



(12)发明专利

(10)授权公告号 CN 106170921 B

(45)授权公告日 2019.10.15

(21)申请号 201580009593.0 图奥马斯·卡尔凯恩 (续)

(22)申请日 2015.02.20 (74)专利代理机构 北京英赛嘉华知识产权代理  
有限责任公司 11204  
代理人 王达佐 王艳春

(65)同一申请的已公布的文献号  
申请公布号 CN 106170921 A

(43)申请公布日 2016.11.30 (51)Int.Cl.  
H03M 7/30(2006.01)  
H03M 7/40(2006.01) (续)

(30)优先权数据  
1403039.9 2014.02.20 GB

(85)PCT国际申请进入国家阶段日  
2016.08.19 (56)对比文件  
US 2010195739 A1,2010.08.05,  
US 5838266 A,1998.11.17,  
CN 101808248 A,2010.08.18,  
JP H09284568 A,1997.10.31,  
US 6573847 B1,2003.06.03,  
CN 101848387 A,2010.09.29,

(86)PCT国际申请的申请数据  
PCT/EP2015/025008 2015.02.20 审查员 鲍旭恒

(87)PCT国际申请的公布数据  
W02015/124324 EN 2015.08.27

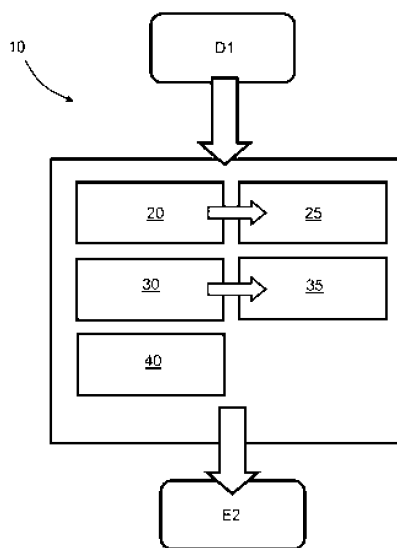
(73)专利权人 古鲁洛吉克微系统公司  
地址 芬兰图尔库

(72)发明人 奥西·卡尔沃 权利要求书7页 说明书24页 附图4页

(54)发明名称  
涉及符号压缩的数据的源编码和解码方法及装置

(57)摘要  
提供了一种在编码器中对输入数据(D1)进行编码以生成对应的编码数据(E2)的方法。该方法包括:(a)将输入数据(D1)拆分和/或变换为一个或多个数据块,并且对存在于输入数据(D1)中的符号进行分析,并且根据符号在一个或多个数据块中的出现对符号进行压缩;(b)为存在于一个或多个数据块中的符号生成一个或多个代码表和/或一个或多个频率表和/或一个或多个代码字长度表;(c)计算一个或多个索引集,该索引集将每个数据块中的符号和/或经压缩的符号关联至一个或多个代码表和/或一个或多个频率表和/或一个或多个代码字长度表中的条目;以及(d)将一个或多个索引集与一个或多个频率表和/或一个或多个代码表和/或指示所述一个或多个表的信息一同集合以用于生成编码数据

(E2)。还提供了使用该方法的编码器(50)以及对应的解码器(60),其中编码器(50)与解码器(60)共同形成编解码器(100)。



CN 106170921 B

[接上页]

(72)发明人 阿图尔·胡塔尼米

*H04N 19/91*(2014.01)

(51)Int.Cl.

*H04N 19/625*(2014.01)

*H03M 7/42*(2006.01)

1. 一种在编码器 (50) 中对输入数据 (D1) 进行编码以生成对应的编码数据 (E2) 的方法, 其特征在于, 所述方法包括:

(a) 对存在于所述输入数据 (D1) 中的符号进行分析, 并且将所述输入数据 (D1) 拆分和/或变换为一个或多个数据块;

(b) 为存在于所述一个或多个数据块中的所述符号, 根据所述符号的出现, 生成一个或多个代码表、一个或多个频率表、一个或多个代码字长度表、一个或多个概率表中的至少之一;

(c) 计算一个或多个索引集, 所述一个或多个索引集使每个数据块中的符号关联至所述一个或多个代码表、所述一个或多个频率表、所述一个或多个代码字长度表、所述一个或多个概率表中的至少之一中的条目;

(d) 将关联每个数据块中的所述符号的所述一个或多个索引集的信息与所述一个或多个频率表、所述一个或多个代码表、所述一个或多个代码字长度表、所述一个或多个概率表、指示一个或多个所述表的信息中的至少之一一同集合至所述编码数据 (E2); 以及

(e) 将一个或多个所述表内的所述符号压缩至所述编码数据 (E2)。

2. 如权利要求1所述的方法, 其特征在于, 一个或多个所述表中的至少一个是预先定义的。

3. 如权利要求1或2所述的方法, 其特征在于, 所述方法包括:

将包含关联每个数据块中的符号的所述一个或多个索引集的信息的所述编码数据 (E2), 与所述一个或多个频率表、所述一个或多个代码表、所述一个或多个代码字长度表、所述一个或多个概率表、所述指示一个或多个所述表的信息以及经压缩的符号中的至少之一一起传输。

4. 如权利要求1所述的方法, 其特征在于, 所述方法包括:

生成一个或多个所述表, 以用于解码期间根据其对称性和对应的对称指示性信息而重新生成一个或多个所述表。

5. 如权利要求1所述的方法, 其特征在于, 所述方法包括:

以一个或多个所述表中的至少一个能够被存储以用于后续再使用的方式, 传递一个或多个所述表中的至少一个。

6. 如权利要求1所述的方法, 其特征在于, 所述方法包括:

将一个或多个数据压缩算法应用在步骤 (d) 中以生成所述编码数据 (E2)。

7. 如权利要求6所述的方法, 其特征在于, 所述一个或多个数据压缩算法包括以下中的至少一种: 哈夫曼编码、VLC、熵编码、算术编码、区间编码。

8. 如权利要求1所述的方法, 其特征在于, 所述方法包括:

将所述输入数据 (D1) 拆分为多个数据块, 并且使用并行架构处理器以基本上并发的方式处理所述多个数据块。

9. 如权利要求1所述的方法, 其特征在于, 所述方法包括:

根据组合在一起的多个数据值生成所述一个或多个索引集。

10. 如权利要求9所述的方法, 其特征在于, 所述索引来源于包含R、G和B像素值的一个或多个RGB像素或包含Y、U和V像素值的一个或多个YUV像素。

11. 如权利要求9或10所述的方法, 其特征在于, 所述方法包括:

根据所述数据块被包含在所述编码数据 (E2) 时可实现的数据压缩比, 在将未编码的数据块集合到所述编码数据 (E2) 与将编码的数据块集合到所述编码数据 (E2) 之间动态地切换。

12. 如权利要求1所述的方法, 其特征在于, 所述方法包括:

将至少一个末尾比特合并到所述编码数据 (E2) 中, 以指示符号是否属于“代码表的更改”或“数据结束”。

13. 如权利要求1所述的方法, 其特征在于, 所述方法包括:

针对给定的数据块, 生成基本上仅足够用于指向存在于所述给定的数据块中的一个或多个符号所需的索引。

14. 如权利要求1所述的方法, 其特征在于, 所述方法还包括:

对所述一个或多个代码表进行压缩以包括在所述编码数据 (E2) 中。

15. 如权利要求14所述的方法, 其特征在于, 所述一个或多个代码表通过使用哈夫曼编码进行压缩。

16. 如权利要求14所述的方法, 其特征在于, 所述一个或多个代码表的压缩使用一个或多个附属代码表。

17. 如权利要求1所述的方法, 其特征在于, 所述方法包括:

以使得所述一个或多个代码表能够在解码器中用于对随后发送的数据进行解码的方式, 传输所述一个或多个代码表。

18. 如权利要求1所述的方法, 其特征在于, 所述方法包括:

将一个或多个识别代码包括在所述编码数据 (E2) 中, 所述一个或多个识别代码指示通过一个或多个数据库和/或一个或多个代理数据库易于从何处访问所述一个或多个代码表。

19. 如权利要求1所述的方法, 其特征在于, 所述方法布置成对以下类型的数据中的一个或多个进行编码: 捕获的音频信号、捕获的视频信号、捕获的图像、文本数据、地震数据、传感器信号数据、模拟数字 (ADC) 转换数据、生物信号数据、日历数据、经济数据、数学数据、二进制数据。

20. 一种用于对输入数据 (D1) 进行编码以生成对应的编码数据 (E2) 的编码器 (50), 其特征在于, 所述编码器 (50) 包括:

(a) 分析器, 用于对存在于所述输入数据 (D1) 中的符号进行分析并将所述输入数据 (D1) 拆分和/或变换为一个或多个数据块;

(b) 生成器, 用于为存在于所述一个或多个数据块中的所述符号, 根据所述符号的出现, 生成一个或多个代码表、一个或多个频率表、一个或多个代码字长度表、一个或多个概率表中的至少之一;

(c) 计算引擎, 用于计算一个或多个索引集, 所述一个或多个索引集使每个数据块中的符号关联至所述一个或多个代码表、所述一个或多个频率表、所述一个或多个代码字长度表、所述一个或多个概率表中的至少之一中的条目;

(d) 数据集合装置, 用于将关联每个数据块中的所述符号的所述一个或多个索引集与所述一个或多个频率表、所述一个或多个代码表、所述一个或多个代码字长度表、所述一个或多个概率表、指示这种一个或多个表的信息中的至少之一一同集合至所述编码数据

(E2); 以及

(e) 压缩器, 用于将一个或多个所述表内的所述符号压缩至所述编码数据 (E2)。

21. 如权利要求20所述的编码器 (50), 其特征在于, 所述编码器 (50) 可操作为:

将包含关联每个数据块中的符号的所述一个或多个索引集的信息的所述编码数据 (E2), 与所述一个或多个频率表、所述一个或多个代码表、所述一个或多个代码字长度表、所述一个或多个概率表、所述指示这种一个或多个表的信息以及经压缩的符号中的至少之一一起传输。

22. 如权利要求21所述的编码器 (50), 其特征在于, 所述编码器 (50) 可操作为以一个或多个所述表中的至少一个能够被存储以用于后续再使用的方式, 传递一个或多个所述表中的至少一个。

23. 如权利要求20所述的编码器 (50), 其特征在于, 所述编码器 (50) 可操作为将一个或多个数据压缩算法应用在所述数据集合装置中以生成所述编码数据 (E2)。

24. 如权利要求23所述的编码器 (50), 其特征在于, 所述一个或多个数据压缩算法包括以下中的至少一种: 哈夫曼编码、VLC、熵编码、算术编码、区间编码。

25. 如权利要求20所述的编码器 (50), 其特征在于, 所述编码器 (50) 可操作为将所述输入数据 (D1) 拆分为多个数据块, 并且使用并行架构处理器以基本上并发的方式处理多个数据块。

26. 如权利要求20所述的编码器 (50), 其特征在于, 所述生成器可操作为根据组合在一起的多个数据值生成所述一个或多个索引集。

27. 如权利要求26所述的编码器 (50), 其特征在于, 所述索引来源于包含R、G和B像素值的一个或多个RGB像素或包含Y、U和V像素值的一个或多个YUV像素。

28. 如权利要求27所述的编码器 (50), 其特征在于, 所述编码器 (50) 可操作为: 根据所述数据块被包含在所述编码数据 (E2) 时可实现的数据压缩比, 在将未编码的数据块集合到所述编码数据 (E2) 与将编码的数据块集合到所述编码数据 (E2) 之间动态地切换。

29. 如权利要求20所述的编码器 (50), 其特征在于, 所述编码器 (50) 可操作为将至少一个末尾比特合并至所述编码数据 (E2) 中, 以指示符号是否属于“代码表的更改”或“数据结束”。

30. 如权利要求20所述的编码器 (50), 其特征在于, 所述生成器可操作为针对给定的数据块, 生成基本上仅足够用于指向存在于所述给定的数据块中的一个或多个符号所需的索引。

31. 一种包括其上存储有计算机可读指令的非暂时性计算机可读存储介质的计算机程序产品, 所述计算机可读指令可由包含有用于实现如权利要求1至19中的任一项所述的方法的处理硬件的计算机化装置执行。

32. 一种在解码器 (60) 对通过如权利要求20所述的编码器 (50) 生成的编码数据 (E2) 进行解码以用于生成对应的解码数据 (D3) 的方法, 其特征在于, 所述方法包括:

(i) 接收所述编码数据 (E2), 并且从所述编码数据 (E2) 提取一个或多个索引集的信息, 以及一个或多个频率表、一个或多个代码表、一个或多个代码字长度表、一个或多个概率表、指示一个或多个所述表的信息中的至少之一;

(ii) 从所述一个或多个索引集, 计算所述一个或多个代码表、所述一个或多个频率表、

所述一个或多个代码字长度表、一个或多个概率表中的条目的经压缩的符号、一个或多个数据块中的至少之一中的对应符号；以及

(iii) 利用来自所述一个或多个代码表、一个或多个频率表、一个或多个代码字长度表、一个或多个概率表中的至少之一的信息，从所述符号重新生成一个或多个数据块；以及

(iv) 对所述一个或多个数据块进行组合和/或变换以生成所述解码数据 (D3)。

33. 如权利要求32所述的方法，其特征在于，一个或多个所述表中的至少一个是预先定义的。

34. 如权利要求32所述的方法，其特征在于，所述方法包括：

对所述解码数据 (D3) 进行转码以生成对应的转码数据 (D4)、和/或从所述编码数据 (E2) 生成所述对应的转码数据 (D4)。

35. 如权利要求32所述的方法，其特征在于，所述方法包括：

根据一个或多个表的对称性和对应的对称指示性信息，重新生成所述一个或多个表。

36. 如权利要求32所述的方法，其特征在于，所述方法包括：

以一个或多个所述表中的至少一个能够被存储以用于后续再使用的方式，接收一个或多个所述表中的至少一个。

37. 如权利要求32所述的方法，其特征在于，所述方法包括：

将一个或多个数据解压算法应用在步骤 (iv) 中以生成所述解码数据 (D3)。

38. 如权利要求37所述的方法，其特征在于，所述一个或多个数据解压算法包括以下中的至少一种：哈夫曼解码、VLC解码、熵解码、算术解码、区间解码。

39. 如权利要求32所述的方法，其特征在于，所述方法包括：

通过使用用于以基本上并发的方式处理多个数据块的并行架构处理器，将所述一个或多个数据块中的多个组合以生成所述解码数据 (D3)。

40. 如权利要求32所述的方法，其特征在于，所述方法包括根据组合在一起的多个数据值生成所述一个或多个索引集。

41. 如权利要求40所述的方法，其特征在于，所述索引来源于包含R、G和B像素值的一个或多个RGB像素或包含Y、U和V像素值的一个或多个YUV像素。

42. 如权利要求40所述的方法，其特征在于，所述方法包括：

根据所述数据块被包含在所述编码数据 (E2) 时可实现的数据压缩比，在将未编码的数据块集合到所述编码数据 (E2) 与将编码的数据块集合到所述编码数据 (E2) 之间动态地切换。

