

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.

G06T 9/00 (2006.01)

H04N 7/30 (2006.01)



[12] 发明专利说明书

专利号 ZL 200480038002.4

[45] 授权公告日 2009年6月24日

[11] 授权公告号 CN 100504926C

[22] 申请日 2004.12.17

[21] 申请号 200480038002.4

[30] 优先权

[32] 2003.12.19 [33] SE [31] 0303497-2

[32] 2004.7.8 [33] SE [31] 0401850-3

[86] 国际申请 PCT/SE2004/001920 2004.12.17

[87] 国际公布 WO2005/059837 英 2005.6.30

[85] 进入国家阶段日期 2006.6.19

[73] 专利权人 艾利森电话股份有限公司

地址 瑞典斯德哥尔摩

[72] 发明人 J·斯特罗姆

[56] 参考文献

US2003/0090709A1 2003.5.15

US2003/0053706A1 2003.3.20

US6658146B1 2003.12.2

CN1452388A 2003.10.29

US6404923B1 2002.6.11

US5956431A 1999.9.21

审查员 王薇洁

[74] 专利代理机构 中国专利代理(香港)有限公司

代理人 程天正 张志醒

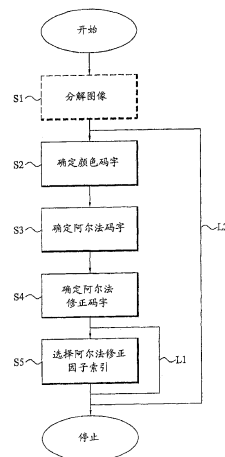
权利要求书6页 说明书38页 附图15页

[54] 发明名称

阿尔法图像处理

[57] 摘要

公开了阿尔法图像编码和解码方案。在编码中，阿尔法图像被分解成包含多个图像元素(610)的图像块(600)。各块(600)被压缩成块表示(700)。块表示(700)包含至少一个颜色码字(710)、阿尔法码字(720)、阿尔法修正码字(730)和阿尔法修正因子索引序列(740)。颜色码字(710)和阿尔法码字(720)分别表示块(600)的图像元素(610)的颜色和阿尔法值。阿尔法修正码字(730)表示一组多个阿尔法修正因子，用于修正由阿尔法码字(720)表示的阿尔法值。索引序列(740)包括用于块(600)中的每个图像元素(610)的阿尔法索引，其中阿尔法索引识别阿尔法修正因子组中的阿尔法修正因子之一。



1. 一种压缩包含多个图像元素(610)的图像块(600)的方法, 所述方法包括步骤:

-确定颜色码字(710), 其表示所述多个图像元素(610)的颜色;

-确定阿尔法码字(720), 其表示所述多个图像元素(610)的阿尔法值;

-提供阿尔法修正码字(730), 其表示一组包含多个阿尔法修正因子的阿尔法修正因子组, 用于修正基于所述阿尔法码字(720)生成的阿尔法值; 以及

-为所述图像块(600)中的每个图像元素(610)选择与来自所述阿尔法修正因子组的一阿尔法修正因子相关的一个阿尔法修正因子索引。

2. 根据权利要求1的方法, 还包括步骤:

-提供颜色修正码字(750), 用于表示一组包含多个颜色修正因子的颜色修正因子组, 用于修正基于所述颜色码字(710)生成的颜色; 以及

-为所述图像块(600)中的每个图像元素(610)选择一个与来自所述颜色修正因子组的一颜色修正因子相关的颜色修正因子索引。

3. 根据权利要求2的方法, 其中所述颜色修正码字提供步骤包括: 从包含多个颜色修正因子组的颜色表格(550)选择所述颜色修正因子组, 藉此所述颜色修正码字(750)使得能够识别来自所述颜色表格(550)的所述被选择的颜色修正因子组。

4. 根据权利要求1至3的任何一项的方法, 其中所述颜色码字确定步骤包括步骤:

-确定第一颜色子码字(710A), 其表示所述多个图像元素(610)的至少第一部分的颜色; 并且

-确定第二颜色子码字(710B), 其表示所述多个图像元素(610)的至少第二部分的颜色, 所述方法还包括步骤:

-为所述多个图像元素(610)的至少一个子组中的每个图像元素(610)选择与所述第一颜色子码字(710A)或所述第二颜色子码字(710B)相关的颜色索引。

5. 根据权利要求4的方法, 其中所述颜色索引的一个序列(770)包括用于所述多个图像元素(610)第一子组中的每个图像元素(610)

的一个颜色索引，该颜色索引与所述第一颜色子码字（710A）或第二颜色子码字（710B）相关，所述多个图像元素（610）的第二剩余子组中的每个图像元素（610）与从所述第一颜色子码字（710A）或第二颜色子码字（710B）选择的预定义颜色子码字相关。

