描绘的方法400来示出旁路模式、固定长度代替的步骤。这里步骤402-416中的处理分别与之前描述的图3的方法中示出的步骤302-316步骤相同,而例外是:对于CABAC编码,步骤408是确定替代的L等于修改的L,而不是确定替代的解码变量值相对修改的解码变量值(即步骤308)。

[0120] 应当清楚,可以对未修改的比特流进行解码以产生原始语法值,包括目标语法元素和跟随在目标语法元素之后的一个或多个的语法元素的值。如前所述,跟随在目标语法元素之后的一个或多个的语法元素的编码至少部分依赖于目标语法元素的值。在目标语法元素的值被修改之后,产生作为结果的修改的比特流,所述修改的比特流对应于修改的语法值和跟随在现在修改的目标语法元素之后的一个或多个的语法元素的值。该修改的比特流也可以被解码。

[0121] 参考图7,示出了处理700,其可以由例如解码器、DVD播放器或访问并解码修改的比特流的另一设备来执行。修改的比特流的解码产生目标语法元素的修改的值,并且也产生跟随在目标语法元素之后的一个或多个的语法元素的原始值。

[0122] 更具体地,参考处理700,访问修改的数据的集合(710)。修改的数据的集合包括修改的部分和在本实施方案中未修改的另一部分。如上面所指示的,未修改部分是可解码的以产生基于未修改部分以及修改的部分的结果。

[0123] 处理700包括对修改的部分(720)和未修改部分(730)进行解码。对未修改部分的解码产生与如果修改的部分未被修改时的情况相同的结果。

[0124] 参考图8,示出了用于恢复有效载荷信息的,或更一般地,用于检测水印的处理800。处理800包括确定用于检查可能的水印的位置(810)。注意,不一定需要已插入水印,所述水印一般被称为允许随后检测的所修改的或所插入的数据。处理800包括从所确定的

位置访问 (820) 和分析 (830) 数据。处理 800 进一步包括如果任何水印存在时则基于该分析来标识水印信息 (840)。所标识的水印信息可以是例如比特或其他信息单位。可以对于一个或多个的其他位置重复处理 800, 以标识组成有效载荷的一个或多个的附加的比特。因而, 可以使用处理 800 来恢复整个有效载荷。

[0125] 处理 800 的实施方案包括分析修改的比特流以及分析像素域数据。例如, 根据前面所描述的实施方案中的一个, 可以通过修改与目标元素相关联的比特来修改比特流。可以通过访问恰当的比特位置以及在这些位置测试水印 (修改) 来在这样的比特流中检测水印。在检测数据中还可以包括诸如在元数据中发现的比特的位置和修改的值 (或可能的修改的值)。

[0126] 替代地, 或附加地, 这样的修改的比特流可以被解码, 并且可能经受包括重新编码的其他处理。在重新编码的情况中, 不期望这样的重新编码的比特流在与原始修改的比特流中的相同的位置处包括相同的修改的目标元素。但是, 可以将重新编码的比特流解码以产生其中仍然典型地呈现原始修改的作用 (effect) 的像素域的数字表示。因而, 标识其中原始修改具有其影响的帧数和块数的检测数据将是有用的。例如, 最初, 目标元素可以是特定块的运动矢量, 修改运动矢量的影响可以是增加该特定块的平均亮度。尽管对原始修改的比特流进行处理, 但将典型地保留平均亮度的增加。然而, 典型地是运动矢量以及对运动矢量的原始改变将不是严格 (exactly) 相同。

[0127] 参考图 9, 处理 900 描述其中从像素域数据来检测水印信息的实施方案。处理 900 包括确定可能的水印的帧标识符和块标识符 (910), 以及访问 (920) 对应于所确定的帧标识符和块标识符的像素域数据。处理 900 包括确定所访问的数据的特征的值 (930), 以及将所确定的值与基准 (benchmark) 比较 (940)。然后处理 900 包括基于该比较来标识水印信息 (如果存在) (950)。

[0128] 处理 800 或处理 900 的各种实施方案使用检测数据, 该检测数据包括所存储的各种数据段中的一个或多个。例如, 检测数据可以包括标识用于检查可能的水印的位置的位置信息。该位置信息可以包括例如帧标识符和 / 或块标识符。

[0129] 检测数据可以包括例如用于特征的基准值。基准可以是例如位置被修改之前的特征的值。例如可以将基准与所访问的位置的特征的实际值 (新基准) 比较 (820) 以提供对改变的指示。例如, 该比较可以指示该值 (新基准) 是否随着修改的结果而增加或减少。

[0130] 检测数据可以包括例如来自不同的 (未加水印的) 位置的特征的基值。这样的基值例如可以是未加水印的位置的平均亮度。如前面所解释的, 基值可以用于确定是否存在对所受影响不止是加水印的位置的数据的改变。在某些的情形中, 可以是全局 (global) 的改变 (例如, 对整个帧的改变)。例如, 可以改变整个帧的平均亮度。然后可以将基值与在未加水印位置的特征的新值 (新基值) 比较。如果该差大于阈值, 则可以确定发生了全局改变。替代地, 可以仅仅将该差作为全局的差, 并且可以将该差添加到为 (可能的) 加水印的位置所确定的值 (新基准) 以解决 (account for) 全局的改变。注意, 在该替代方案中, 可以 (替代地) 将该差添加到基准而不是添加到该值 (新基准)。进一步地, 在某些的实施方案中, 可以在基值与新基值之间确定比率 (ratio), 并且可以将基准或新基准乘以该比率。