43. 如权利要求32所述的方法，其特征在于，所述解码器 (50) 可操作为：从所述编码数据 (E2) 提取指示符号是否属于“代码表的更改”或“数据结束”的至少一个末尾比特。

44. 如权利要求32所述的方法，其特征在于，所述方法包括：

通过基本上仅足够用于指向存在于给定的数据块中的一个或多个符号所需的索引，生成所述给定的数据块。

45. 如权利要求32所述的方法，其特征在于，所述方法包括：

对包含在所述编码数据 (E2) 中的所述一个或多个代码表进行解压缩。

46. 如权利要求45所述的方法，其特征在于，所述方法包括：

通过使用哈夫曼解码对所述一个或多个代码表进行解压缩。

47. 如权利要求45所述的方法,其特征在于,所述一个或多个代码表的解压缩使用一个或多个附属代码表。

48. 如权利要求32所述的方法,其特征在于,所述方法包括:

以使得所述一个或多个代码表能够在所述解码器(60)中用于对随后发送的数据进行解码的方式,接收所述一个或多个代码表。

49. 如权利要求32所述的方法,其特征在于,所述方法包括:

从所述编码数据(E2)中提取一个或多个识别代码,所述一个或多个识别代码指示通过一个或多个数据库和/或一个或多个代理数据库易于从何处访问所述一个或多个代码表。

50. 如权利要求32所述的方法,其特征在于,所述方法包括:

对以下类型的数据中的一个或多个进行解码:捕获的音频信号、捕获的视频信号、捕获的图像、文本数据、地震数据、传感器信号数据、模拟数字(ADC)转换数据、生物信号数据、日历数据、经济数据、数学数据、二进制数据。

51. 如权利要求32所述的方法,其特征在于,所述方法包括:

从多个数据源接收所述编码数据(E2),并且将由所述源提供的数据组合以重新生成所述编码数据(E2)。

52. 一种对通过如权利要求20所述的编码器(50)生成的编码数据(E2)进行解码以用于生成对应的解码数据(D3)的解码器(60),其特征在于,所述解码器(60)可操作为:

(i) 接收所述编码数据(E2),并且从所述编码数据(E2)提取一个或多个索引集的信息,以及一个或多个频率表、一个或多个代码表、一个或多个代码字长度表、一个或多个概率表、指示这些一个或多个表的信息中的至少之一;

(ii) 从所述一个或多个索引集,计算所述一个或多个代码表、所述一个或多个频率表、所述一个或多个代码字长度表、一个或多个概率表中的条目的经压缩的符号、一个或多个数据块中的至少之一中的对应符号;以及

(iii) 利用来自所述一个或多个代码表、一个或多个频率表、一个或多个代码字长度表、一个或多个概率表中的至少之一的信息,从所述符号重新生成一个或多个数据块;以及

(iv) 对所述一个或多个数据块进行组合和/或变换以生成所述解码数据(D3)。

53. 如权利要求52所述的解码器(60),其特征在于,所述解码器(60)还包括:

转码器(70),用于对所述解码数据(D3)进行转码以生成对应的转码数据(D4)和/或从所述编码数据(E2)生成所述对应的转码数据(D4)。

54. 如权利要求52或53所述的解码器(60),其特征在于,所述解码器(60)可操作为:根据一个或多个表的对称性和提供至所述解码器(60)的对应的对称指示性信息,重新生成所述一个或多个表。

55. 如权利要求52所述的解码器(60),其特征在于,所述解码器(60)可操作为:以一个或多个所述表中的至少一个能够被存储以用于后续再使用的方式,接收一个或多个所述表中的至少一个。

56. 如权利要求52所述的解码器(60),其特征在于,所述解码器(60)可操作为将一个或多个数据解压算法应用在步骤(iv)中以生成所述解码数据(D3)。

57. 如权利要求56所述的解码器(60),其特征在于,所述一个或多个数据解压算法包括以下中的至少一种:哈夫曼解码、VLC解码、熵解码、算术解码、区间解码。

58. 如权利要求52所述的解码器(60),其特征在于,所述解码器(60)可操作为:通过使用用于基本上并发的方式处理多个数据块的并行架构处理器,将所述一个或多个数据块中的多个组合以生成所述解码数据(D3)。

59. 如权利要求52所述的解码器(60),其特征在于,所述解码器(60)可操作为根据组合在一起的多个数据值生成所述一个或多个索引集。

60. 如权利要求59所述的解码器(60),其特征在于,所述索引来源于包含R、G和B像素值的一个或多个RGB像素或包含Y、U和V像素值的一个或多个YUV像素。

61. 如权利要求59或60所述的解码器(60),其特征在于,所述解码器(60)可操作为:根据所述数据块被包含在所述编码数据(E2)时可实现的数据压缩比,在将未编码的数据块集合到所述编码数据(E2)与将编码的数据块集合到所述编码数据(E2)之间动态地切换。

62. 如权利要求52所述的解码器(60),其特征在于,所述解码器(60)可操作为从所述编码数据(E2)中提取指示符号是否属于“代码表的更改”或“数据结束”的至少一个末尾比特。

63. 如权利要求52所述的解码器(60),其特征在于,所述解码器(60)可操作为:通过基本上仅足够用于指向存在于给定的数据块中的一个或多个符号所需的索引,生成所述给定的数据块。

64. 如权利要求52所述的解码器(60),其特征在于,所述解码器(60)可操作为对包含在所述编码数据(E2)中的所述一个或多个代码表进行解压缩。

65. 如权利要求64所述的解码器(60),其特征在于,所述解码器(60)可操作为通过使用哈夫曼解码对所述一个或多个代码表进行解压缩。

66. 如权利要求64所述的解码器(60),其特征在于,所述一个或多个代码表的解压缩使用一个或多个附属代码表。

67. 如权利要求52所述的解码器(60),其特征在于,所述解码器(60)可操作为:以使得所述一个或多个代码表能够在所述解码器(60)中用于对随后发送的数据进行解码的方式,接收所述一个或多个代码表。

68. 如权利要求52所述的解码器(60),其特征在于,所述解码器(60)可操作为:从所述编码数据(E2)中提取一个或多个识别代码,所述一个或多个识别代码指示通过一个或多个数据库和/或一个或多个代理数据库易于从何处访问所述一个或多个代码表。

69. 如权利要求52所述的解码器(60),其特征在于,所述解码器(60)可操作为对以下类型的数据中的一个或多个进行解码:捕获的音频信号、捕获的视频信号、捕获的图像、文本数据、地震数据、传感器信号数据、模拟数字(ADC)转换数据、生物信号数据、日历数据、经济数据、数学数据、二进制数据。

70. 如权利要求52所述的解码器(60),其特征在于,所述解码器(60)可操作为从多个数据源接收所述编码数据(E2),并且将由所述源提供的数据进行组合以重新生成所述编码数据(E2)。

71. 一种编解码器(100),其特征在于,所述编解码器(100)包括至少一个如权利要求20所述的用于对输入数据(D1)进行编码以生成对应的编码数据(E2)的编码器(50),以及至少一个如权利要求52所述的用于对所述编码数据(E2)进行解码以生成解码输出数据(D3)的解码器(60)。

72. 如权利要求71所述的编解码器(100),其特征在于,至少一个所述编码器(50)和至



少一个所述解码器(60)在空间上彼此远离,并且通过数据通信网络彼此连接在一起。

73.如权利要求71所述的编解码器(100),其特征在于,所述数据通信网络以点对点通信网络的方式配置。

74.如权利要求71所述的编解码器(100),其特征在于,所述编码器(50)和所述解码器(60)对数据进行处理的过程是对称的。

## 涉及符号压缩的数据的源编码和解码方法及装置

### 技术领域

[0001] 本公开涉及对输入数据进行编码以生成对应的编码数据的方法。另外，本公开还涉及对上述编码数据进行解码以生成对应的解码输出数据的方法。此外，本公开还涉及可操作为实现上述方法的编码器和解码器。附加地，本公开涉及包括其上存储有计算机可读指令的非暂时性计算机可读存储介质的计算机程序产品，其中计算机可读指令可由包括用于执行上述方法的处理硬件的计算机化装置执行。

### 背景技术

[0002] 大体上，如图1所示，对输入数据D1进行编码以生成对应的编码输出数据E2的已知编码方法涉及将一个或多个变换T应用到输入数据D1，以生成对应的经变换的编码输出数据E2，其中经变换的编码输出数据E2具有与其相关联的编码表数据C信息，该信息指示定义所使用的一个或多个变换T的一个或多个编码表。经变换的编码数据E2和编码表数据C信息（统称为编码输出数据E2）常通过数据载体和/或数据通信网络传输至一个或多个解码器，该一个或多个解码器可操作为应用一个或多个逆变换 $T^{-1}$ 以对编码输出数据E2进行解码，从而生成对应的解码数据D3。通常希望相对于输入数据D1压缩编码输出数据E2，例如以减少传输编码输出数据E2时的通信网络容量负荷。另外，还希望以基本上无损的方式对编码输出数据E2进行压缩，使得解码数据D3是包含在输入数据D1中的信息的准确复制。当编码表数据C信息相对于经变换的编码数据E2具有明显大的尺寸时，即当编码表数据C信息对应于经变换的编码数据E2中的大量数据开销时，编码输出数据E2关于输入数据D1可实现的数据压缩会缺乏效率。

[0003] 存在若干已知的对输入数据D1进行编码以生成编码输出数据E2的方法。例如，常使用已知的哈夫曼编码或其他VLC编码方法来压缩各种类型的数据。另外，使用算术编码或区间编码来压缩输入数据也变得越来越普遍，但是在如下的情形中甚是缺乏效率：

[0004] (i) 输入数据D1的频率表对可操作为对输入数据D1进行编码以生成对应的编码输出数据E2的编码器以及可操作为对编码输出数据E2进行解码的解码器并非已知的；以及

[0005] (ii) 输入数据的量相对小，例如在输入数据D1以小数据段或数据块的形式传输的情况下，其中每个数据段或数据块附有对应的频率表。

[0006] 如上所述，如果不能通过相对少的识别参数从可能的频率表列表（例如解码器在其本地所存储的）中进行选择，则由于传递一个或多个频率表消耗相当大的数据空间而导致产生上述低效率。另外，相比于合适的代码表，从这种列表中同样更不可能找到合适的频率表。通常，待编码的输入数据D1在本地还可能是经常变化的，例如在通过通信网络进行传输时其经过变换以符合通信网络的空间本地数据标准。

[0007] 存在可用于在传输源自符号的编码数据内容的同时传递代码表或频率表的已知方法。大多数已知方法采用直接传递符号的哈夫曼树或频率的方法。由于这种已知方法需要将大量信息从编码器传递至对应的解码器，因此其并不令人十分满意。另外，也存在用于传递代码表符号的长度的已知方法，例如，在已被弃用的已知的IPP库中所采用的方法；其

中使用通过“HuffLenCodeTablePack”压缩代码表并且通过“HuffLenCodeTableUnpack”重新将其解压缩的方法；然而，该方法不是很令人满意甚至有时在编码过程中增加数据尺寸。此外，该方法还需要256个符号，并且所有0至255的符号具有非零的代码字长度。传递代码表的方法是当前可用于例如通过哈夫曼编码技术生成的像素代码的方法中仍然是最有效的传递机制。当哈夫曼树从编码器传递至对应的解码器时，从编码器生成的代码符号通常在编码器和解码器中是相似的。当仅传递频率表时，则需要在编码器和解码器中使用相似的算法，以在需要哈夫曼树的情况下从频率表生成真正的哈夫曼树，使得在解码器中能够以恰当的方式对符号进行解码。如果传递代码表符号的长度，则在编码器和解码器中还需要用于从符号长度至频率表的变换的类似方法，从而能够以恰当的方式对符号进行解码。对于算术编码和区间编码而言，将符号长度从编码器传输至解码器不是传递频率的实用方法，因为他们设计为支持更精确的频率表而不仅用于传输代码符号的长度。代码符号的长度也可用于算术编码和区间编码。然而，如果随后不对表进行针对后续数据的适应性更新，则这些方法例如相比于哈夫曼编码并不具有优势。传递表示概率的信息通常在区间编码或算术编码中相比于哈夫曼编码产生更优的编码结果。符号的概率可通过将符号出现频率除以符号的出现频率之和（即等于符号的数量）来计算。这种概率的传递有益地通过经缩放的概率值来实现。经缩放的概率值可通过将原始符号概率值乘以有利地为2值的幂（即 $2^n$ ，其中n为整数）的整数，并且将其四舍五入为最接近的整数值来计算。这些为整数的经缩放的概率之和等同为与乘数值相同。另外，有益地为原本未分配其自身非零的经缩放概率值的符号创建转义代码符号。这意味着需要转义代码的那些符号具有比能够通过所选的乘数值更小的概率。也可能在不使用通过两个不同机制的转义代码的情况下创建经缩放的概率。可增加乘数值，然后可计算出新的概率值。还可将用于可用符号的经缩放的概率值从等于零提升为等于一。该概率值的提升需要通过降低其他符号的概率值来补偿概率值的增加。这么做是为了使概率之和与乘数值完全相同。这个过程未能使概率值尽可能得优化，但是不需要转义符号，并且在某些情况下，这可能仍是最优编码方法。符号的长度或概率值定义可用于采用可变长度编码符号方法的频率表的粗略估计，例如哈夫曼编码、区间编码、算术编码以及任何其他可变长度编码方法。应理解，在需要时，经缩放的概率表可直接用作符号频率的粗略估计，并且在使用前首先需要将符号长度转换为符号频率的粗略估计。稍后将在描述数据编码和表传递时说明符号长度到频率表的转换。