6. 根据权利要求1至3的任何一项的方法，其中所述阿尔法修正码字提供步骤包括：从包括多个阿尔法修正因子组的阿尔法表格（500）选择所述阿尔法修正因子组，藉此所述阿尔法修正码字（730）使得能够识别来自所述阿尔法表格（500）的所述被选择的阿尔法修正因子组。

7. 一种编码图像的方法，所述方法包括步骤：

-分解所述图像为多个图像块（600），每个图像块（600）包含多个图像元素（610）；并且

-按照权利要求1至6任何一项所述的方法压缩至少一个图像块（600），为所述至少一个图像块（600）确定压缩表示（700）。

8. 一种处理包含多个图像元素（610）的图像块（600）的压缩表示（700）的方法，所述压缩表示（700）包含颜色码字（710）、阿尔法码字（720）、阿尔法修正码字（730）和阿尔法修正因子索引序列（740），并且所述方法包括步骤：

-基于所述阿尔法修正码字（730）提供一组包含多个阿尔法修正因子的阿尔法修正因子组；

对于所述图像块（600）中的至少一个图像元素（610）：

-基于所述颜色码字（710）生成颜色表示；

-基于所述阿尔法码字（720）生成阿尔法表示；

-基于所述阿尔法修正因子索引序列（740）从所述阿尔法修正因子组中选择阿尔法修正因子；以及

-基于所选择的阿尔法修正因子修正所述阿尔法表示。

9. 根据权利要求8的方法，其中所述压缩表示（700）还包含颜色修正码字（750）和颜色修正因子索引序列（760），所述方法包括进一步的步骤：

-基于所述颜色修正码字（750）提供一组包含多个颜色修正因子的颜色修正因子组；

-基于所述颜色修正因子索引序列（760）从所述颜色修正因子组中选择颜色修正因子；以及

-基于所选择的颜色修正因子修正所述颜色表示。

10. 根据权利要求9的方法，其中所述提供所述颜色修正因子组的步骤包括：基于所述颜色修正码字（750），从包含多个颜色修正因子组的颜色表格（550）中选择所述颜色修正因子组。

11. 根据权利要求8至10的任何一项的方法，其中所述颜色码字（710）包括一第一颜色子码字（710A）和一第二颜色子码字（710B），并且所述压缩表示（700）还包括颜色索引序列（770），并且所述颜色表示生成步骤包括步骤：

-基于从所述第一颜色子码字（710A）和第二颜色子码字（710B）选择的至少一个颜色子码字，生成所述颜色表示。

12. 根据权利要求11的方法，其中所述颜色索引序列（770）包括用于所述多个图像元素（610）第一个子组中的每一个图像元素（610）的一个颜色索引，该颜色索引与所述第一颜色子码字（710A）或第二颜色子码字（710B）相关，在所述多个图像元素（610）的第二剩余子组中的每个图像元素（610）与从所述第一颜色子码字（710A）或第二颜色子码字（710B）选择的预定义颜色子码字有关。

13. 根据权利要求8至10的任何一项的方法，其中所述提供所述阿尔法修正因子组的步骤包括：基于所述阿尔法修正码字（730），从包含多个阿尔法修正因子组的阿尔法表格（500）中选择所述阿尔法修正因子组。

14. 一种解码被编码图像的方法，所述被编码图像包含图像块（600）的压缩表示（700），该图像块（600）包含多个图像元素（610），压缩表示（700）包括颜色码字（710）、阿尔法码字（720）、阿尔法修正码字（730）和阿尔法修正因子索引序列（740），并且所述方法包括步骤：

-按照权利要求8至13任何一项所述的方法来处理所述至少一个压缩表示（700），为至少一个压缩表示（700）确定至少一个解压缩图像元素表示；以及

-通过处理所述至少一个解压缩图像元素表示生成图像。

15. 一种压缩包含多个图像元素（610）的图像块（600）的系统（300），所述系统（300）包括：

-颜色量化器（310），用于确定表示所述多个图像元素（610）的颜色的颜色码字（710）；

-阿尔法量化器(320),用于确定表示所述多个图像元素(610)的阿尔法值的阿尔法码字(720);

-用于提供表示一组包含多个阿尔法修正因子的阿尔法修正因子组的阿尔法修正码字(730)的装置(330),以便修正基于所述阿尔法码字(720)生成的阿尔法值;以及

-索引选择器(340),用于为所述图像块(600)中的每个图像元素(610)选择与来自所述阿尔法修正因子组的一阿尔法修正因子相关的一个阿尔法修正因子索引。

16. 根据权利要求15的系统,还包括:

-用于提供表示一组包含多个颜色修正因子的颜色修正因子组的颜色修正码字(750)的装置(360),以便修正基于所述颜色码字(710)生成的颜色,其中所述索引选择器(340)被配置为所述图像块(600)中的每个图像元素(610)选择一个与来自所述颜色修正因子组的一颜色修正因子相关的颜色修正因子索引。