[0131] 清楚地, 可以由包括处理设备的各种设备执行处理 800。这样的设备还可以包括用

于存储诸如例如所访问的数据、基准值、基值和可能水印信息的（多个）位置之类的数据的存储设备。进一步地，同样可以在指令的集合中实施处理 800。

[0132] 应当清楚，语法元素可以表示各种不同的信息类型中一个或多个。例如，语法元素可以是运动矢量、残留 (residue) DCT (离散余弦变换) 系数、宏块类型、量化参数，或者标志 (flag)。据此，不需要所有的语法元素具有相同的长度，而是可以具有各种不同的长度。本申请中所公开的各种实施方案可以在任何类型的语法元素上运行 (work)，而无论由语法元素所表示的是什么类型的信息、并且无论语法元素的尺寸 / 长度。

[0133] 另外，如前面所指示的，可以在多个的语法元素（例如数据流中邻近的语法元素）上一起运行各种实施方案。例如，实施方案将多个邻近的语法元素（重叠的或非重叠的）视为单个组，并且确定组的替换，以使在该组结尾处（但不一定在组内的语法元素之间）处的解码变量值保持不变。

[0134] 可以在例如方法或处理、装置、或软件程序中实施在这里所描述的实施方案。尽管只在实施方案的单个形式的环境中来讨论（例如只作为方法来讨论），但也可以以其他形式（例如装置或程序）来实施具有所讨论的特征的实施方案。可以在例如恰当的硬件、软件和固件中实施装置。可以在例如诸如处理器之类的装置中实施该方法，所述处理器指包括例如计算机、微处理器、集成电路或可编程逻辑器件的一般的处理设备。处理设备还包括通信设备，诸如例如计算机、蜂窝电话、便携 / 个人数字助理 (PDA) 和其他便于终端用户之间的信息通信的设备。

[0135] 在这里所描述的各种处理和特征的实施方案可以在多种不同的设施或应用中实施，具体地，例如与数据编码和解码相关联的设施或应用中实施。设施的例子包括视频编码器、视频解码器、视频编解码器、网页 (web) 服务器、机顶盒、笔记本电脑、个人计算机、蜂窝电话、PDA 和其他通信设备。应当清楚，该设备可以是移动的或甚至是安装在移动车辆中的。

[0136] 另外，可以由指令来实施该方法，所述指令由处理器执行，并且可以在处理器可读介质上存储这样的指令，所述处理器可读介质是诸如例如集成电路、软件载体 (software carrier)、或其他诸如例如硬盘、密致盘 (compact diskette)、随机存取存储器 (“RAM”) 或只读存储器 (“ROM”) 之类的存储设备。指令可以形成在处理器可读介质上有形地实施的应用程序。指令可以处于例如硬件、固件、软件或其组合中。可以在例如操作系统、分离的应用或两者的结合中发现指令。因此，例如，处理器的特征可以表现为既是被配置为执行处理的设备、也是包括具有执行处理的指令的计算机可读介质的设备。

[0137] 如对于本领域技术人员来说应当明显的是，实施方案还可以产生被格式化为承载信息的信号，所述信息例如可以被存储或传送。该信息可以包括例如用于执行方法的指令、或由所描述的实施方案中的一个实施方案所产生的数据。这样的信号可以被格式化为例如电磁波（例如使用频谱的射频部分）或基带信号。该格式化可以包括例如对数据流编码以及利用编码的数据流来调制载波。该信号承载的信息可以是例如模拟的或数字的信息。如已知的，可以在各种不同的有线的或无线链路上传送该信号。

[0138] 已经描述了许多实施方案。然而，将理解可以进行各种修改。例如，可以将不同的实施方案的元素合并、补充、修改或移除，以产生其他实施方案。另外，本领域普通技术人员将理解，可以用其他结构和处理来代替所公开的那些结构和处理，并且作为结果的实施方

案将与所公开的实施方案以至少实质上相同的（多个）方式来执行至少实质上相同的（多个）功能，以实现至少实质上相同的（多个）结果。据此，该申请预料了这些和其他实施方案，并且这些和其他实施方案在随后的权利要求书的范围内。