[0008] 很多已知的对数据进行编码的实用方法完全不使用优化的代码表，即他们使用固定的代码表来对数据进行编码以生成对应的编码数据，并且随后使用固定的代码表对编码数据进行解码。有时，通过基于所传递符号的适应性方法来更新表。在某些已知的方法中，有时在编码器中使用若干不同的代码表（可替换的频率表）来对数据进行编码，并且相应地在解码器中对编码数据进行解码，其中将定义所选择的代码表、概率表或频率表的索引作为信息从编码器传递至解码器。在某些方法中，对于亮度和颜色通道、块间和块内或者对不同类型的数据使用不同的表，但是，不同的表以低效的方式被传输，例如，本文中引用以下互联网网站（维基网）：[http://en.wikipedia.org/wiki/Huffman\\_coding](http://en.wikipedia.org/wiki/Huffman_coding)。在解压期间，使用基于哈夫曼的方法，必须重建哈夫曼树。在字符频率相对可预测的最简单情形中，易于重建树，甚至可在每个压缩循环中进行统计调整，从而每次以至某些程度的压缩效率为代价地重复使用；可替代地，必须预先（即提前）发送哈夫曼树信息。

[0009] 预先准备与编码为压缩数据输出流的符号有关的频率计数的简单方法的主要缺点在于：其实际上使压缩数据中的数据量至少增加好几千字节 (KB)，因此这种简单方法具有极小的实际用途。如果通过标准编码压缩数据，则可仅利用  $B2^B$  比特的信息来精确地重建压缩模型，其中  $B$  为每个符号的比特数，例如，对于 8 比特需要 2kB。

[0010] 另一方法简单地通过逐个比特的方式预先准备哈夫曼树以得到压缩的输出流。例如，假设数值 0 表示母节点并且 1 表示叶节点，每当碰到后者，建树程序则读取下一个 8 比特以确定上述叶节点的字符值。递归地持续进行该过程直到到达最后一个叶节点，此时，例如在解码器中将准确地重建出哈夫曼树。假设一个字母为 8 比特，则使用该方法产生的数据开销大约为 2 字节至 320 字节。

[0011] 为了解释其他用于对数据进行编码以及相应的对编码数据进行解码的已知方法，下面将大致描述哈夫曼解码。将理解，除了哈夫曼解码之外，也可使用诸如区间解码或算术解码的任何其他方法。在开始对数据文件进行压缩之前，编码器中的压缩器需要确定在执行压缩时所使用的代码。

[0012] 当使用哈夫曼解码时，在开始对包含有符号的给定数据文件进行压缩以生成对应的编码输出数据之前，编码器需要确定待用于表示给定数据的代码。合宜地，代码基于给定数据文件中符号的概率，即出现频率。然而，需要在编码输出数据中将频率、概率或符号长度记录例如为边信息，即补充信息，使得任意哈夫曼解码器能够对编码输出数据进行解码以生成对应的解码数据。合宜地，出现频率或符号长度为整数，或者是可表示为比例整数的概率；这种包含在补充信息中的整数通常在编码输出数据上仅添加几百字节。可选地，也可将可变长度代码本身写入编码输出数据，但是由于代码可具有彼此不同的尺寸，因此这在某些情况下可能不适合。可替代地，将哈夫曼树写入编码输出数据是可行的，但是相比于仅传输给定数据中符号的出现频率，这需要传输更多的数据。

[0013] 在操作期间，必须将与解码器所接收的编码压缩文件的开头相关的信息提供至解码器以对其进行解码。从由编码压缩文件提取的数据开始，例如从该数据的起始位置开始，解码器可操作为构建哈夫曼树字母表。在解码器中构建哈夫曼树之后，解码器则能够使用哈夫曼树作为解码工具对文件的剩余部分进行解码。解码器使用相对简单的解码算法，该算法包括以下步骤：

[0014] (a) 从哈夫曼树的根开始，然后利用哈夫曼树读取待解码的编码输出数据的第一个比特；

[0015] (b) 如果第一个比特为“1”，则朝着哈夫曼树的顶端移动；如果第一个比特为“0”，则朝着哈夫曼树的底端移动；

[0016] (c) 读取编码输出数据的第二个比特，然后对第二个比特使用与步骤 (b) 类似的方法朝着哈夫曼树的“叶子”移动，直到最终到达“叶子”，在此处将找到未压缩的符号，通常为相关的 ASCII 代码；然后从解码器输出该代码；以及

[0017] (d) 重复步骤 (b) 和 (c) 直到编码输出数据被解码。

[0018] 当前已知的哈夫曼编码有益于在编码字符串相对于生成字符串所使用的代码表具有较大尺寸时使用。另外，这种当前的哈夫曼编码有益于在编码器和解码器中均预先定义了代码表的情况下使用。因此，需要替代的编码方法，该编码方法解决与对数据进行编码和解码的已知方法（诸如上述哈夫曼编码和解码方法）相关的局限性。

## 发明内容

[0019] 本发明旨在提供对数据 (D1) 进行编码以生成对应的编码数据 (E2) 的改进的方法。

[0020] 本发明还旨在提供可操作为使用上述对数据进行编码的改进方法的改进的编码器。

[0021] 本发明旨在提供对编码数据 (E2) 进行解码以生成对应的解码数据 (D3) 的改进方法。

[0022] 本发明旨在提供用于对上述编码数据 (E2) 进行解码以生成对应的解码数据 (D3) 的改进解码器。

[0023] 根据第一方面,提供了在编码器中对输入数据 (D1) 进行编码以生成对应的编码数据 (E2) 的方法,其特征在于,该方法包括:

[0024] (a) 对存在于输入数据 (D1) 中的符号进行分析,并将输入数据 (D1) 拆分和/或变换为一个或多个数据块;

[0025] (b) 为存在于一个或多个数据块中的符号,根据所述符号的出现,生成一个或多个代码表、一个或多个频率表、一个或多个代码字长度表、一个或多个概率表中的至少之一;

[0026] (c) 计算一个或多个索引集,一个或多个索引集使每个数据块中的符号关联至一个或多个代码表、一个或多个频率表、一个或多个代码字长度表、一个或多个概率表中的至少之一中的条目;

[0027] (d) 将关联每个数据块中的符号的一个或多个索引集的信息与一个或多个频率表、一个或多个代码表、一个或多个代码字长度表、一个或多个概率表、指示这种一个或多个表的信息中的至少之一一同集合至生成编码数据 (E2); 以及

[0028] (e) 将一个或多个表内的符号压缩至所述编码数据 (E2)。

[0029] 本发明的优点在于该方法涉及将输入数据 (D1) 拆分即划分为一个或多个数据块,和/或将符号压缩至输入数据 (D1) 中,使得能够对输入数据 (D1) 例如以对于每个数据块或经压缩的符号最合适的方式、利用索引(“指示(indexes)”)和通过索引引用的相关联的一个或多个表而有效地进行编码。

[0030] 可选地,在该方法中,一个或多个表中的至少一个是预先定义的。

[0031] 可选地,该方法包括:将包含关联每个数据块中的符号的一个或多个索引集的信息的编码数据 (E2),与一个或多个频率表、一个或多个代码表、一个或多个代码字长度表、一个或多个概率表、指示这种一个或多个表的信息以及经压缩的符号中的至少之一一起传输。

[0032] 可选地,该方法包括:生成一个或多个表,以用于解码期间根据其对称性和对应的对称指示性信息而重新生成一个或多个表。

[0033] 可选地,该方法包括以使得一个或多个表中的至少一个可存储为后续再使用的方式,传递一个或多个表中的至少一个。

[0034] 上述拆分即划分可选地包括将输入数据 (D1) 进行分割。

[0035] 可选地,在步骤 (a) 中,通常进行拆分,但是有时也有必要利用最优代码表对可用的数据块进行压缩,而不将数据拆分为新的数据块。另外,有时不对原始数据进行拆分,而是相反地,例如通过将一个或多个变换应用到需要以最有效的方式被压缩的一个或多个数据块,来创建新的数据。可使用根据本公开实现的编码器来创建不同的数据块。因此,将理

解,该方法尤其适用于可操作为通过来自不同时隙的数据块对数据进行编码的视频编解码器和音频编解码器。不同帧或区段是不同的数据块,并且这些帧和区段仍可以通过使用实现根据本公开的一个或多个方法的编码器被拆分为一个或多个数据块。所有这些数据块可以再利用之前所传递的任何表,例如在同一帧中或先前的帧中传递的表。

[0036] 本发明的实施方式实现了代码表或频率表的有效传递。这减少了表的传递和/或存储所需要的数据传输和/或数据存储开销。本发明的实施方式还通过使用更好地对每个单独的数据块进行优化的编码表,使得能够对较小的数据块进行编码。由此,能够显著地提升压缩效率,这意味着能够减小数据存储容量、传输带宽以及能量损耗。

[0037] 数据的各部分的频率通常是彼此不同的,并且通常其相关的数据熵也彼此不同,因此有益地将数据拆分为多个部分,即数据块。有益地,根据各部分的数据性质和/或数据类型和/或数据内容,对各部分使用不同的代码表;“性质”指的是数据的一个或多个特性和/或参数。本发明提供了使给定的大型数据文件能够更有效地拆分为较小部分即数据块的方法,其相关的益处在于能够针对这种数据块优化代码表或频率表的传递。这种大型数据文件的拆分对于修改所涉及数据的熵也具有实质性的益处,因此其能够大幅地减少待传输的编码数据的量。如上所述,也能够拆分一个或多个数据块中的数据值。例如通过将MSB(最高有效位)和LSB(最低有效位)彼此分开可实现数据值的拆分,即分裂。数据值也可选择性地拆分为多于两个的分开的数据值块。

[0038] 可选地,该方法包括将一个或多个数据压缩算法应用在步骤(d)中以生成编码数据(E2)。更可选地,在该方法中,一个或多个数据压缩算法包括以下中的至少一种:哈夫曼编码、VLC、熵编码、算术编码、区间编码,但是并不限于此。

[0039] 可选地,该方法包括将输入数据(D1)拆分为多个数据块,并且使用用于以基本上并发的方式处理多个数据块的并行架构处理器。

[0040] 可选地,该方法包括根据组合在一起的多个数据值,生成一个或多个索引集。更可选地,在该方法中,索引(即“指示(indexes)”)来源于包含R、G和B像素值或Y、U和V像素值的一个或多个RGB像素或YUV像素。

[0041] 更可选地,该方法包括根据数据块被包含在编码数据(E2)时可实现的数据压缩比,在将未编码的数据块集合到编码数据(E2)与将编码的数据块集合到编码数据(E2)之间动态地切换。

[0042] 可选地,该方法包括将至少一个末尾比特合并到编码数据(E2)中,以指示符号是否属于“代码表的更改”或“数据结束”。

[0043] 可选地,该方法包括针对给定的数据块,生成基本上仅足够用于指向存在于给定的数据块中的一个或多个符号所需的索引。

[0044] 可选地,还通过例如使用哈夫曼编码对所传递的代码表进行压缩,并且这种代码表的压缩方法还可选地提供其自身相关联的一个或多个代码表。更可选地,在该方法中,一个或多个代码表的压缩使用一个或多个附属代码表。

[0045] 可选地,该方法包括:以使得一个或多个代码表能够在解码器中用于对随后发送的数据进行解码的方式,传输一个或多个代码表。

[0046] 可选地,该方法包括:将一个或多个识别代码包括在编码数据(E2)中,一个或多个识别代码指示一个或多个代码表易于通过一个或多个数据库和/或一个或多个代理数据库

从何处访问。

[0047] 可选地,该方法布置成对以下类型的数据中的一个或多个进行编码:捕获的音频信号、捕获的视频信号、捕获的图像、文本数据、地震数据、传感器信号数据、模拟数字(ADC)转换数据、生物信号数据、日历数据、经济数据、数学数据、二进制数据。

[0048] 根据第二方面,提供了用于对输入数据(D1)进行编码以生成对应的编码数据(E2)的编码器,其特征在于,该编码器包括:

[0049] (a) 分析器,用于对存在于所述输入数据(D1)中的符号进行分析并将输入数据(D1)拆分和/或变换为一个或多个数据块;

[0050] b) 生成器,用于为存在于一个或多个数据块中的符号,根据所述符号的出现,生成一个或多个代码表、一个或多个频率表、一个或多个代码字长度表、一个或多个概率表;

[0051] (c) 计算引擎,用于计算一个或多个索引(即“指示(indexes)”)集,一个或多个索引集使每个数据块中的符号关联至一个或多个代码表、一个或多个频率表、一个或多个代码字长度表、一个或多个概率表中的至少之一中的条目;

[0052] (d) 数据集合装置,用于将关联每个数据块中的所述符号的一个或多个索引集与一个或多个频率表、一个或多个代码表、一个或多个代码字长度表、一个或多个概率表、指示这种一个或多个表的信息中的至少之一一同集合至编码数据(E2);以及(e) 压缩器,用于将一个或多个所述表内的所述符号压缩至所述编码数据(E2)。

[0053] 这种拆分即划分可选地包括输入数据D1的分割。

[0054] 可选地,在步骤(a)中,通常进行拆分,但是有时也有必要利用最优代码表对可用的数据块进行压缩,而不将数据拆分为新的数据块。另外,有时不对原始数据进行拆分,而是相反地,例如通过将一个或多个变换应用到需要以最有效的方式被压缩的一个或多个数据块,来创建新的数据。可使用根据本公开实现的编码器来创建不同的数据块。因此,将理解,该方法尤其适用于可操作为通过来自不同时段的数据块对数据进行编码的视频编解码器和音频编解码器。不同帧或区段是不同的数据块,并且这些帧和区段仍可以通过使用实现根据本公开的一个或多个方法的编码器被拆分为一个或多个数据块。所有这些数据块可以再利用之前所传递的任何表,例如在同一帧中或先前的帧中传递的表。

[0055] 可选地,编码器可操作为:将包含关联每个数据块中的符号的一个或多个索引集的信息的编码数据(E2),与一个或多个频率表、一个或多个代码表、一个或多个代码字长度表、一个或多个概率表、指示这种一个或多个表的信息以及经压缩的符号中的至少之一一起传输。