17. 根据权利要求16的系统,其中所述颜色修正码字提供装置(360)被配置用于从包含多个颜色修正因子组的颜色表格(550)选择所述颜色修正因子组,藉此所述颜色修正码字(750)使得能够识别来自所述颜色表格(550)的所述被选择的颜色修正因子组。

18. 根据权利要求15至17的任何一项的系统,其中所述颜色量化器(310)被配置用于确定表示所述多个图像元素(610)至少第一部分的颜色的第一颜色子码字(710A)和确定表示所述多个图像元素(610)至少第二部分的颜色的第二颜色子码字(710B),并且所述索引选择器(340)被配置用于为所述多个图像元素(610)的至少一个子组中的每个图像元素(610)选择与所述第一颜色子码字(710A)或所述第二颜色子码字(710B)相关的颜色索引。

19. 根据权利要求18的系统,其中所述颜色索引的一个序列(770)包括用于所述多个图像元素(610)第一子组中的每个图像元素(610)的一个颜色索引,该颜色索引与所述第一颜色子码字(710A)或第二颜色子码字(710B)相关,所述多个图像元素(610)的第二剩余子组中的每个图像元素(610)与从所述第一颜色子码字(710A)或第二颜色子码字(710B)选择的预定义颜色子码字相关。

20. 根据权利要求15至17的任何一项的系统,其中阿尔法修正码

字提供装置(330)被配置来从包括多个阿尔法修正因子组的阿尔法表格(500)选择所述阿尔法修正因子组,藉此所述阿尔法修正码字(730)使得能够识别来自所述阿尔法表格(500)的所述被选择的阿尔法修正因子组。

21. 一种图像编码系统(210),包括:

-图像分解(215),用于分解图像为多个图像块(600),每个图像块(600)包含多个图像元素(610);以及

-至少一个按照权利要求15至20任何一项的图像块压缩系统(300)。

22. 一种用于处理包含多个图像元素(610)的图像块(600)的压缩表示(700)的系统(400),所述压缩表示(700)包含颜色码字(710)、阿尔法码字(720)、阿尔法修正码字(730)和阿尔法修正因子索引序列(740),并且所述系统(400)包括:

-基于所述阿尔法修正码字(730)而提供一组包含多个阿尔法修正因子的阿尔法修正因子组的装置(430);

-颜色生成器(410),其基于所述颜色码字(710)、为所述图像块(600)中的至少一个图像元素(610)生成颜色表示;

-阿尔法生成器(420),其基于所述阿尔法码字(720)为所述至少一个图像元素生成阿尔法值;

-选择器(440),其基于所述阿尔法修正因子索引序列(740)从所述阿尔法修正因子组中为所述至少一个图像元素(610)选择阿尔法修正因子;以及

-阿尔法修正器(450),其基于所选择的阿尔法修正因子修正所述阿尔法值。

23. 根据权利要求22的系统,其中所述压缩表示(700)还包含颜色修正码字(750)和颜色修正因子索引序列(760),所述系统(400)还包括:

-基于所述颜色修正码字(750)而提供一组包含多个颜色修正因子的颜色修正因子组的装置(460);以及

-颜色修正器(470),其基于使用所述颜色修正因子索引序列(760)从所述颜色修正因子组中由所述选择器(440)选择的颜色修正因子,修正所述颜色表示。

24. 根据权利要求23的系统,其中所述颜色修正因子组提供装置

(460) 被配置来基于所述颜色修正码字 (750)，从包含多个颜色修正因子组的颜色表格 (550) 中选择所述颜色修正因子组。

25. 根据权利要求 22 至 24 的任何一项的系统，其中所述颜色码字 (710) 包括一第一颜色子码字 (710A) 和一第二颜色子码字 (710B)，并且所述压缩表示 (700) 还包括颜色索引序列 (770)，并且所述颜色生成器 (410) 被配置来基于从所述第一颜色子码字 (710A) 和第二颜色子码字 (710B) 由所述选择器 (440) 选择的至少一个颜色子码字，生成所述颜色表示。

26. 根据权利要求 25 的系统，其中所述颜色索引序列 (770) 包括用于所述的多个图像元素 (610) 第一子组中的每个图像元素 (610) 的一个颜色索引，该颜色索引与所述第一颜色子码字 (710A) 或第二颜色子码字 (710B) 相关，在所述多个图像元素 (610) 的第二剩余子组中的每个图像元素 (610) 与从所述第一颜色子码字 (710A) 或第二颜色子码字 (710B) 选择的预定义颜色子码字有关。