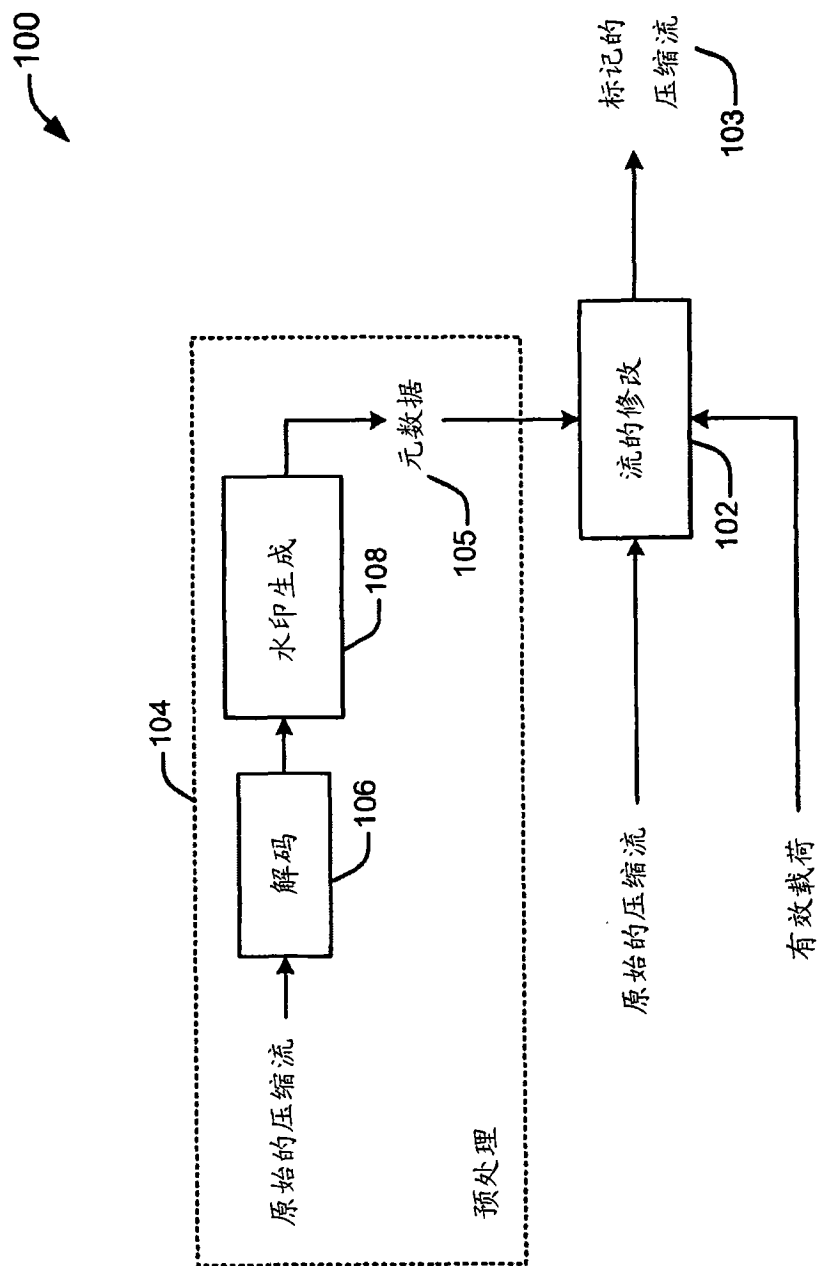


图 1

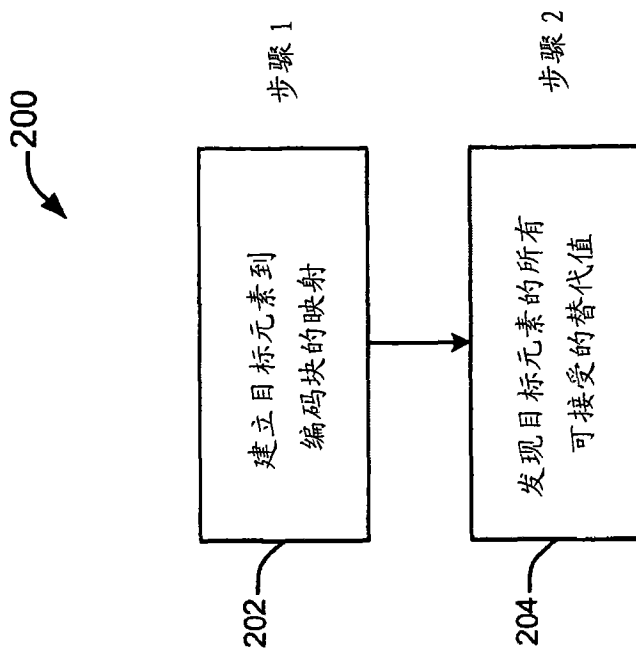


图 2a