[0056] 可选地,编码器可操作为以一个或多个表中的至少一个能够被存储以用于后续再使用的方式,传递一个或多个表中的至少一个。

[0057] 可选地,编码器可操作为将一个或多个数据压缩算法应用在数据集合装置中以生成编码数据(E2)。更可选地,在该编码器中,一个或多个数据压缩算法包括以下中的至少一种:哈夫曼编码、VLC、熵编码、算术编码、区间编码。

[0058] 可选地,编码器可操作为将输入数据(D1)拆分为多个数据块,并且使用以基本上并发的方式处理多个数据块的并行架构处理器。

[0059] 可选地,在该编码器中,生成器可操作为根据组合在一起的多个数据值生成一个或多个索引集。更可选地,在该编码器中,索引来源于包含R、G和B像素值或Y、U和V像素值的

一个或多个RGB像素或YUV像素。更可选地,编码器可操作为根据数据块被包含在编码数据(E2)时可实现的数据压缩比(可实现的),在将未编码的数据块集合到编码数据(E2)与将编码的数据块集合到编码数据(E2)之间动态地切换。

[0060] 可选地,编码器可操作为将至少一个末尾比特合并至编码数据(E2)中,以指示符号是否属于“代码表的更改”或“数据结束”。

[0061] 可选地,在编码器中,生成器可操作为针对给定的数据块,生成基本上仅足够用于指向存在于给定的数据块中的一个或多个符号所需的索引。

[0062] 可选地,例如通过使用哈夫曼编码对所传递的代码表进行压缩,并且有时,这种压缩方法可选地提供其自身相关的一个或多个代码表。

[0063] 可选地,在相同的数据帧或在接下来的数据帧中能够再利用所传递的代码表,即,数据块编码能够为该数据帧或者先前的数据帧中的其他数据块,再利用任何先于该数据块传递的表或先前的帧中传递的表。

[0064] 可选地,当实现编码器时例如在编码数据中,有益地采用了用于传递表的可选实施方案,或者通过将一个或多个识别代码包括在编码数据来实现编码器,一个或多个识别代码指示例如一个或多个数据库、一个或多个代理数据库和诸如此类从何处易于访问表。

[0065] 可选地,为提供更有效的数据编码、编码数据的传递以及编码数据的解码,如上所述,例如在传递了所存储表的索引的情况中,有益地将所传递的和/或所指向的表例如存储以用于后续使用。根据本公开,这种方法能够减少例如需要从编码器传输至相应解码器的数据的量。

[0066] 根据第三方面,提供了包括其上存储有计算机可读指令的非暂时性计算机可读存储介质的计算机程序产品,其中计算机可读指令可由包含有用于实现根据第一方面的方法的处理硬件的计算机化装置执行。

[0067] 根据第四方面,提供了对由根据第二方面的编码器生成的编码数据(E2)进行解码的方法;提供了在解码器中对由编码器生成的编码数据(E2)进行解码以生成对应的解码数据(D3)的方法,其特征在于,该方法包括以下步骤:

[0068] (i) 接收编码数据(E2),并且从编码数据(E2)提取一个或多个索引集的信息,以及一个或多个频率表、一个或多个代码表、一个或多个代码字长度表、一个或多个概率表、指示这些一个或多个表的信息中的至少之一;

[0069] (ii) 从一个或多个索引集,计算一个或多个代码表、一个或多个频率表、一个或多个代码字长度表、一个或多个概率表中的条目的经压缩的符号、一个或多个数据块中的至少之一中的对应符号;以及

[0070] (iii) 利用来自一个或多个代码表、一个或多个频率表、一个或多个代码字长度表、一个或多个概率表中的至少之一的信息,从符号重新生成一个或多个数据块;以及

[0071] (iv) 对一个或多个数据块进行组合和/或变换以生成解码数据(D3)。

[0072] 可选地,在该方法中,一个或多个表中的至少一个是预先定义的。

[0073] 可选地,该方法包括对解码数据(D3)进行转码以生成对应的转码数据(D4),和/或从编码数据(E2)生成对应的转码数据(D4)。

[0074] 可选地,该方法包括:根据一个或多个表的对称性和对应的对称指示性信息,重新生成一个或多个表。



[0075] 可选地,该方法包括以一个或多个表中的至少一个能够被存储以用于后续再使用的方式,接收一个或多个表中的至少一个。

[0076] 可选地,该方法包括将一个或多个数据解压算法应用在步骤(iv)以生成解码数据(D3)。更可选地,在该方法中,一个或多个数据解压算法包括以下中的至少一种:哈夫曼解码、VLC解码、熵解码、算术解码、区间解码。

[0077] 可选地,该方法包括通过使用以基本上并发的方式处理多个数据块的并行架构处理器,将一个或多个数据块中的多个进行组合以生成解码数据(D3)。

[0078] 可选地,该方法包括根据组合在一起的多个数据值生成一个或多个索引集。更可选地,在该方法中,索引来源于包含R、G和B像素值的一个或多个RGB像素或包含Y、U和V像素值的一个或多个YUV像素。更可选地,该方法包括根据数据块被包含在编码数据(E2)时可实现的数据压缩比,在将未编码的数据块集合到编码数据(E2)与将编码的数据块集合到编码数据(E2)之间动态地切换。

[0079] 可选地,在该方法中,解码器可操作为从编码数据(E2)中提取指示符号是否属于“代码表的更改”或“数据结束”的至少一个末尾比特。

[0080] 可选地,该方法包括通过基本上仅足够用于指向存在于给定的数据块中的一个或多个符号所需的索引,生成给定的数据块。

[0081] 可选地,该方法包括对包含在编码数据(E2)中的一个或多个代码表进行解压缩。更可选地,该方法包括通过使用哈夫曼解码对一个或多个代码表进行解压缩。更可选地,在该方法中,一个或多个代码表的解压缩使用一个或多个附属代码表。

[0082] 可选地,该方法包括以使得一个或多个代码表能够在解码器中使用为对随后发送的数据进行解码的方式,接收一个或多个代码表。

[0083] 可选地,该方法包括从编码数据(E2)中提取一个或多个识别代码,一个或多个识别代码指示一个或多个代码表易于通过一个或多个数据库和/或一个或多个代理数据库从何处访问。

[0084] 可选地,该方法包括对以下类型的数据中的一个或多个进行解码:捕获的音频信号、捕获的视频信号、捕获的图像、文本数据、地震数据、传感器信号数据、模拟数字(ADC)转换数据、生物信号数据、日历数据、经济数据、数学数据、二进制数据。

[0085] 可选地,该方法包括从多个数据源接收编码数据(E2),并且将由源提供的数据进行组合以重新生成编码数据(E2)。

[0086] 根据第五方面,提供了包括其上存储有计算机可读指令的非暂时性计算机可读存储介质的计算机程序产品,上述计算机可读指令可由包含有用于实现根据第四方面的方法的处理硬件的计算机化装置执行。

[0087] 根据第六方面,提供了对由根据第二方面的编码器生成的编码数据(E2)进行解码的解码器;提供了用于对由编码器生成的编码数据(E2)进行解码以生成对应的解码数据(D3)的解码器,其特征在于,该解码器可操作为:

[0088] (i) 接收编码数据(E2),并且从编码数据(E2)提取一个或多个索引集的信息,以及一个或多个频率表、一个或多个代码表、一个或多个代码字长度表、一个或多个概率表、指示这些一个或多个表的信息中的至少之一;

[0089] (ii) 从一个或多个索引集,计算一个或多个代码表、一个或多个频率表、一个或多

个代码字长度表、一个或多个概率表中的条目的经压缩的符号、在一个或多个数据块中的至少之一中的对应符号;以及 (iii) 利用来自一个或多个代码表、一个或多个频率表、一个或多个代码字长度表、一个或多个概率表中的至少之一的信息,从符号重新生成一个或多个数据块;以及

[0090] (iv) 对一个或多个数据块进行组合和/或变换以生成解码数据 (D3)。

[0091] 可选地,该解码器还包括转码器,用于对解码数据 (D3) 进行转码以生成对应的转码数据 (D4),和/或从编码数据 (E2) 生成对应的转码数据 (D4)。

[0092] 可选地,解码器可操作为:根据一个或多个表的对称性和提供至解码器的对应的对称指示性信息,重新生成一个或多个表。

[0093] 可选地,该解码器可操作为以一个或多个表中的至少一个能够被存储以用于后续再使用的方式,接收一个或多个表中的至少一个。

[0094] 可选地,该解码器可操作为将一个或多个数据解压算法应用在步骤 (iv) 中以生成解码数据 D3。还可选地,在该解码器中,一个或多个数据解压算法包括以下中的至少一种:哈夫曼解码、VLC解码、熵解码、算术解码、区间解码。

[0095] 可选地,该解码器可操作为通过使用以基本上并发的方式处理多个数据块的并行架构处理器将一个或多个数据块中的多个进行组合,以生成解码数据 (D3)。

[0096] 可选地,该解码器可操作为根据组合在一起的多个数据值生成一个或多个索引集。更可选地,在该解码器中,索引来源于包含R、G和B像素值的一个或多个RGB像素,或包含Y、U和V像素值的YUV像素。更可选地,解码器操作为通过根据数据块被包含在编码数据 (E2) 时可实现的数据压缩比,在将未编码的数据块集合到编码数据 (E2) 与将编码的数据块集合到编码数据 (E2) 之间动态地切换。

[0097] 可选地,该解码器可操作为从编码数据 (E2) 中提取指示符号是否属于“代码表的更改”或“数据结束”的至少一个末尾比特。

[0098] 可选地,该解码器可操作为通过基本上仅足够用于指向存在于给定的数据块中的一个或多个符号所需的索引,生成给定的数据块。

[0099] 可选地,该解码器可操作为对包含在编码数据 (E2) 中的一个或多个代码表进行解压缩。更可选地,该解码器可操作为通过使用哈夫曼解码对一个或多个代码表进行解压缩。更可选地,在解码器中,一个或多个代码表的解压缩使用一个或多个附属代码表。

[0100] 可选地,解码器可操作为以使得一个或多个代码表能够在解码器 (60) 中使用为对随后发送的数据进行解码的方式,接收一个或多个代码表。

[0101] 可选地,解码器可操作为从编码数据 (E2) 中提取一个或多个识别代码,一个或多个识别代码指示一个或多个代码表易于通过一个或多个数据库和/或一个或多个代理数据库从何处访问。

[0102] 可选地,该解码器可操作为对以下类型的数据中的一个或多个进行解码:捕获的音频信号、捕获的视频信号、捕获的图像、文本数据、地震数据、传感器信号数据、模拟数字 (ADC) 转换数据、生物信号数据、日历数据、经济数据、数学数据、二进制数据。

[0103] 可选地,该解码器可操作为从多个数据源接收编码数据 (E2),并且将由源提供的数据进行组合以重新生成编码数据 (E2)。

[0104] 根据第七方面,提供了包括根据第二方面的用于对输入数据 (D1) 进行编码以生成

对应的编码数据 (E2) 的至少一个编码器, 以及根据第六方面的用于对编码数据 (E2) 进行解码以生成解码输出数据 (D3) 的至少一个解码器的编解码器。

[0105] 可选地, 该编解码器实现为使得至少一个编码器和至少一个解码器在空间上彼此远离, 并且通过数据通信网络彼此连接在一起。更可选地, 该编解码器实现为使得数据通信网络以点对点通信网络来配置。可选地, 编解码器实现为使得其编码器和其解码器对数据的处理过程是对称的; 换句话说, 在编码器中执行的处理函数实施为相应的反函数且在编码器中以相反顺序执行。

[0106] 将理解, 在不脱离由随附的权利要求所限定的本发明的范围的情况下, 本发明的特征适合于以多种组合方式进行结合。

### 附图说明

[0107] 下面将仅以示例的方式参照附图描述本公开的实施方式, 在附图中:

[0108] 图1是用于对数据进行编码和解码的已知编码器和已知解码器的图示;

[0109] 图2是根据本公开的实施方式对数据进行编码的方法的图示;

[0110] 图3A是根据本公开的组合作为编解码器的编码器及解码器的实施方式的图示; 以及

[0111] 图3B是根据本公开的组合作为编解码器的编码器及解码器的可替代实施方式的图示; 其中在解码器中对解码数据D3进行转码以生成转码数据D4。

[0112] 在附图中, 使用了添加下划线的数字以表示下划线数字所在位置上的项或者与下划线数字相邻的项。当数字未添加下划线且附有相关的箭头时, 该非下划线数字用于识别箭头所指的通用项。

### 具体实施方式

[0113] 大体上, 本公开例如涉及编码器、解码器、编解码器和相关的操作方法。此外, 本公开的实施方式相比于已知方法能够改进代码表、频率表、代码字长度表或概率表的传递操作。另外, 本公开的实施方式还能够以使得更少比特用于传递一个或多个表的方式来传递一个或多个哈夫曼树, 从而在数据编码期间可实现数据压缩比的提升, 在伴随一个或多个表的编码数据量相对小时尤为如此。代码表、频率表、代码字长度表或概率表是很多不同的熵编码方法所需的, 例如霍夫曼编码、算术编码、区间编码的可变长度编码 (VLC) 方法, 但是并不限于此。诸如发射器的编码器和诸如接收器的解码器均有益地采用将在下文中描述的方法。

[0114] 下文中描述的本公开的实施方式涉及存储和传递的数据量随着时间流逝快速增长的情况。这种数据存储和传输消耗相当大的存储容量、传输带宽和能量。该情况中的大部分数据为捕获的音频信号、捕获的视频信号、捕获的图像、文本数据、地震数据、传感器信号数据、模拟数字 (ADC) 转换数据、生物信号数据、日历数据、经济数据、数学数据、二进制数据, 但是并不限于此。本公开的实施方式可操作为减少用于上述所有数据类型及其他类型的数据的编码数据量; 从而能有效地传递代码表、频率表、代码字长度表或概率表, 因此能够使用有效减少数据的熵 (即有效减小数据的尺寸) 的较小数据块。此外, 能够通过并行处理有效地处理较小数据块以更快速度输出结果, 并且这种并行方式在现代微处理机体系结构中、尤其在将来微处理器的配置中是普遍的, 例如RISC (精简指令集计算机) 处理器的数