27. 根据权利要求 22 至 24 的任何一项的系统，其中所述阿尔法修正因子组提供装置 (430) 被配置来基于所述阿尔法修正码字 (730)，从包含多个阿尔法修正因子组的阿尔法表格 (500) 中选择所述阿尔法修正因子组。

28. 一种用于解码被编码图像的系统 (220)，所述被编码图像包含图像块 (600) 的压缩表示 (700)，每个图像块 (600) 包含多个图像元素 (610)，压缩表示 (700) 包括颜色码字 (710)、阿尔法码字 (720)、阿尔法修正码字 (730) 和阿尔法修正因子索引序列 (740)，所述系统 (220) 包括：

-至少一个按照权利要求 22 至 27 任何一项的系统 (400)，为至少一个压缩表示 (700) 确定至少一个解压缩图像元素表示；以及

-用于处理所述至少一个解压缩图像元素表示以便生成图像的装置 (224)。

29. 一种图像处理终端 (100)，包括按照权利要求 15 至 28 任何一项的系统 (210; 220; 300; 400)。

阿尔法图像处理

技术领域

本发明一般地涉及图像处理，并且特别地涉及用于编码和解码图像的方法和系统。

发明背景

在数据处理系统和用户终端比如计算机上并且特别是在移动终端上的图像和图形的呈现和再现，最近几年已经得到极大的提高。例如，在这样的终端上，三维（3D）图形和图像具有许多引人注目的应用，包括游戏、3D地图和消息传送、屏幕保护以及人机界面。

3D图形再现进程典型地包括三个子阶段。简言之，第一阶段，应用阶段，创建几个三角形。在第二阶段，几何阶段，这些三角形的角被转换、投影并且点亮。在第三阶段，光栅化阶段（rasterization stage），通常表示为纹理的图像可以粘合到三角形上，从而增加所再现的图像的真实感。第三阶段典型地还使用z缓冲器执行拣选（sort）。

然而，图像和纹理的再现，并且特别是3D图像和图形的再现，在图形系统所需的存储器带宽和处理功率方面，是一项在计算上昂贵的任务。例如，纹理在存储器方面和在存储器带宽方面都成本很高，在存储器方面，纹理必须被放置或高速缓存在快速片内存储器上，而在存储器带宽方面，纹理可以被存取数次以描画单个像素。

为了降低带宽和处理功率要求，典型地使用图像（纹理）编码方法或系统。这样的编码系统应当带来在再现期间更高效地使用昂贵的片内（on-chip）存储器和更低的存储器带宽，并且因此带来更低的功率消耗和/或更快的再现。

除了高效和准确地压缩和解压缩图像之外，还有一个一般的期望，就是纹理压缩和解压缩系统应当具有管理阿尔法（ α ）分量的能力。这样的阿尔法分量可以被用于表示纹理中的透明的或半透明的像素或纹理元（texel），其使用于广泛范围的应用中。

在由Iourcha等人的专利US 5,956,431的专利说明书中，公开了被称之为S3TC（S3纹理压缩）或DXTC（DirectX纹理压缩）的纹理压缩方案。图像被解压缩为多个4像素×4像素的图像块。每个这样的图像

块被编码成一个 64 比特的比特序列，因此带来了 4bpp（比特每像素）的压缩率。该 64 比特序列包含两个基本颜色或颜色码字以及一个像素索引序列，图像块中的每个像素对应一个索引。在解码期间，生成一个有四种颜色的调色板。调色板的头两个 RGB（红、绿和兰）颜色对应于两个基本颜色（码字）。于是位于 RGB 空间中的基本颜色之间的两种附加颜色从那里被内插。每个像素索引然后为每个像素识别出调色板的四种颜色之一来用于那个像素。

在调色板的这两种附加颜色的内插期间，执行乘以 $1/3$ 和 $2/3$ 的乘法，这在硬件上并不理想。而且，使用 S3TC 的压缩相对比较费时。

S3TC 方案将阿尔法纹理作为特殊情况处理。比特然后被不同地解释，以便考虑到所谓的击穿阿尔法（punch-through α ）。阿尔法机制的这一简单版本将每个像素表示成完全不透明或透明。S3TC 不能处理更准确的阿尔法这一事实是该系统的主要缺点之一。

发明内容

本发明克服了现有技术安排中的这些和其他的缺点。

本发明的一个一般的目的是来提供高效的图像处理。

本发明的另一个目的是来提供高效的图像编码和图像解码。

本发明的又一个目的是来提供适合于包含透明和/或半透明图像元素的阿尔法图像的图像编码和解码。

这些和其他的目的由所附的专利权利要求所定义的发明来达到。

简言之，本发明包含以编码（压缩）阿尔法图像和解码（解压缩）被编码的（被压缩的）图像形式的阿尔法图像处理。

根据本发明，将被编码的阿尔法图像被分解成多个图像块，图像块包含多个图像元素（像素、纹理元素、纹理元、或体积元素、体元（voxel））。图像块优选地包含八个图像元素并且优选地具有大小为 $2^m \times 2^n$ 的图像元素，其中 $m = 3 - n$ 并且 $n = 0, 1, 2, 3$ ，或者大小为 $2^m \times 2^n \times 2^p$ 的图像元素，其中 $m, n, p = 0, 1, 2, 3$ ，并且优选地 $m + n + p = 3$ 。块中的每个图像元素由一个颜色和阿尔法或透明度值来表征。各个图像块然后被压缩。