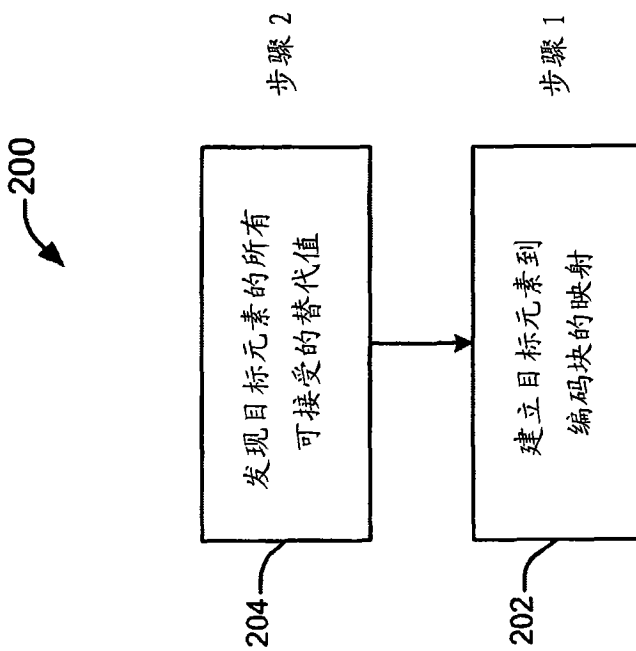


图 2b

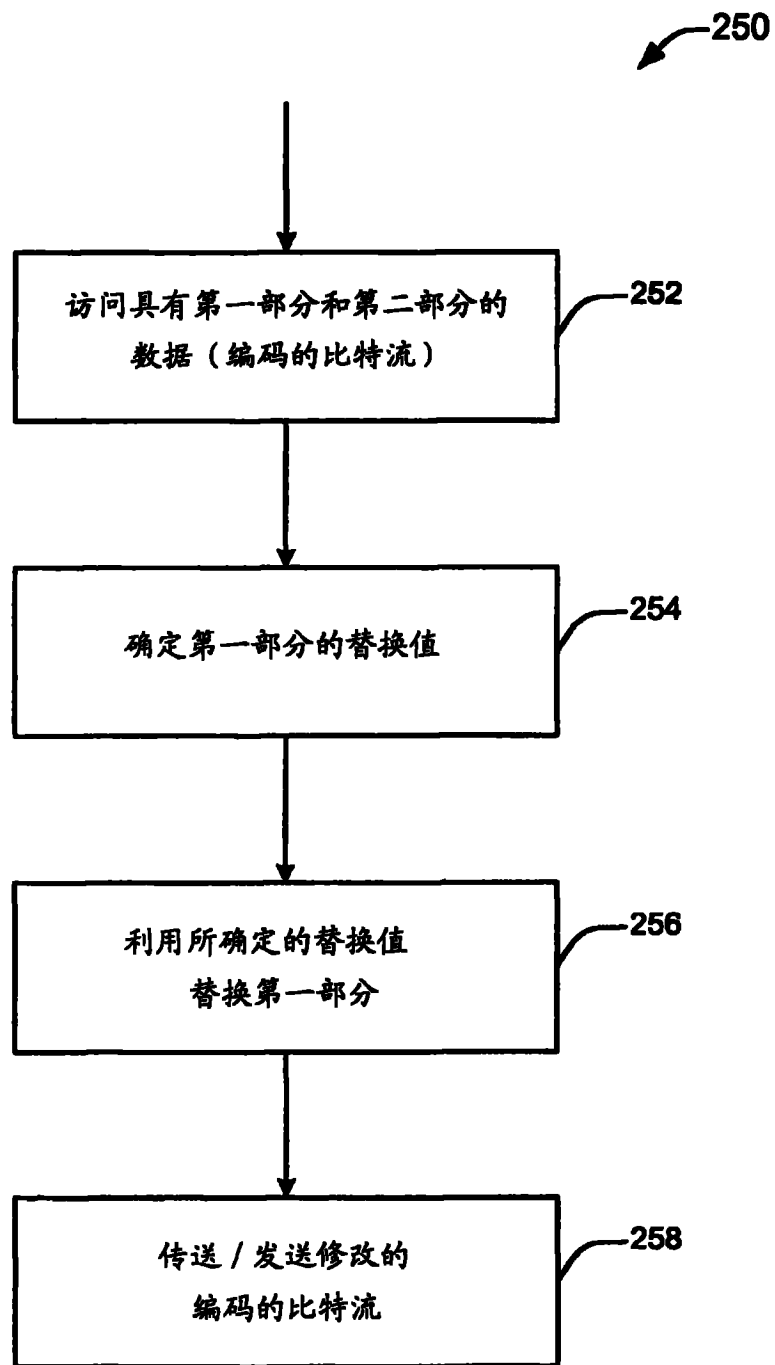


图 2c