据处理机阵列和高速配置。

[0115] 对于给定的编码方法,相应的代码表包括表示代码字长度的信息,例如以比特、表示代码字的代码以及代码字的索引(即“指示(indexes)”)。代码表还可由代码字的长度生成。代码字的索引(即“指示”)表示通过代码字进行编码的对应原始符号的值。类似地,频率表包括符号的出现频率和符号的索引。符号的索引表示分别通过索引被编码的原始符号的值。频率表可转变为概率表,并且概率表可用作频率表的粗略估计。频率表与代码字长度之间的转化及代码字长度与频率表之间的转化也是可以实现的。

[0116] 当传递上述表中的任意一个时,这种传递中的一种非常重要的参数为表的最大索引。表的最大索引表示在所传输的表中以及在输入数据中存在多少可用的不同符号或可能的不同符号。例如,如果已知数据为:4,3,0,1,0,4,3,4,

[0117] 则真正的最大索引是4,并且最小索引是0,这意味着可能有5(最大-最小+1=4-0+1)个不同的符号(0,1,2,3,4)存在于该数据中。由于实际上只有4个不同的符号(0,1,3,4)存在于数据中,因此也可选地通过使用“3”作为最大索引(即,可用的不同符号的个数)而非“4”(即可能的不同符号)来传递该表。当值3用作表的最大索引值时,则需要用于传递与哪些符号用作每个表索引有关的信息的其他一些机制。

[0118] 当符号是按顺序时,可选地传递真正最大索引(4)和可用性比特,例如在本示例中为11011。这种传递映射使得表索引0等于标注为索引的符号0,并且相关的符号对则为(0,0)。类似地,索引与符号对的其余部分为(1,1)、(2,3)和(3,4)。可选地,也可直接使用索引和符号对来定义不同的表索引所使用的索引,然后将“3”作为表的最大索引进行传递。例如在被传递的表中的符号根据其频率被排序的情况下,这是种很有价值的方法。

[0119] 例如,在上述示例中,索引和符号对例如为(0,4)、(1,0)、(2,3)和(3,1)。有时,所使用的索引和符号对是预先定义的,而在其他情况下,则传递所使用的索引和符号对表的索引。在另一不同的情况中,将索引和符号对与编码数据(E2)一同传递。在另一不同的情况中,解码器60可操作为从已知位置获取所使用的索引和符号对;在其他不同的情况中,解码器60可操作为从其对应的位置信息已与编码数据(E2)一同被传递的位置获取所使用的索引和符号对。

[0120] 在一些情形中,有益地借助于对称性来估计待传递的表。使用对称性使得利用相比于传递整个表所需的数据尺寸较小的数据尺寸来传递编码表成为可能,而不管该表是否基于代码长度、基于概率或基于出现频率。此外,由于对称表包含较少元素,即较少的符号,因此在编码器50和解码器60中均较快速地生成对称表。然而,利用对称性使得表稍显次优,但是所传递的表数据尺寸上的节约整体上或更多地补偿了其损失,在发送的数据量相当少时尤为如此。

[0121] 在以上最后一个示例中,可选地在传递表时使用对称性,因为符号值“0”相比于“1”的概率大,相应地,符号值“4”相比于“3”的概率大。此外,符号值“0”和“4”的概率彼此接近,相应地,符号值“1”和“3”的概率也彼此接近。然而,值“2”在数据中根本不存在,因此不管从哪个角度观察它都是概率最小的值。

[0122] 因此在使用对称性的情况下,总能够根据对称对应的元素的出现总数而生成编码表。在这种情况下,符号值“0”和“4”总共出现5次,相应地,符号值“1”和“3”总共出现3次。元素“2”在数据中根本未出现,因此不需要为其提供单独的符号。然而,在某些情形中,也能够

为元素“2”生成符号,并且在这种情况下,当使用对称性时,右侧表和左侧表中将均包括该符号。

[0123] 因此,可选地以如下方式生成区间编码表,即区间的出现次数为5、3和(0),虽然基于最优出现频率的区间表为3,2,0,1,2且相比于如上所述的仅需要发送(即传递)出现频率的两个值的、基于对称性的最优实施方式,其将需要传递出现频率的四个值,但是对于符号零至四所使用的区间表则为5,3,0,3,5。

[0124] 基于对称性的相同概念可选地与诸如哈夫曼编码的其他编码方法一同使用,并且在这种情况下,基于对称性的表例如为:左侧值接收代码“0”而右侧表值接收代码值“1”的表。因此,霍夫曼编码字例如为:01,00,不存在10,11。替代地,若希望为值“2”保留选项以备将来出现,则代码将为01,001,000/100,101,11。在该实施方案中,原则上发送/传递只需两个代码长度(1和1),或者三个代码长度(1,2,2),并且在已知表使用对称性的情况下所使用的表在解码器60中可完全恢复。在较长的表中,可更加明显地看出其优点。

[0125] 将理解,可选地,预先已知晓关于表是否使用对称性的信息片段,或者该信息片段是否通过与以下信息片段相同的方式传输/传递,即与是否使用最优表或预先定义的表或从先前的表动态生成的表有关的信息片段。传递与表是否使用对称性有关的信息片段通过将所使用的表的索引发送至解码器60来实现。

[0126] 例如:

[0127] (i) 表索引“0”意味着发送/传递整个表;

[0128] (ii) 表索引“1”意味着表是对称的并且将只传递其一半。

[0129] (iii) 表索引2至63意味着使用了预先定义的表;以及

[0130] (iv) 表索引64至127意味着使用了先前传递并存储的动态表。

[0131] 将理解,对称表也可用作预先定义的表或动态存储的表。

[0132] 可选地,使用多种编码方法,例如使用或不使用ODelta编码的方法,其中ODelta编码涉及将数据区别地编码为值0和1的序列并且采用概括计数。此外,当使用这些各种编码方法时,有利地将这些方法与使用中的表的表索引组合使用,并且可选地表达是否在编码之前对数据执行了ODelta方法以及解码器60是否需要随后在解码之后执行反向操作。

[0133] 在这种情况下,例如,表索引原本是一样的,但是添加了值128<sub>dec</sub>以表示使用了ODelta。如果没有执行该插入操作,则在编码之前不执行ODelta。当然,可选地,也可将其他值添加到表索引值;然而,将本质上理解表索引表示哪些类型的表被使用,以及哪些类型的附加数据与表索引以及编码数据一同被传输/传递。

[0134] 参照图2和图3A,提供了对输入数据D1进行编码以生成对应的编码输出数据E2的方法步骤的图;该方法的步骤大致上表示为10,并且可选地使用了以下步骤中的一个或多个:生成一个或多个频率表25的步骤20;生成一个或多个编码表35的步骤30;对输入数据D1进行分析以选择最适合的编码方法的步骤40,但是并不限于此。当实施这种对输入数据D1进行编码的方法10时,需要一些机制,这些机制可用于将存在于输入数据D1中的一个或多个符号改变为对应的索引(即“指示”);例如,包含在索引中的给定索引等于例如在像素阵列图像中的像素值。该索引还可等于像素值减去存在于像素阵列图像中的最小像素值。在这种情况下,该方法还需要传递编码输出数据E2中的最小像素值,即通过使用方法10或者通过使用其相反的方法从编码器50传递至对应的解码器60,因为如果不这样的话,解码器

60不能使用其给定的索引将对应的给定符号解码回其原始值。该索引还可从多种信息创建,例如通过以下所列信息中的一种或多种:离散余弦转换(DCT)、包含绝对AC系数值的单项AC系数、AC系数的符号、零AC系数与先前的非零AC/DC系数之间运行零AC系数、以及代表与当前AC系数有关的信息的指示标记,以及最后一个非零AC系数。索引还易于根据组合在一起的多个像素值,例如包含8比特R、G和B像素值的24比特RGB像素或包含两个5比特Y像素值的10比特值而创建。

[0135] 参照图3B,可选地例如在以下的多播情形中,通过转码器70实现编码器60中对数据D3的转码以生成对应的转码数据D4,其中:

[0136] (i) 存在多个装置,每个装置包括用于接收编码数据E2的解码器60;

[0137] (ii) 多个装置中的至少某些彼此不同,并且具有彼此不同的相关的输出格式,例如显示屏外观、分辨率、宽高比、显示屏驱动器缓冲容量等。

[0138] 对数据D3的转码需要生成能够通过硬件和/或通过支持装置的软件层以兼容的方式处理的对应数据D4;该装置可选地基于计算硬件,例如智能手机、专用科研装置,电视机、高保真装置、视频会议装置和类似装置。转码器70实现为软件和/或诸如ASIC的专用数据处理硬件。

[0139] 如下面将提及的,当实现方法10时,通常必须包括将一个或多个符号值转换为一个或多个相应索引的步骤,并且必须将该步骤或者与其相反的步骤传输至解码器60,否则需要为编码器50和解码器60均预先设置这种步骤。实现这种传输的一种最简单方法是利用给定符号值与其对应的索引值之间的直接关系,例如,一个索引值等于对应像素值,或该索引值是基于表示 $S$ =符号标记、 $V$ =10比特系数值、 $R$ =6比特非零运行值、以及 $L$ =末尾标记的比特的数字,例如,表示为:

[0140] S V V V V V V V V V V R R R R R L

[0141] 由于多种原因,使用直接关系并不总是可行的,例如,当使用直接关系时:

[0142] (a) 对数据、索引(即“指示”)、频率、概率或字符长度进行编码或解码是不可行的或缺乏效率的;

[0143] (b) 不同索引的数量庞大;或者

[0144] (c) 不是所有的符号都具有可用的频率信息并且由于此原因某些算法无法为这些符号生成代码。

[0145] 如上问题(a)至(c)中的某些能够(至少部分)通过使用转义码或使用为所有符号生成频率信息的某些逻辑得到解决。经常,仍有益地使用将符号转换为索引(也称为“指示”)的其他一些方法。其中一种方法经常涉及确保存在一些指定用于可用符号的索引的查询表(LUT);这里,转义码对减小编码表的尺寸是非常有益的。该LUT必须在编码器50和解码器60中是可用的,或者需要从编码器50传递至解码器60,反之亦然。当需要更多最优编码以便能够实现更好的压缩时,有益地使用可根据可用的LUT索引选择的多个表。然而,有时这是不实用的,因为频率或代码字的长度、组合太大以至于将所有不同的表存储至数据存储器中是没有意义的,或者这种LUT的传递需要在编码器50与解码器60之间传输太多数据。因此,根据本公开的方法10能够通过使用适当的符号索引一个或多个变换,来实现频率、代码字长度或概率从编码器50到解码器60的有效传输。如果编码方法无法利用以频率信息的方式提供的更精确的信息,那么使用代码字长度代替频率总是有益的,例如,VLC编码方法

不能使用频率信息,但是相反地,算术编码和区间编码能够利用频率或概率信息。

[0146] 下面将更详细地描述本公开的实施方式的示例,该示例中使用了用于编码目的的代码字长度;但是,使用频率信息或概率信息的对应实施方式也是可行的。

[0147] 在图2所示的编码数据E2中,例如在编码数据流中,包括可包含20个符号的数据值,即从0至19的值,但是在当前数据流中实际上可用的只有其中的8个数据值,即最小值=2和最大值=19。这些最小值和最大值同样可选地如以下将描述的分开传递,以便能够在传递表时实现更多的节约。例如,如以下表1所示,对于各符号,对应的频率总和为148、代码字长度具有最小长度=1和最大长度=6、以及索引(“指示”)如表所示,可以确定的是,在不压缩这些符号的情况下,需要 $148*5$ 比特=740比特来传输比特流。

[0148] 表1:示例性比特流编码

[0149]

值	频率	CW 长度	频率 1 (比特)	频率 2	频率 3 (比特)	索引 1	索引 2
2	7	4	4(4)	84	128(4)	2	0
4	2	6	1(6)	24	32(6)	4	1
7	81	1	32(1)	972	1024(1)	7	2
9	1	6	1(6)	12	16(7)	9	3
12	35	2	16(2)	420	512(2)	12	4
13	9	4	4(4)	108	128(4)	13	5
14	5	5	2(5)	60	64(5)	14	6
19	8	4	4(4)	96	128(4)	19	7

[0150] 对于表1中所示的编码方法,如果在编码器50和解码器60中不存在可用的且适当的频率表或代码表,例如通过参考索引预先定义的或可列举的频率表或代码表,则潜在地存在可用于将这些所需代码字(即,代码和长度)从编码器50发送至解码器60的若干方法。

[0151] 第一种示例性方法在修改数据的频率之后生成对应的代码表,在该代码表中,最小可能符号(即,最长代码字)被分配一个附加比特,并且数据中不可用的所有符号被分配足够长的代码字长度使得他们不进行编码,但是使编码器50和解码器60能够创建彼此相似的频率表以及彼此相似的哈夫曼树。由于缺少12个符号,这种频率修改例如可通过将原始的频率乘以12并且为所有不具有真实频率值(即,频率值等于0)的符号设置频率值1来实现。对于具有可用频率的符号,其经修改的频率值例如可参见表1的频率2。根据这些新的频率,所有20个字符的代码字长度可创建为:11,11,4,11,6,11,11,1,11,7,11,11,2,4,5,10,10,10,10,4

[0152] 使用这种代码表将经编码符号从编码器50传递至解码器60需要 $(7*4+2*6+81*1+1*7+35*2+9*4+5*5+8*4=)$ 291比特。为了准确的数据编码和随后的数据解码,编码器50和解码器60需要使用相似的频率,其中这种频率可根据这些新的代码字长度产生,并且对于具有频率的符号,其结果可参见频率3 $(=2^{(\text{maxbitlen}-\text{bitlen})})$ ;“maxbitlen”是“最大比特长度”的缩写,并且“bitlen”是“比特长度”的缩写。其他符号被分配频率2(bitlen=10)或1

(bitlen=11)。

[0153] 该第一种方法能够生成包含所有可能符号(即,前述示例中的20个符号)的代码表;其在相同的代码表也用于其他相似类型的数据时是有益的。也能够可选地通过任意压缩方法对这种类型的长度进行压缩,而不需要任何附加的信息,因此可能在所有的情况中易于在编码器50与解码器60之间传递代码表。例如,在不进行压缩的情况下,该代码表需要4比特以表示所有的代码字长度,即,20长度\*4比特/长度=80比特。