在这个有损块编码中，为该图像块确定一个颜色码字。该颜色码字表示图像块的图像元素的颜色。而且，为图像块确定一个阿尔法码字。这个阿尔法码字表示块中的图像元素的阿尔法值。优选的表示是块中的

图像元素的量化的平均阿尔法值。

然后，为该图像块提供一个阿尔法修正码字。这个阿尔法修正码字表示一组多个阿尔法修正因子，（在解码期间）用于修正或调制由阿尔法码字表示的阿尔法值。一旦提供了阿尔法修正码字，为图像块中的图像元素选择阿尔法修正因子索引。每个这样的修正因子索引与来自阿尔法修正因子组的一个阿尔法修正因子有关。

在本发明的一个优选的实施例中，图像块可以根据多个、优选为两个、可能的压缩模式之一被压缩。通过能够更好地适合于阿尔法图像中的各个图像块的特性，两个不同压缩模式的使用为编码方法提供了灵活性。作为具有在每个块基础上在两个不同压缩模式之间的选择的结果，与具有单个模式编码的已有技术阿尔法适配的编码方案相比，被处理图像的图像质量得到了提高。

在第一压缩模式下，生成一个颜色修正码字。这个颜色修正码字表示（在解码期间）用于修正或调制块中图像元素的颜色的一组多个强度或颜色修正因子。一旦提供了颜色修正码字，则为图像块中的图像元素选择颜色修正因子索引。每个这样的修正因子索引与来自阿尔法修正因子组的一个阿尔法修正因子有关。

在第二压缩模式下，颜色码字包含第一颜色子码字和第二颜色子码字。第一颜色子码字优选地表示图像块的第一部分中的图像元素的颜色，而第二子码字表示该块的第二部分中的图像元素的颜色。而且，优选地为该块中的图像元素的第一子组生成与第一或第二颜色子码字有关的颜色索引。该块中的图像元素的剩余的第二个子组，优选地与颜色子码字之一具有预定义的关系，这样不需要颜色索引用于这个剩余子组的图像元素。

尽管按照这两个模式压缩的图像块表示包括不同的码字和与图像元素有关的索引，在比特数量方面，被压缩的块表示的大小优选地在两个模式下相等。

在确定码字和索引序列之前，优选地在块中图像元素特性的研究的基础上，例如在颜色空间中图像元素的颜色值分布的基础上，可以执行用于图像块的压缩模式的选择。可选地，该模式选择在生成码字和索引序列期间或之后做出。在后一种情况中，将产生图像块的两个压缩的表示，一个是根据第一压缩模式并且一个是根据第二压缩模式。与根据第一模

式压缩图像块有关的第一误差值使用根据该第一模式压缩的块表示来确定。相应地，与根据第二模式压缩图像块有关的第二误差值使用根据该第二模式压缩的块表示来确定。然后比较这两个误差值，并且选择带来最高质量（最小误差值）的压缩模式。根据这个所选模式压缩的块表示将被用作当前块的编码表示。

在解码期间，应当被解压缩的压缩图像块被识别并且从例如存储器位置中取装。一旦识别出正确的压缩图像块，使用码字和索引生成所识别的压缩图像块中的至少一个图像元素的解码表示。

本发明提供下面的优势：

- 由于允许阿尔法值表示的、除完全透明度和不透明性、半透明度之外的灵活性，为阿尔法图像给出了高质量（峰值信号/噪声比）；

- 解码的硬件实现非常简单；

- 编码很快，使之也能够实现在具有低时钟频率的终端中；

- 可以被扩充为根据适应于不同图像块的两种模式来工作，这将进一步提高峰值信号/噪声比；并且

- 在计算机上，可能在可行的速率下进行穷尽编码。

本发明提供的其他优势将在阅读本发明实施例的下述描述的基础上得到了解。

附图简述

通过参照下面结合附图的描述，本发明及其进一步的目标和优势可以得到最佳理解，其中：

图 1 示例了根据本发明的图像编码方法实施例的流程图；

图 2A 和 2B 示例了根据本发明的图像块的实施例；

图 3A 至 3C 示例了图像块的压缩表示的实施例；

图 4 是一个流程图，示例了图 1 的图像编码方法的颜色码字确定步骤的实施例；

图 5 是一个流程图，示例了图 1 的图像编码方法的颜色码字确定步骤的另一个实施例；

图 6 是一个流程图，更详细地示例了图 1 的确定阿尔法码字、确定阿尔法修正码字以及选择阿尔法修正因子索引的步骤；

图 7 示例了根据本发明的图像解码方法的流程图；

图 8 更详细地示例了图 7 的颜色表示生成步骤的实施例的流程图；

图 9 更详细地示例了图 7 的颜色表示生成步骤的另一个实施例的流程图;