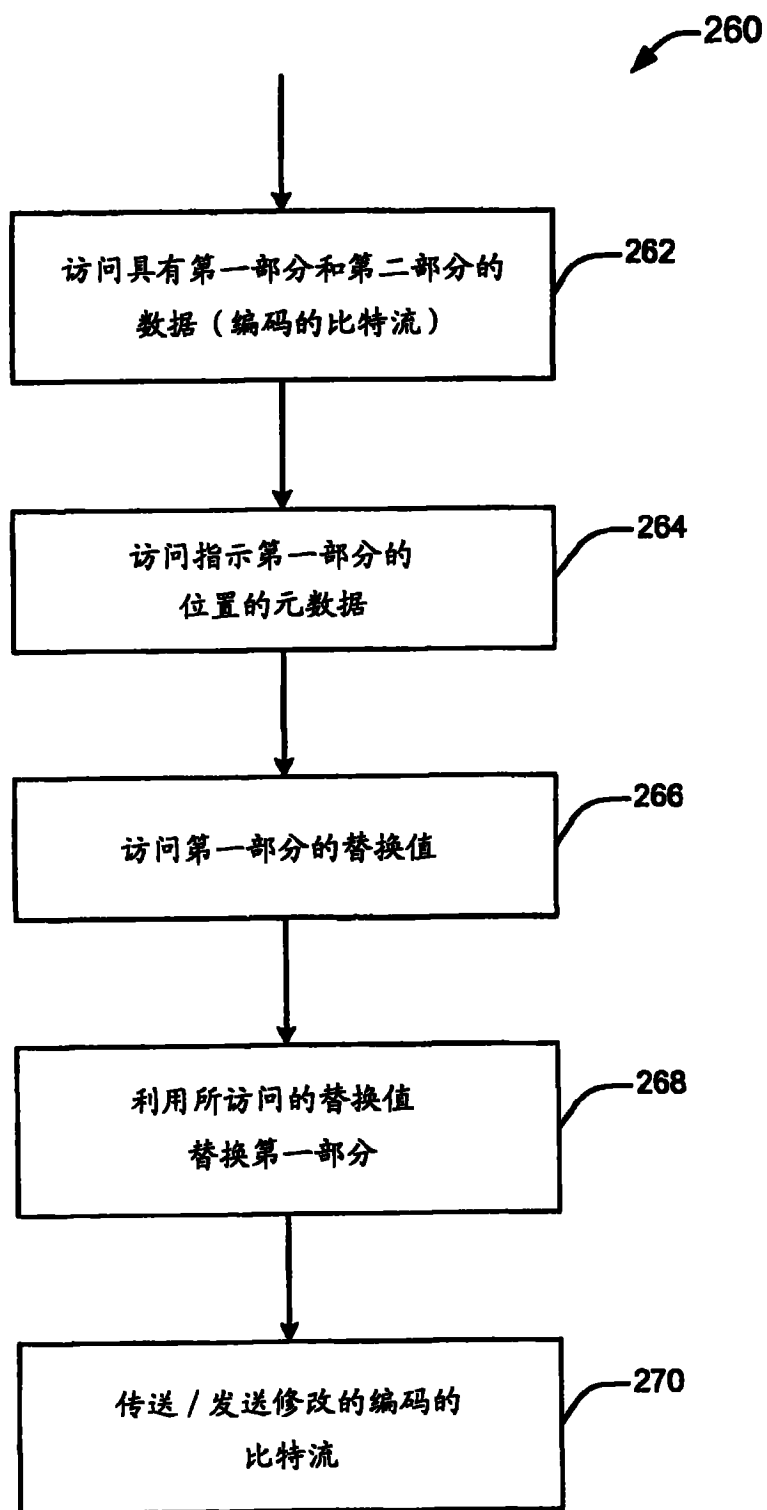


图 2d

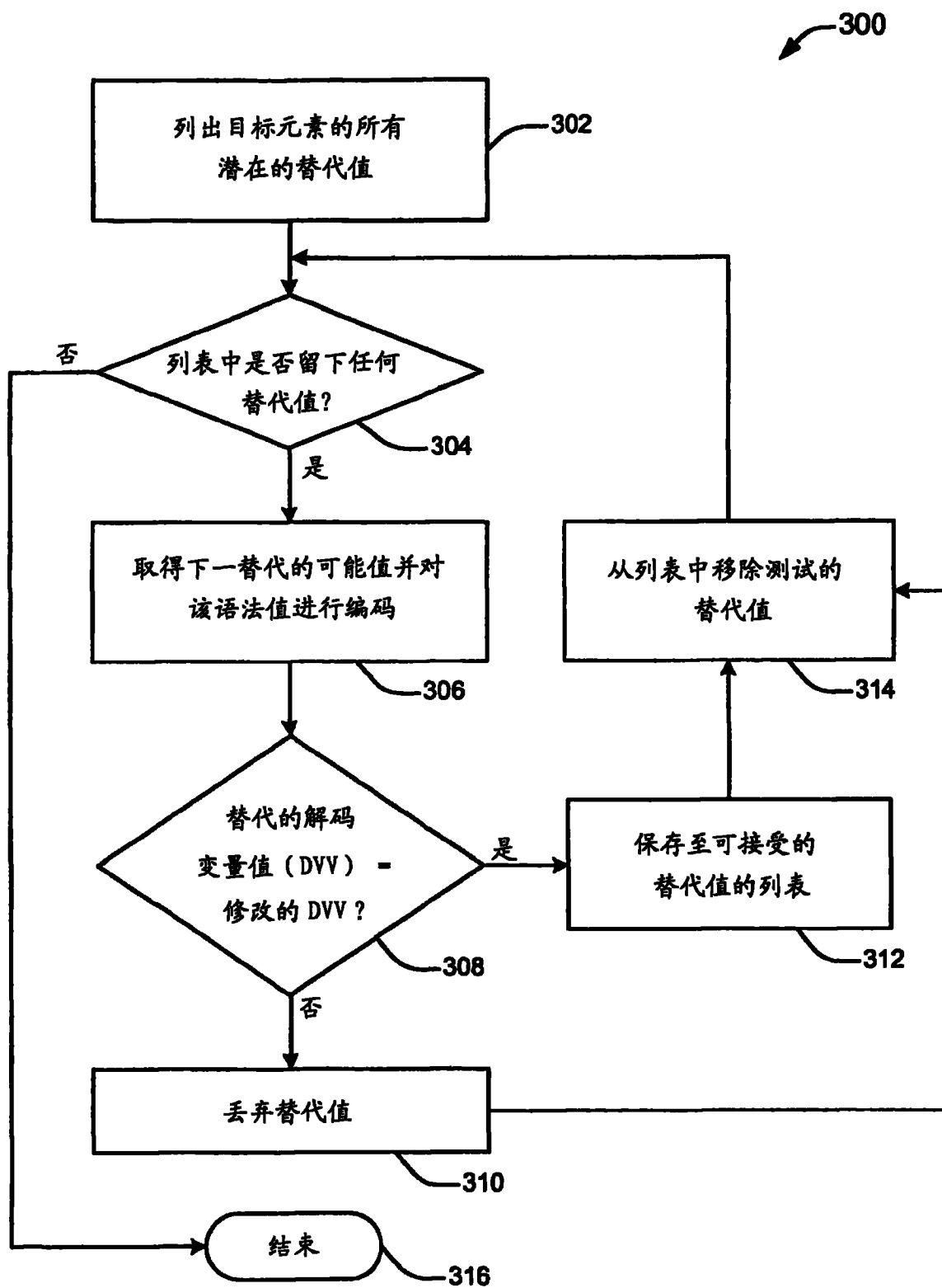