[0154] 由于该第一种方法对每个最小可能符号只分配1个比特(即,在该第一种示例性方法中等于1比特),因此相比于最优哈夫曼编码缺乏效率。此外,这些比特也需要用于对代码字长度(即,代码表)进行压缩以及将其从编码器50传递至解码器60。可选地传递数字20(即,所有可能符号的数量),或者数字20也可以是解码器60已知的。

[0155] 接下来,将描述第二种示例性方法以说明本发明的替代的实施方式。第二种方法仅为可用的符号生成代码字长度,即那些频率值大于0的符号。用于生成哈夫曼代码表的索引为索引2(见表1),但是需要从编码器50传递至解码器60的索引为索引1,即等于符号值(见表1)。生成的代码字长度可参见CW长度列。使用这种代码表传递编码符号需要(7\*4+2\*6+81\*1+1\*6+35\*2+9\*4+5\*5+8\*4=)290比特。根据这些代码字长度,编码器50和解码器60可操作为创建如表1的“频率1”列所示的频率表。

[0156] 这种类型的代码表的第一种传递方法将代码字长度和代码字的索引作为一对数字进行传递,如下:

[0157] (2,4), (4,6), (7,1), (9,6), (12,2), (13,4), (14,5), (19,4)

[0158] 其中,用括号表示一对。

[0159] 在这种传递方法中,每个索引需要5比特,并且每个代码字长度需要3比特,即每一对需要8个比特,因此总共为8\*8比特=64比特。

[0160] 这些索引(即“指示”)还可被delta编码并且数字对为如下:

[0161] (2,4), (2,6), (3,1), (2,6), (3,2), (1,4), (1,5), (5,4)

[0162] 现将理解,每个索引仅需要3比特并且每个代码字长度仅需要3比特,即每一对需要6比特,因此传递代码表需要8\*6比特=48比特。

[0163] 有益地将这些索引和代码字长度分开以具有对应的数据流,相比于组合的8比特或6比特值,这通常得能够改善压缩。数据流现在为:2,4,7,9,12,13,14,19

[0164] 以及 4,6,1,6,2,4,5,4

[0165] 则总共为8\*5比特+8\*3比特=64比特。

[0166] 当第一数据流的索引(“指示”)被delta编码时,结果生成:2,2,3,2,3,1,1,5

[0167] 以及 4,6,1,6,2,4,5,4

[0168] 则总共为8\*3比特+8\*3比特=48比特。

[0169] 当可能的符号数量较多而数据仅包含少数彼此不同的符号时,这是最佳(即最优)的传递方法。

[0170] 在操作中,所有前述这些数据流可被压缩并从编码器50传递至解码器60。相比于最优哈夫曼编码,该方法不存在缺乏效率的问题,但是本公开的方法仍然消耗相当数量的比特以传递包含有索引(“指示”)和代码字长度的信息的流。还需要传递数值8,即可用符号的数量,否则解码器60无法确定应将多少个值或多少对解码至代码表。在这种情况下,不需



要在编码器50与解码器60之间传递数字20,即所有可能的符号的数量。

[0171] 上文所描述的方法中的某些适合以组合的方式使用,尤其在实现编码器50中对输入数据D1的良好编码以生成对应的编码数据E2时,并且同样地在解码器60中进行相反操作时亦是如此。可选地,所有这些解决方案例如通过类似于第二种方法的方式仅为可用的符号创建哈夫曼代码。此外,从这些数据长度生成的频率表可参见频率1列(见表1)。

[0172] 相比于第一种方法,第二种方法还可在符号不存在时将代码字长度设置为零,但是仍需将这种信息从编码器50传递至解码器60;这种方法可行是因为代码字的真实长度永远不可能为零。由此得出以下长度流:

[0173] 0,0,4,0,6,0,0,1,0,6,0,0,2,4,5,0,0,0,0,4

[0174] 其中,所有的代码字长度只需要3比特,即 $20 \times 3$ 比特=60比特。

[0175] 这种类型的长度也可通过任意压缩方法进行压缩,而不需要任何附加信息,因此在所有的情况中,易于在编码器50与解码器60之间传递代码表。另外,这种数据也经常包含许多零值,因此例如通过使用VLC或RLE可以很容易对其进行压缩。可选地将数字20(即,所有可能符号的数量)从编码器50传递至解码器60,或者其对解码器60是先验已知的。

[0176] 另一实施方式使用比特来指定哪些数据符号可用而哪些不可用。在生成两个数据流的情况下,其中第一流包括比特并且第二流包括代码字长度,如下:

[0177] 0,0,1,0,1,0,0,1,0,1,0,0,1,1,1,0,0,0,0,1

[0178] 以及

[0179] 4,6,1,6,2,4,5,4

[0180] 其中每个符号需要1比特,并且每个可用代码字的长度需要3比特,则总共需要 $20 \times 1$ 比特+ $8 \times 3$ 比特=44比特。这通常是最佳(即最优)的传递方法。

[0181] 还可通过利用熵调节器(EM)和/或VLC对这种流进行进一步压缩。如上所述,可选地在编码器50与解码器60之间传递数字20(即,所有可能符号的数量)或者将其预先提供至解码器60。此外,不需要在编码器50与解码器60之间传递数字8(即,可用符号的数量),因为其可通过值1的比特计算出。

[0182] 对于根据本公开的前述所有方法,编码器50和解码器60需要具有与多少不同的符号能使用有关的信息(例如,前述示例中的20个符号)以及与多少个不同的符号为可用的有关的信息(例如,前述示例中的8)。如果解码器60中不存在该值(即,不同符号的数量),则需要将其传递至解码器60。可选地,可将某些数据存储在通过发送少量附加信息来传递的表中,以指定数据值可用的范围,例如,传递值2(最小值)和19(最大值)以指定值被包含的范围。在该示例中,这种传递经常使用相比其存储的比特更多的比特,但是例如在8比特像素仅包括值60至200的情况下,极大地节约了待从编码器50传输至解码器60的比特。这种范围的传递使得不需要在编码器50与解码器60之间传递小于最小值的值以及大于最大值的值。此外,将理解,在索引值经过或不经过delta编码被发送的情况下,该范围从编码器50传递至解码器60时,不需要传递第一个和最后一个索引值。同样的处理也适用于最后一个示例中的第一个以及最后一个值1比特。可选地,在采用其他方法,例如在于2013年3月1日由Gurulogic Microsystems Oy提交的专利申请GB1303661.1中所公开的ODelta编码方法时也可利用最小值和最大值的传递,上述申请通过引用并入本文。传递最小值和最大值在用于调节熵和实现熵编码的所有方法使用相同的信息并且将该信息仅传递一次时发挥其最

大的优势。

[0183] 可选地,例如根据给定数据块中存在多少个待编码符号而选择性地使用如上所述的方法,其中该数据块例如是从待由编码器50传输至解码器60的数据整体中划分的。因此,对前述方法的选择取决于多少个不同的符号为可用的、其中真正使用了多少个、最小可能符号的频率如何、以及可用符号的索引(即“指示”)是如何分布到可能的符号。

[0184] 根据频率值,也可为表1中所示的符号计算经缩放的概率。符号的数量例如为148。该示例中的经缩放的概率有益地通过使用两个不同的概率乘数,即256和32。对第一个符号使用概率乘数,由此计算出经缩放的概率值为 $\text{Round}(256*7/148) = 12$ ,其中“Round”为整数四舍五入函数。因此,通过乘数256计算出的所有经缩放的概率值如下:

[0185] 12,3,140,2,61,16,9,14

[0186] 经缩放的概率值的总和为257,即大于256,并且有益地将某些值减少1。有益地使用了这种减少以最大限度地降低对实际编码的影响。例如,在该示例中,作为最小值的值2可减小为值1,或者作为最小四舍五入值的值9可相应地减小为值8,以使得通过区间编码或算术编码的乘数256计算出的经缩放的概率值如下:

[0187] 12,3,140,1,61,16,9,14(总和=256=概率乘数)。

[0188] 可如下传递经缩放的概率(通过乘数256):

[0189] 0,0,1,0,1,0,0,1,0,1,0,0,1,1,1,0,0,0,0,1

[0190] 以及

[0191] 12,3,140,1,61,16,9,14

[0192] 当概率乘数值为32时,经缩放的概率值如下:

[0193] 2,0,18,0,8,2,1,2

[0194] 并且经过和均衡之后为:

[0195] 1,0,18,0,8,2,1,2

[0196] 现将理解,对于经缩放的概率中的一些计算出零值。这意味着例如通过使用转义符号必须传递这些值。需要计算出用于转义符号的经缩放的概率,并且其可能小于值1。在这种情况下,将为其分配值1,因为 $\text{Round}(32*(2+1)/148) = 1$ 。下面,需要将这些转义符号添加到其他符号,则新的符号集为:“转义”、2、7、12、13、14和19。这些新的符号有益地被分配0至6范围内的索引。当转义符号的概率从一个或多个其他符号减小时,用于区间编码或算术编码的新符号的经缩放的概率如下:

[0197] 1和1,18,8,2,1,1(和=32=概率乘数)。

[0198] 则可以如下传递经缩放的概率值(通过乘数32和转义代码):

[0199] 0,0,1,0,0,0,0,1,0,0,0,0,1,1,1,0,0,0,0,1

[0200] 以及

[0201] 1,1,18,8,2,1,1

[0202] 将理解,第一个符号是转义符号并且比特指定其他符号值。

[0203] 如上所述,当限定了转义符号时,也可在将来更好地利用该表。通过使用转义符号,如果数据中不存在的符号在后续的数据中出现则仍可传递。

[0204] 于2014年2月20日由申请人Gurulogic Microsystems Oy提交的名称为“编码器、解码器和方法”的申请号为GB1403038.1的另一专利申请中示出了通过区间编码使用转义

代码,该申请通过引用并入本文。

[0205] 如在表1的描述中提及的,当在编码器50中对数据D1进行编码并且随后在解码器60中对其进行解码时,可选地,也可使用预先定义的表或通过索引描述的表,而不是传递的表。这意味着所使用的代码、频率、概率或代码长度表对于编码器50和解码器60是先前已知的,或者从有限的替代表集中选择的,并且编码器50将该选择传递至解码器60。可选地,预先定义的表在本地对解码器60是可用的。

[0206] 可根据所传递的参数预先存储表,例如,表的索引和/或表的最大索引。可替代地,可通过初始化函数实现的算法或通过预先存储在表里的算法生成表,表的这种创建(即生成)也是基于所传递的参数,例如基于表的索引和/或表的最大索引和/或表的最小索引。除了预先存储的表之外,可选地,也可使用在对数据D1进行编码之前传递的表来生成编码数据E2。

[0207] 例如,当使用VLC编码时,通常提前为不同的表索引存储代码长度,并且可以基于这些长度值通过使用整个表或仅使用表值的一部分来生成代码表。可基于所传递的表参数或基于表索引来定义所使用的部分。类似地,当使用区间编码时,通常基于表的形式(例如,利用表索引传递的)以及表的长度(例如被传递为表的最大索引)来生成概率表。

[0208] 下面将描述本公开的另一示例性方法,该方法在不使用单独的转义符号的情况下能够实现高效的代码传递,同时还使得能够高效地编码当前给定的数据库以及具有稍微不同的符号频率的后续数据块的所有符号。该另一方法可实现为使得具有可用值和不具有可用值的所有符号至少被分配作为经缩放的概率的值。如果缩放的概率值为1,则可用性比特将等于0,并且对于其他经缩放的概率值,可用性比特将等于1。则只需传递可用性比特等于1的符号的经缩放的概率值。该方法的详细信息可参照申请人Gurulogic Microsystems Oy的GB1403038.1号专利申请,该申请的内容通过引用并入本文,但是在以上先前示例中的表传递则变成:

[0209] 0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,0

[0210] 以及

[0211] 10,4(和=18\*1+10+4=32=概率乘数)。

[0212] 该示例示出了有效的解决方案,在对数据的熵进行编码时,相比于霍夫曼编码,该方案产生非常类似于区间编码的效果。这种解决方案能够非常有效地传递代码概率表。通过其他类型的数据或通过其他类型的概率乘数值,该方案明显地在很多情形中是可使用的最好的,即最优的编码方法。因此,下面将更详细地描述其代码表的传递。

[0213] 可选地,编码器50和解码器60能够以静态的方式或动态可更新的方式存储所有表,编码器50和解码器60将利用这些表。如果表被存储,则在从编码器50发送至解码器60的数据中通过索引有益地识别这些表,其中索引唯一地识别其对应的所存储的表。这种表的索引潜在地节省了很多原本在将代码表从编码器50传递至解码器60时需要发送的开销数据。

[0214] 例如可通过以下的示例来理解预先存储的表的使用。对于所传递的表,出于再利用的目的来选择添加先前示例中所指示的最后一个概率表,即1,1,1,1,1,1,1,10,1,1,1,1,4,1,1,1,1,1,1;为此目的,表例如被分配索引值“17”。下面需要对新的数据块进行编码,并且其具有表2所提供的符号值和频率。

[0215] 表2:用于表的重复使用的第二符号值和频率的示例

[0216]

值	频率
1	3
5	1
7	68
8	1
10	4
12	32
14	3
18	1

[0217] 下面可通过表2估算出用于区间编码的所有可用的概率表,并且很有可能选择表17作为用于通过区间编码将新数据从编码器50传递至解码器60的最佳概率表。至少,容易看出,对于理想的熵编码结果而言,新概率表的传递所需数据相比于表17所创建的附加数据的量多很多。因此,可以重复使用表17或其他一些概率表,而不需要传递新的更优化的概率表。当重复使用表时,将其索引17从编码器50传递至解码器60,然后可将区间编码值从编码器50传递至解码器60。当无法重新使用表时,先将定义新的表传递的例如为0的值或者下一可用表索引从编码器50传递至解码器60,然后还将表从编码器50传递至解码器60,然后,可将区间编码值从编码器50传递至解码器60。经常,例如一般在区间编码数据之前也需要传递编码符号的量,这使得在解码器60中能够对数据进行恰当的解码。