图 10 更详细地示例了图 7 的生成阿尔法表示、选择阿尔法修正因子以及修正阿尔法表示的步骤的流程图;

图 11 示意性示例了根据本发明的具有图像编码器和解码器的用户终端的例子;

图 12 为示意性示例了根据本发明的图像编码器的实施例的框图;

图 13 为示意性示例了根据本发明的块编码器的实施例的框图;

图 14 为示意性示例了根据本发明的块编码器的另一个实施例的框图;

图 15 为更详细地示意性示例了图 14 或 15 的块编码器的颜色量化器的框图;

图 16 为更详细地示意性示例了图 14 或 15 的块编码器的阿尔法量化器的框图;

图 17 为更详细地示意性示例了图 14 或 15 的块编码器的索引选择器的框图;

图 18 为示意性示例了根据本发明的图像解码器的实施例的框图;

图 19 为示意性示例了根据本发明的块解码器的实施例的框图;

图 20 为示意性示例了根据本发明的块解码器的另一个实施例的框图;

图 21 为示意性示例了根据本发明的块解码器的实施例的硬件框图;

图 22 为更详细地示例了图 21 的比特扩充器的实施例的硬件框图;

图 23 为更详细地示例了图 21 的箝位器的实施例的硬件框图; 以及

图 24 为更详细地示例了图 21 的表格查找的实施例的硬件框图。

具体实施例

在所有的附图中, 同样的参考字符被用于相应的或同样的部件。

本发明涉及图像和图形处理, 并且特别地涉及编码或压缩阿尔法图像和解码或解压缩被编码的(被压缩的)阿尔法图像。

一般地, 根据本发明, 在图像编码期间, 阿尔法图像被分解或分割成多个图像块。每个这样的图像块然后包含多个即至少两个图像元素, 图像元素具有与图像元素相关的特性, 尤其是一定的颜色和相关的阿尔法或透明度值。图像块然后被编码或压缩来生成图像的编码表示。

当编码图像要显示在屏幕上时，或与编码图像相关的几何图元将被再现时，编码的图像块的有关图像元素被识别并且解码。这些被解码的图像元素然后可以被用于生成原始图像的解码表示用于显示，或者它们可以被用于再现几何图元。

本发明也很适合于供三维（3D）图形和图像，比如照片、文本和“合成”图像使用，所有的这些都可以用于应用中，比如游戏、3D地图和场景、3D消息例如动画消息、屏幕保护、人机界面（MMI）等等，但并不局限于此。因此，本发明也可以应用于编码其他类型的图像或图形，例如一维（1D）、二维（2D）或3D图像。

在本发明中，表达法“图像元素”指的是图像块或图像块的编码（压缩）表示中的元素。这个图像块，反过来，对应于图像或纹理的一部分。

因此，根据本发明，图像元素可以是（1D、2D或3D）纹理的纹理元或（1D、2D或3D）图像的像素。相应地，图像元素可以是3D纹理或图像中的体元。通常地，图像元素由一定的图像元素相关特性比如颜色值和阿尔法或透明度值来表征。而且，下面，术语“图像”被用于表示可以根据本发明被编码或解码的任何1D、2D或3D图像或纹理。

如上面简单提及的，图像元素经常尤其由颜色来表征。这一颜色可以是单一分量（一维）特性，典型地为灰度级或值。在这样的情况中，图像元素的颜色可以取从最小灰度级（例如0，典型地表示黑色）到最大的灰度级（例如255，典型地表示白色）的一个值。可选地，图像元素的颜色可以是多分量（多维）特性，典型地包含三个颜色分量。在这样的情况中，颜色可以是RGB（红、绿、兰）颜色、YUV空间中的颜色、YCrCb空间中的颜色、或者任何其他的用于图像和图形处理的颜色空间。下面，本发明将主要结合RGB颜色进行描述。然而，这应当仅被视作合适的颜色格式的示例性而非限制性的例子。相应的描述因此也可以应用到其他的多分量颜色格式和单一分量颜色格式，例如灰度级。