图 3

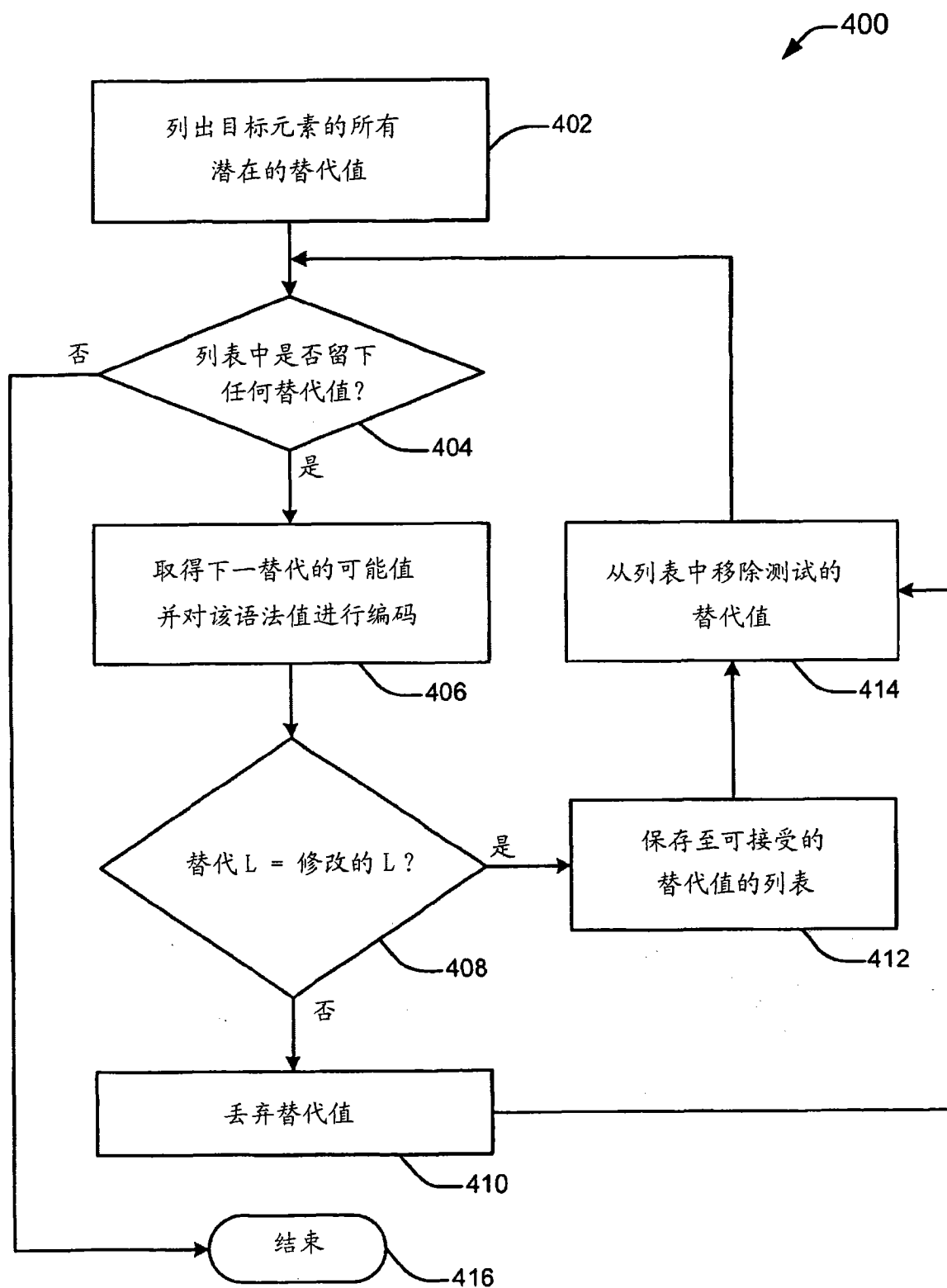


图 4

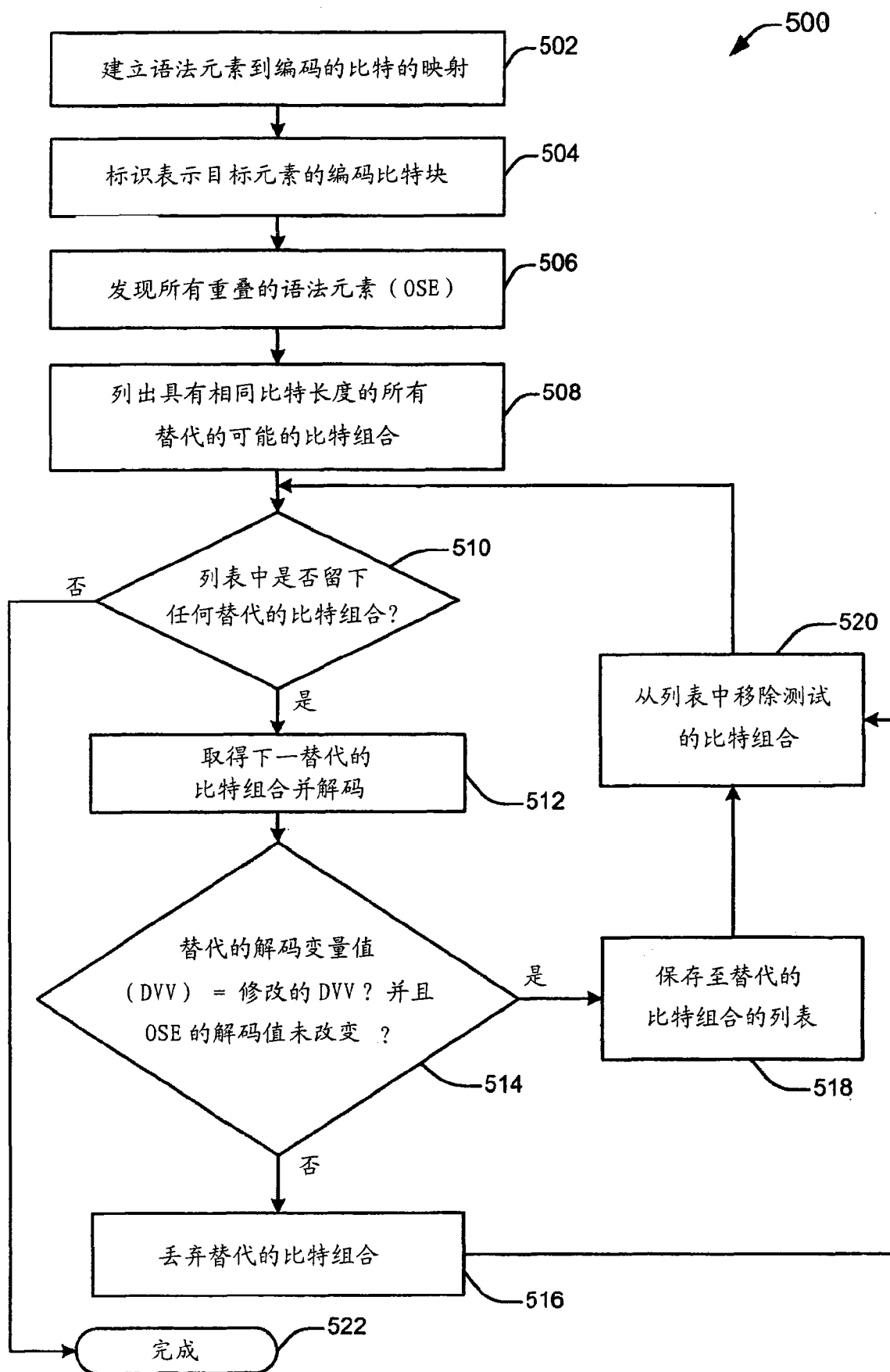