[0218] 可选地,还可修改已在使用中的代码表,并且仅发送其中的更改,从而产生新的代码表。还可选地,可在传递/接收之后以自适应方式使用已传递的代码表。

[0219] 可选地,编码器50和解码器60可操作为创建整体相似的表,该表还能够对其他符号进行编码以用于将来使用类似类型的数据,该数据中可能存在当前不存在的符号。可通过新的代码表索引存储并设置这些表。可存储两种类型的表,即不完整的和完整的。此外,也可仅存储原始的表,并且在下一次需要完整的性能时,可在从编码器50到解码器60通信时单独指出。存储完整表的方案是更优选的,因为其使得决策简单化并且不需要传递表示所需表是否为完整的额外指示信息。表的完整还可实现为利用例如从4减少至1的频率填充包含所有值的表,使得接下来的符号分配有相对短的符号,并且之前的符号分配有相对长的符号。通过使用这种方法,符号的顺序与将来数据流中的可用符号的顺序对应。

[0220] 在如上所述的示例中,虽然用于传递哈夫曼编码、算术编码或区间编码所需的频率值可使用类似的技术,但是有益地使用了代码字长度的传递以及经缩放的概率的传递。有益地,采用的最佳编码方法在编码数据添加有传递代码表、代码字长度、概率或频率表所需的开销信息时使用尽量少的比特,由此例如能够利用专门为数据的较小部分(即数据块)中所包含内容的性质和/或数据类型而优化的编码方法,将数据的较小部分从编码器50发送至解码器60。因此,当频率值至少被量化为有限程度时,能够实现算术编码和区间编码的最佳效果,使得频率表代表能够利用明显比精确表更少的比特传递的近乎准确的值,因此

例如在使用熵编码时,对于符号的编码,最佳性仅降低了少许。此外,经缩放的概率表的传递实现了非常有效率的并且近乎最优的区间编码和算术编码。

[0221] 当只有少量的数据从编码器50传输至解码器60时,通常最好在基本上不使用任何形式的编码的情况下传输数据。然而,当数据的量增加时,有益地采用具有近乎准确的代码字长度的哈夫曼编码。随着从编码器50传输至解码器60的数据的量增加,使用更准确的代码表逐渐获得了更多益处。此外,当存在相当大量的、待进行算术编码或区间编码的数据时,使用更有效的熵编码时获得最佳的编码结果;因此相对于传递代码表所需的比特,在传递频率或概率所需的比特上实现了编码优势。对概率表的索引、频率表、代码字长度表或代码表的传递总是相似的,并且当使用索引时,具有最佳可用表的方法能够实现最佳的压缩性能。如果对所使用的编码方法的选择未根据数据或数据的量而固定,则也需要将该选择从编码器50传递至解码器60。

[0222] 编码器50和解码器60共同地形成编解码器100。在实际情形中,可存在一个编码器50和一个或多个解码器60,例如在编码器50生成编码数据E2的情况下,其中编码数据E2例如通过无线、光纤传输网络以及类似介质被广泛地传播至多个解码器60,即“组播”。此外,可出现例如点对点数据通信网络的情况,其中解码器60从连接至点对点网络中的若干编码器50接收其编码数据E2,其中编码数据E2在一同收集在解码器60处的单独编码数据块中提供;由于编码数据E2的某些部分对于解码器60而言更加本地化地接收,并且减轻了用于实现这种点对点网络所采用的长距离数据通信网络链接上的数据负载,所以这种配置是有益的。编码器50和解码器60易于实现为计算硬件中的专用数字化硬件,其中计算硬件可操作为执行存储在非暂时性数据存储介质中的一个或多个软件产品或其组合。编码器50和解码器60可用于音频录制和/或回放装置、视频录制和/或回放装置、个人计算机、智能手机、数字摄像机、视频摄像机、电视机、互联网终端、科研装置、监控和/或安全系统、配置有地面监控功能的卫星、地震感测系统,但是并不限于此。

[0223] 本公开的实施方式能够实现表例如编码表、频率表、概率表或代码字长度的有效传递,从而易于将数据拆分为较小的数据块,例如可随后以最优方式单独进行编码。此外,可选地利用熵编码方法对表进行编码。这些较小的数据块可选地需要其各自对应的代码表、频率表、概率表或代码字长度;这在具有很多不同的可用表时这是有益的,因为只需要为不同的数据块传递给定表的索引即可。否则,还需要将新表传递给解码器60。当传递给定表时,通常有益地将其存储至数据存储中(例如,解码器60的数据存储器)以在将来使用其自身唯一的指向索引。

[0224] 在本公开的示例性实施方式中,通过发送描述多少个数据块属于彼此相似的数据体的支持信息,将每个数据块作为单独的数据块进行传递;这种数据块的传输方式通常非常缺乏效率,因为对于所有数据块而言,需要传递所采用的编码方法的标识、符号的量以及所使用的代码表、频率表或概率表。另外,还需将属于彼此相似的数据体的数据块的数量从编码器50传递至解码器60。

[0225] 可选地,在代码表中,有益地采用插入具有其自身对应含义的一个或多个附加符号的方式。通常,在大的代码表中,“转义”符号有益地具有其自身的代码字。此外,有益地,在用于JPEG的DCT系数的编码表中还存在可用于“系数的结束”符号的代码字。这意味着该方法对于解码器60是已知的,使得可通过添加新的编码符号以非常有效的方式使用该方

法,其中新的编码符号例如可用于“代码表的更改”、“数据结束”以及如果需要的话还用于“转义”。这些附加的符号可生成为使得每当这些符号被使用时它们的频率为1。如果使用了可用的表,则添加对应的标识作为符号,以使得其通过具有最长代码字的符号拆分代码。如果还有剩余的数据,例如在当前数据块之后存在新的数据块,则编码器50有益地使用“编码表的更改”符号而非“数据结束”符号。当传递该“编码表的更改”符号时,在其之后传递新的编码表的索引。例如在不存在可用的表时使用定义新的编码表的索引值,并且当已存在可用的表时使用从1至表的数量的索引值。可选地,该新的编码表的索引具有这样的值,即该值使用与展示用于数据的所有可用的或适当的表所需的比特一样多的比特。如果值0作为新的编码表的索引进行传递,则在将接下来的符号被编码为从编码器50提供至解码器60的数据流之前,需要传递代码表。否则,在识别新的编码表的索引之后,可在该新的编码表中立即对新的符号进行编码。当最后一个数据块被编码并且最后一个数据值被传递至解码器60时,编码器50传递“数据结束”符号。在这种情况下,仅“数据结束”符号是有效的并且未使用“编码表的更改”。当传递“数据结束”符号时,其后不需要继续传递任何数据。该“数据结束”符号使得不需要为每个数据块传递数据值的数量。此外,也不需要将编码方法传递至解码器60,因为对于不同的数据块,只改变了所使用的代码表、频率表、概率表或代码字长度。因此,在数据的编码及随后的解码期间,当代码表被更改时,待从编码器50发送至解码器60的开销数据的总量非常小。需要一个末尾比特来检测符号是否属于“代码表的更改”或“数据结束”,或者可通过将代码表的频率设为1来生成上述两个符号。

[0226] 有时,根据数据、数据量、使用的编码方法和解码器60和编码器50的实现方式有益地发送数据值的量、编码数据的量或者使用“数据结束”符号。此外,可选地在编码器50和/或解码器60中处理数据时采用平行性是有益的,即传递了编码数据的量并因此解码器60能够容易地为不同的处理器、进程和线程拆分数据。通常,存在多种最适合传递与需要对多少数据值进行编码有关的信息的方法,在这种情况下,不需要传递对应的选择;然而,当多个最优选择可用时,编码器50选择方法并将与最适当选择有关的对应决策传递给解码器60。

[0227] 从上文可理解,当数据D1和数据D3例如如图3A所示彼此基本上相似时,解码器60基本上采用编码器50中所执行的编码函数的反函数。然而,如图3B所示,很多实际情况(例如在将编码数据E2组播到多个彼此不同的装置时),需要使用转码器70来对数据D3进行转码以生成对应的转码数据D4,转码数据D4与包含有解码器60及其相关的转码器70的给定装置兼容。可选地,解码器60和转码器70均通过计算硬件实现;可选地,转码器70实现为专用转码硬件,例如硬件解密器或类似装置。本公开的实施方式适于配置成提供对数据无损的或有损的编码和解码。可选地,解码器60还可操作为执行转码,例如将数据提供至与渲染数据(D1)所需的装置不同的显示装置。在这种情况下,通过编解码器100处理的数据无法解码回其原始形式。相反地,编码数据E2例如直接转换成其他一些格式,在这种格式下数据例如被渲染到屏幕或存储至文件。在这种转码的示例性实施方式中,数据D1初始为YUV格式,然后其被压缩并传输至接收器;接收器逐块地恢复数据,并对其执行颜色转换并将其缩放为适于渲染至屏幕上的RGB图像,而甚至无需重建全解析度的YUV结果图像。

[0228] 解码器60可操作为采用对由编码器50生成的编码数据(E2)进行解码以生成对应的解码数据D3的方法,其中该方法包括以下步骤:

[0229] (i) 接收编码数据(E2)并且从编码数据E2提取一个或多个索引集,以及一个或多

个频率表和/或一个或多个代码表和/或一个或多个代码字长度表和/或一个或多个概率表和/或指示这些一个或多个表的信息；

[0230] (ii) 从一个或多个索引集计算：一个或多个数据块中的对应的符号，和/或一个或多个代码表和/或一个或多个频率表和/或一个或多个代码字长度表和/或一个或多个概率表中的条目的经压缩的符号；以及 (iii) 利用来自一个或多个代码表和/或一个或多个频率表和/或一个或多个代码字长度表和/或一个或多个概率表的信息，从符号重新生成一个或多个数据块；以及

[0231] (iv) 对一个或多个数据块进行组合和/或变换，以生成解码数据 (D3)。

[0232] 可选地，该方法包括以一个或多个表中的至少一个可存储以用于后续再使用的方式，接收一个或多个表中的至少一个。

[0233] 可选地，该方法包括将一个或多个数据解压算法应用在步骤 (iv) 中，以生成解码数据 (D3)。更可选地，在该方法中，一个或多个数据解压算法包括以下中至少一种：哈夫曼解码、VLC解码、熵解码、算术解码、区间解码。

[0234] 可选地，该方法包括通过使用以基本上并发的方式处理多个数据块的并行架构处理器，将一个或多个数据块中的多个进行组合以生成解码数据 (D3)。

[0235] 可选地，该方法包括根据组合在一起的多个数据值，生成一个或多个索引集。更可选地，在该方法中，索引来源于包含R、G和B像素值或Y、U和V像素值的一个或多个RGB像素。更可选地，该方法包括根据数据块被包含在编码数据 (E2) 中的可实现的数据压缩比，在将未编码的数据块集合到编码数据 (E2) 与将编码的数据块集合到编码数据 (E2) 之间动态地切换。

[0236] 可选地，在该方法中，解码器60可操作为从编码数据 (E2) 提取指示符号是否属于“代码表的更改”或“数据结束”的至少一个末尾比特。

[0237] 可选地，该方法包括基本上从用于表示存在于给定的数据块中的一个或多个符号所需的足够索引来生成给定的数据块。

[0238] 可选地，该方法包括对包含在编码数据 (E2) 中的一个或多个代码表进行解压缩。更可选地，该方法包括通过使用哈夫曼解码对一个或多个代码表进行解压缩。更可选地，在该方法中，一个或多个代码表的解压缩使用一个或多个附属代码表。

[0239] 可选地，该方法包括：以使得一个或多个代码表能够在解码器 (60) 中用于对随后发送的数据进行解码的方式来接收一个或多个代码表。

[0240] 可选地，该方法包括从编码数据 (E2) 提取指示从何处一个或多个代码表易于通过一个或多个数据库和/或一个或多个代理数据库而访问的一个或多个识别代码。

[0241] 可选地，该方法包括对以下类型的数据中的一个或多个进行解码：捕获的音频信号、捕获的视频信号、捕获的图像、文本数据、地震数据、传感器信号数据、模拟数字 (ADC) 转换数据、生物信号数据、日历数据、经济数据、数学数据、二进制数据。

[0242] 可选地，该方法包括从多个数据源接收编码数据 (E2)，并且将由源提供的数据进行组合以重新生成编码数据 (E2)。

[0243] 解码器60可操作为实现前述的对编码数据 (E2) 进行解码以生成解码数据 (D3) 的方法；提供了用于对由编码器50生成的编码数据 (E2) 进行解码的解码器60，其中解码器60可操作为：

[0244] (i) 接收编码数据 (E2) 并且从编码数据 (E2) 提取一个或多个索引集, 以及一个或多个频率表和/或一个或多个代码表和/或一个或多个代码字长度表和/或一个或多个概率表和/或指示这些一个或多个表的信息;

[0245] (ii) 从一个或多个索引集计算一个或多个数据块中的对应的符号, 和/或一个或多个代码表和/或一个或多个频率表和/或一个或多个代码字长度表和/或一个或多个概率表中的条目的经压缩的符号; 以及

[0246] (iii) 利用来自一个或多个代码表和/或一个或多个频率表和/或一个或多个代码字长度表和/或一个或多个概率表的信息, 从符号重新生成一个或多个数据块; 以及

[0247] (iv) 对一个或多个数据块进行组合和/或变换, 以生成解码数据 (D3)。

[0248] 可选地, 解码器 60 还包括转码器 70, 转码器 70 用于对解码数据 (D3) 进行转码以生成对应的转码数据 (D4), 和/或从编码数据 (E2) 生成对应的转码数据 (D4)。

[0249] 可选地, 解码器 60 可操作为以使得一个或多个表中的至少一个可存储以用于后续再使用的方式, 来接收一个或多个表中的至少一个。

[0250] 可选地, 解码器 60 可操作为将一个或多个数据解压算法应用在步骤 (iv) 中以生成解码数据 (D3)。还可选地, 在解码器 60 中, 一个或多个数据解压算法包括以下至少一种: 哈夫曼解码、VLC 解码、熵解码、算术解码、区间解码。

[0251] 可选地, 解码器 60 可操作为通过使用以基本上并发的方式处理多个数据块的并行架构处理器, 将一个或多个数据块中的多个进行组合以生成解码数据 (D3)。

[0252] 可选地, 解码器 60 可操作为根据组合在一起的多个数据值生成一个或多个索引集。更可选地, 在解码器 60 中, 索引来源于包含 R、G 和 B 像素值或 Y、U 和 V 像素值的一个或多个 RGB 像素。更可选地, 解码器 60 操作为根据数据块包含在编码数据 (E2) 时可实现的数据压缩比, 在将未编码的数据块集合到编码数据 (E2) 与将编码的数据块集合到编码数据 (E2) 之间动态地切换。