在阿尔法图像处理领域，阿尔法值用于表示图像元素的透明度特性。通常，阿尔法值典型地被定义为阿尔法-值 = 1 - 透明度，如果 $0 \leq \text{阿尔法-值} \leq 1$ 并且 $0 \leq \text{透明度} \leq 1$ 。因此，阿尔法值为1在该定义中对应无透明度或非透明性（透明度为0），而阿尔法值为0那么将表示全透明度（透明度为1）。尽管这个阿尔法值定义是该领域中最常用的，本发明并不局限于此。实际上，图像元素的阿尔法值根据本发明因此可以表示该图像

元素的任何透明度特性，这取决于所使用的阿尔法值透明度关系。

本发明预见到“阿尔法图像”的图像元素除它们各自的阿尔法值外也具有相关的颜色值。因此，本发明的阿尔法图像可以是所谓的 RGBA-图像。

图像编码

图 1 为根据本发明的、编码阿尔法图像的（有损）方法的实施例的流程图。在第一步骤 S1 中，图像被分解或分割成多个图像块。每个这样的图像块然后包含多个图像元素。在本发明的优选实施例中，一个图像块包含八个图像元素（像素或纹理元）并且具有大小为 $2^m \times 2^n$ 的图像元素，其中 $m = 3 - n$ 并且 $n = 0, 1, 2, 3$ 。更为优选地， n 为 1 或 2。图 2A 和 2B 示意性示例了根据本发明的具有八个图像元素 610 的图像块 600 的两个例子。在图 2A 中，高度为两个图像元素 610 并且宽度为四个图像元素 610，即 $m = 1$ 并且 $n = 2$ ，而对于图 2B 中的图像块 600， $m = 2$ 并且 $n = 1$ 。相应地，当压缩 3D 图像时，优选的图像块大小可以是 $2 \times 2 \times 2$ 图像元素（体元）。然而，本发明并不局限于具有八个图像元素的块，而是可以可选地结合具有小于例如 2×2 或多余八个图像元素例如 4×4 图像元素的图像块使用。

返回图 1，整个图像块在步骤 S1 中优选地被分解为（非重叠的）图像块。然而，在一些应用中，只有图像的一部分被编码并且因此只有这部分被分解为图像块。

在步骤 S2 中，为图像块确定颜色码字。这个颜色码字表示图像块中的图像元素的颜色。在典型的实现中，颜色码字表示块的图像元素的平均颜色，但是可以可选地表示多个颜色值。颜色码字优选地具有与原始图像同样的颜色格式（空间）。然而，在一些情况下，将图像转换为不同的颜色格式可能有用，即具有第一颜色空间中的颜色码字和具有第二颜色空间中的原始图像。在下一个步骤 S3 中，为当前的图像块确定阿尔法码字。这个阿尔法码字表示图像块中的图像元素的阿尔法值。在典型的实现中，阿尔法码字表示块中的图像元素的平均阿尔法值。

然后，在步骤 S4 中提供阿尔法修正码字。这个阿尔法修正码字表示用于（在解码期间）修正图像块中的图像元素的阿尔法表示的一组多个阿尔法修正因子。

在本发明的优选实施例中，阿尔法修正码字为允许识别一个阿尔法

修正因子组的索引或表示。这个索引然后可以在包含若干不同的阿尔法修正因子组的阿尔法表格或密码本中识别或指向该组。每组包含两个或多个阿尔法修正因子值，优选地为至少四个修正因子值。

阿尔法表格优选地包含包括小阿尔法修正因子值的组，该组适合于允许表示平滑地改变透明度特性。而且，该表格优选地也包含包括大阿尔法修正因子值的组，该组适合于允许表示强烈的透明度对比。

表格中的组的实际阿尔法修正因子值可以通过以随机值开始，然后使用多个不同的最佳化方案和算法来最佳化这些值而得到，所述算法比如是 LBG-算法(Linde、Buzo 和 Gray)[2]的版本，模拟退火(annealing)和协调搜索，这是本领域技术人员所已知的。少量的不同类型的阿尔法图像可以被用作训练数据。

为了使阿尔法表格的硬件实现不太昂贵，一组阿尔法修正因子被强制为算术上互补的值，即各组优选地为对称的，和/或给定组的阿尔法修正因子可以是另一组阿尔法修正因子的通过因子例如 2 修正的副本。

表 1 示例了包含 16 组阿尔法修正因子的阿尔法表格的非限制性例子，每组中具有四个修正因子值。

表 1

组	码字	阿尔法修正因子值			
0	0000 _{bin}	-8	-2	2	8
1	0001 _{bin}	-12	-4	4	12
2	0010 _{bin}	-31	-6	6	31
3	0011 _{bin}	-34	-12	12	34
4	0100 _{bin}	-50	-8	8	50
5	0101 _{bin}	-47	-19	19	47
6	0110 _{bin}	-80	-28	28	80
7	0111 _{bin}	-127	-42	42	127
8	1000 _{bin}	-16	-4	4	16
9	1001 _{bin}	-24	-8	8	24
10	1010 _{bin}	-62	-12	12	62
11	1011 _{bin}	-68	-24	24	68
12	1100 _{bin}	-100	-16	16	100
13	1101 _{bin}	-94	-38	38	94