图 5

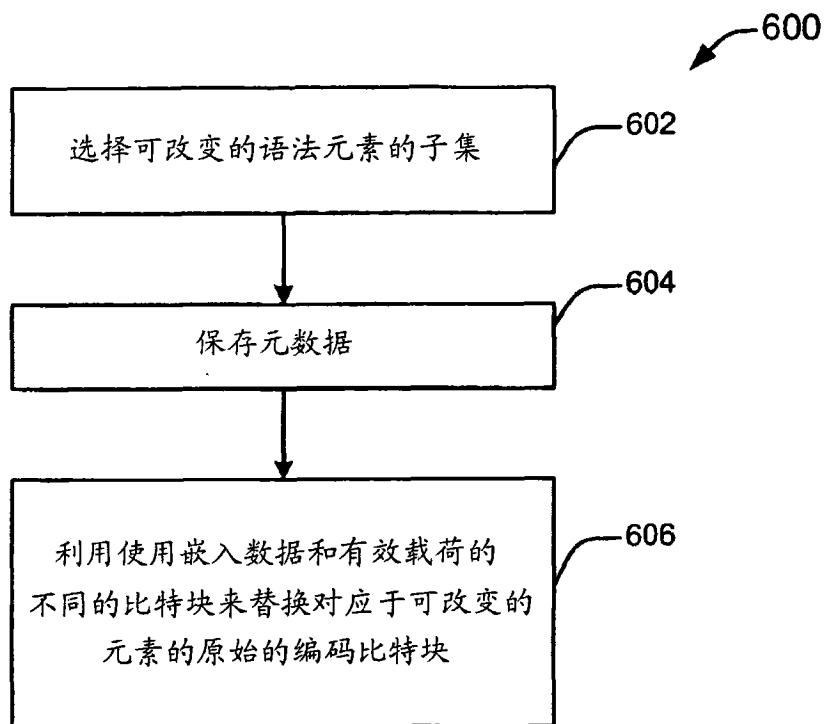


图 6

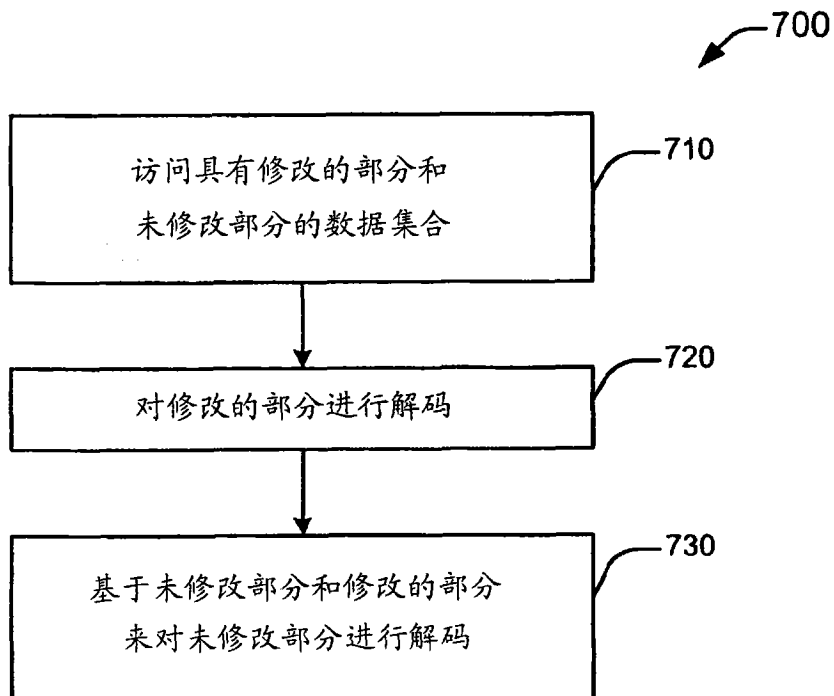


图 7