[0253] 可选地, 解码器 60 可操作为从编码数据 (E2) 提取指示符号是否属于“代码表的更改”或“数据结束”的至少一个末尾比特。

[0254] 可选地, 解码器 60 可操作为基本上从用于表示存在于给定的数据块中的一个或多个符号所需的足够索引来生成给定的数据块。

[0255] 可选地, 解码器 60 可操作为对包含在编码数据 (E2) 中的一个或多个代码表进行解压缩。更可选地, 解码器 60 可操作为通过使用哈夫曼解码对一个或多个代码表进行解压缩。更可选地, 在解码器 60 中, 一个或多个代码表的解压缩使用一个或多个附属代码表。

[0256] 可选地, 解码器 60 可操作为以使得一个或多个代码表能够在解码器 (60) 中用于对随后发送的数据进行解码的方式来接收一个或多个代码表。

[0257] 可选地, 解码器 60 可操作为从编码数据 (E2) 提取指示从何处一个或多个代码表易于通过一个或多个数据库和/或一个或多个代理数据库而访问的一个或多个识别代码。

[0258] 可选地, 解码器 60 可操作为对以下类型的数据中的一个或多个进行解码: 捕获的音频信号、捕获的视频信号、捕获的图像、文本数据、地震数据、传感器信号数据、模拟数字 (ADC) 转换数据、生物信号数据、日历数据、经济数据、数学数据、二进制数据。

[0259] 可选地, 解码器 60 可操作为从多个数据源接收编码数据 (E2), 并且将由源提供的数据进行组合以重新生成编码数据 (E2)。例如, 多个源包含在将编码数据 (E2) 从编码器 50



传输至解码器60的点对点数据通信网络中。

[0260] 在不脱离如所附权利要求所限定的本发明的范围的情况下,可以对上文所描述的本发明的实施方式进行修改。用于描述并主张本发明的诸如“包括(including)”、“包括(comprising)”、“并入(incorporating)”、“由……组成(consisting of)”、“具有(have)”、“是(is)”的表达旨在以非排他性的方式进行解释,即也允许存在未明确地描述的项目、部件或元件。对单数的指示也应解释为涉及复数。所附权利要求中包含在括号中的数字旨在帮助对权利要求的理解并不应解释为以任何方式限制这些权利要求所主张的主题。

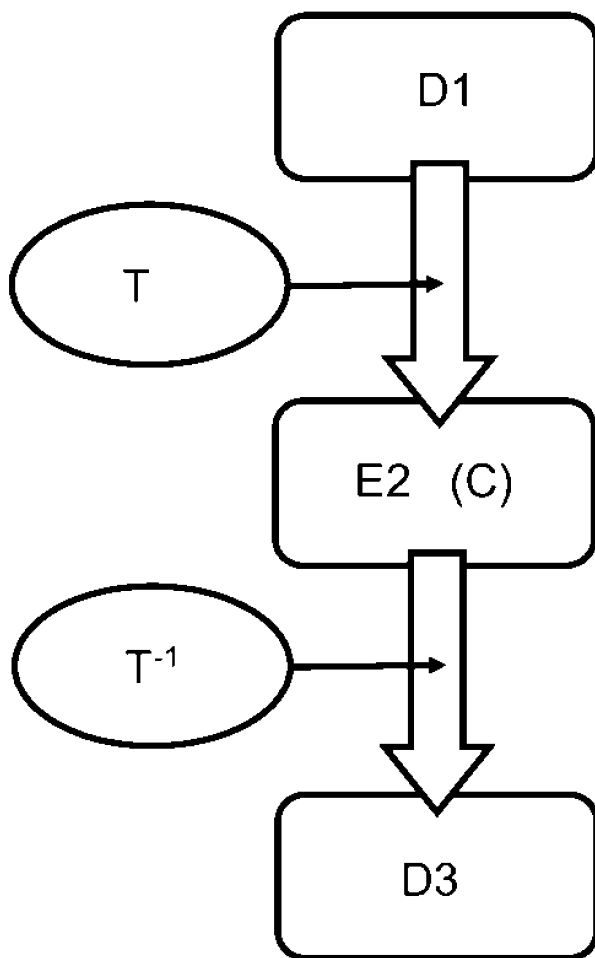


图1

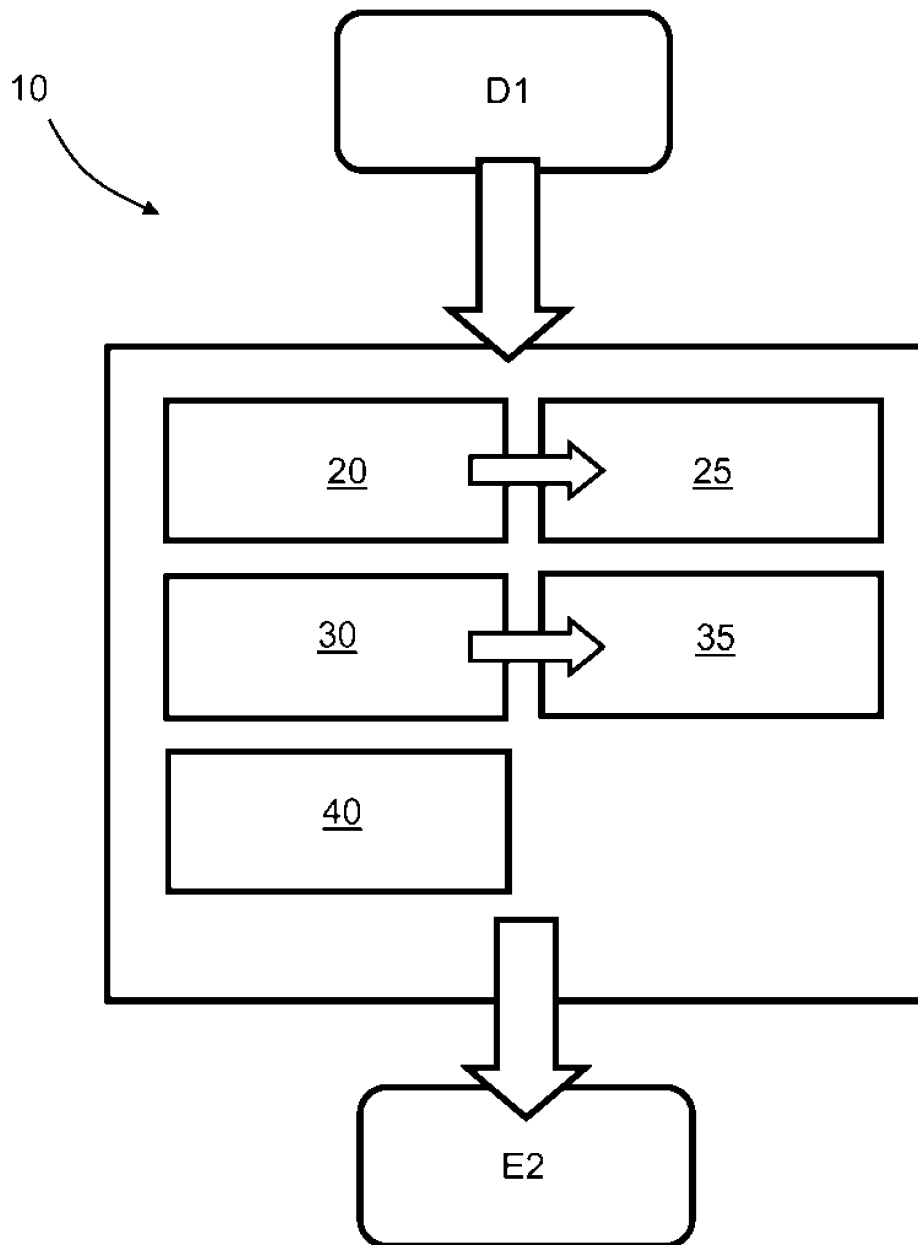


图2

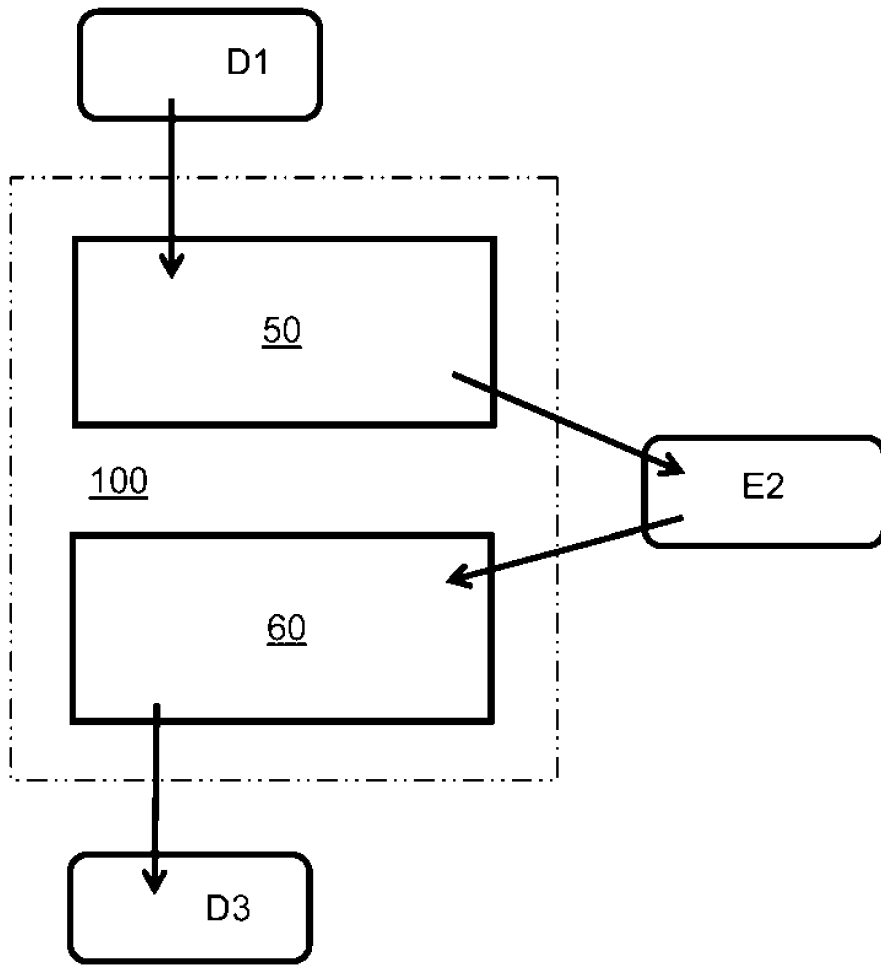


图3A

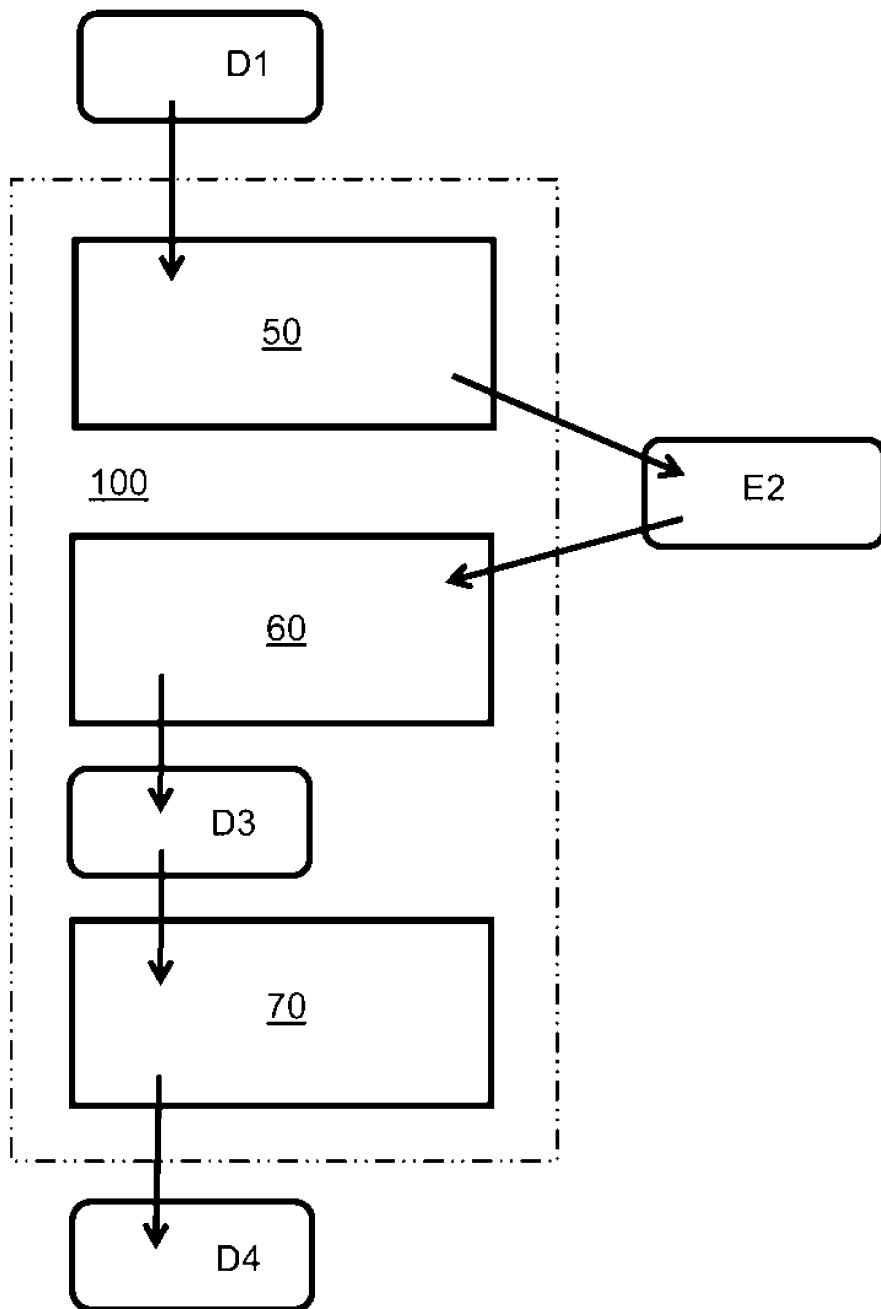


图3B