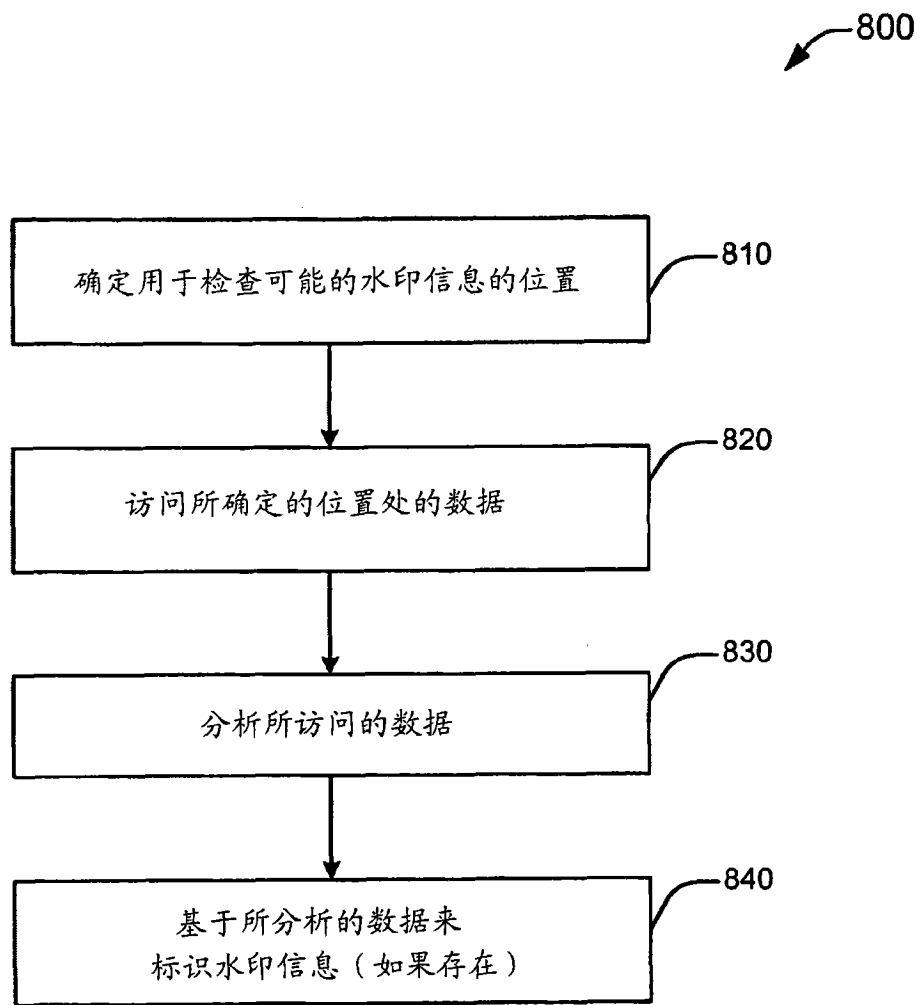


图 8

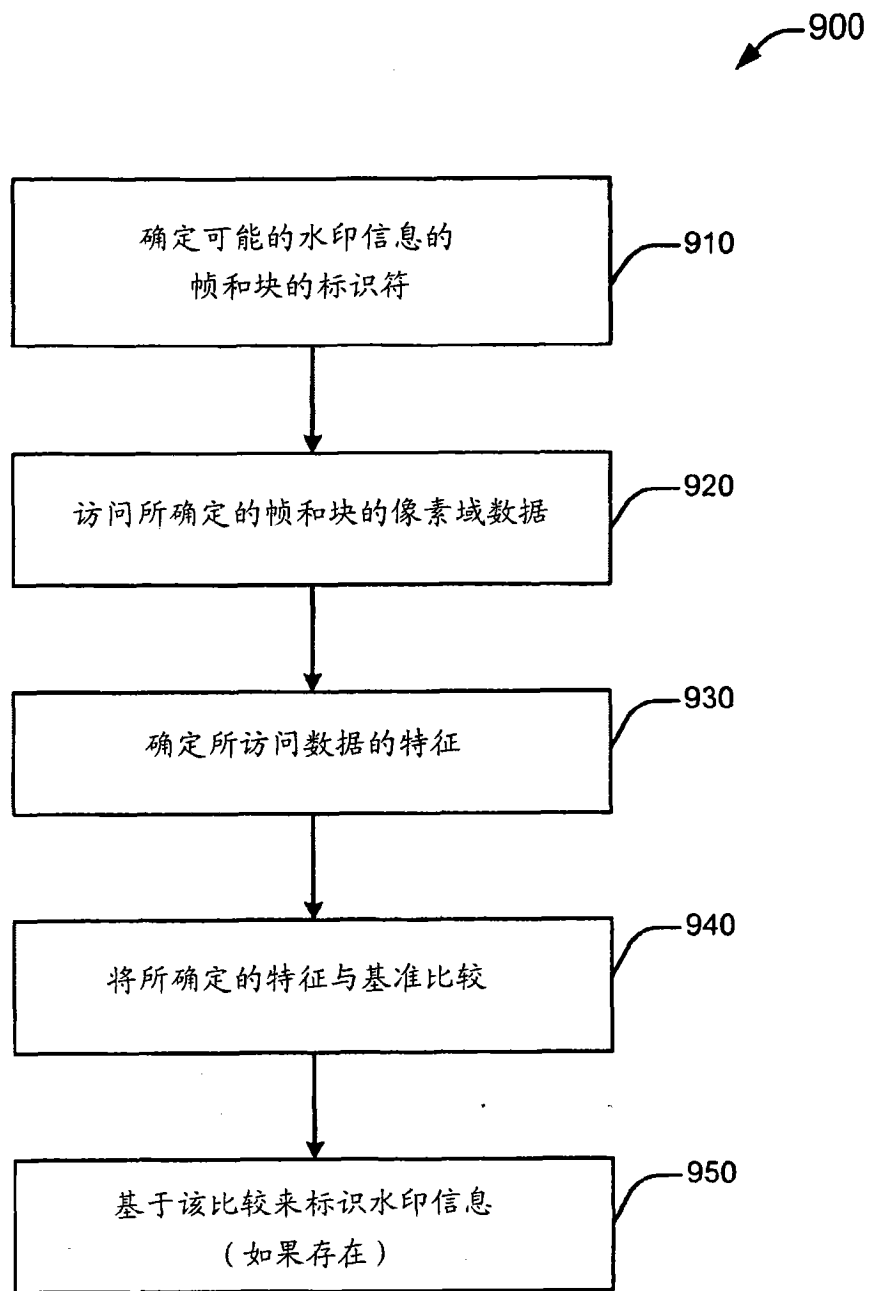


图 9