14	1110_{bin}	-160	-56	56	160
15	1111_{bin}	-254	-84	84	254

在表 1 中，阿尔法修正因子组 8-15 为组 0-7 乘以因子 2 的副本。

如果阿尔法表格包含至多 16 个不同的阿尔法修正因子组，阿尔法码字优选地为一 4 比特索引 (0000_{bin} - 1111_{bin}) 识别该表格的 (八) 组之一，例如对于码字 0000_{bin} 为 $[-8, -2, 2, 8]$ 。由于仔细选择组中的修正因子值，整个表 1 可以使用仅 16 个修正因子值来重建并且剩余的 48 个值可以根据其来计算。

但本发明不局限于表 1 的使用，而是可以使用具有其他阿尔法修正因子组和其他表格。而且，对于一个表格中多于或少于 16 组的情况，阿尔法码字的大小可能必须被改变。例如，如果表格包含两个 (3-4、5-8 或多余 16) 阿尔法修正因子组，阿尔法码字大小可以被限制到一比特 (两比特、三比特或大于四比特)。而且，每组的阿尔法修正因子值的数量可以不同于四个，例如可使用每组五个值，给定 $[-8, -2, 0, 2, 8]$ 的例子。

表格中各组的阿尔法值可以使用几个不同类型的图像作为训练数据来确定，如上所讨论的。然而，如果只有一个特定图像类型将被编码，修正因子值可以使用对应于该图像类型的训练数据来确定，即给定专用于特定图像类型的阿尔法表格。也有可能使一个阿尔法表格具有适合于特定图像的强度修正因子值。在这些情况中，即专用于图像或图像类型的表格，可能需要在编码图像块的压缩文件中包括该表格的强度修正因子值，或使其相关。

而且，阿尔法码字不必是指向表格中的阿尔法修正因子组的一个指针，而实际上可以是阿尔法修正因子组本身，例如包括两个修正因子值，比如 2 和 8，并且其中其他的修正因子值，比如 -2 和 -8，可以从这两个值确定。可选地，所使用的阿尔法修正因子组可以是按照 $[-ka, -a, a, ka]$ 或 $[-k_2a, -k_1a, k_1a, k_2a]$ ，其中 a 为 0 或一个正整数并且 k 、 k_1 和 k_2 为乘法因子。在这样的情况中，阿尔法码字仅需要包括值 a 并且所有剩余的三个值可以从中计算，如果 k 或 k_1 和 k_2 为预定义的乘法因子的话。

一旦在步骤 S4 中提供了阿尔法码字，下一个步骤 S5 为图像块中的图像元素选择阿尔法修正因子索引或表示。每个这样的阿尔法修正因子索引与来自步骤 S4 中提供的阿尔法修正因子组的一个阿尔法修正因子值相关。换句话说，阿尔法修正因子索引允许识别该组的哪个阿尔法修正

因子用于块中的特定图像元素。

在阿尔法修正因子组包含四个修正因子值的情况中，比如-8、-2、2、8，阿尔法修正因子索引可以是一个识别这四个值之一的2比特序列。

步骤 S5 优选地为块中的所有图像元素重复执行，如图中的线 L1 所示意的。

步骤 S2 到 S5 的编码结果是编码图像块或更准确地为图像块的编码（压缩）表示。这样的编码块表示 700 如图 3A 所示。表示 700 包含颜色码字 710、阿尔法码字 720、阿尔法修正码字 730 和阿尔法修正因子索引序列或位图 740。应当看到，编码图像块 700 的颜色码字 710、阿尔法码字 720、阿尔法修正码字 730 和阿尔法修正因子索引序列 740 的相互间顺序可以与图中所示不同。也应当看到，码字和索引序列不必被连续布置。

步骤 S2 到 S5 优选地为在步骤 S1 的分解期间提供的所有图像块重复执行（如线 L2 所示意的）。然后，结果是得到压缩图像块的序列或文件。所得到的压缩图像块可以在文件中从左到右并且从上到下排序，与它们在步骤 S1 的块分解中被分解的次序相同。然后该方法结束。

编码图像可以被提供给存储器用于存储其中，直到后来的图像的再现，例如显示。而且，编码图像可以作为编码块表示的一个信号提供给发送器用于（无线或有线地）发送到另一个单元。

图 4 更详细地示例了图 1 的颜色确定步骤 S2 的实施例的流程图。该方法继续从图 1 的步骤 S1 开始。在下一个步骤 S10 中，确定图像块中的图像元素的平均颜色。在下文中，假设图像元素的颜色由 24 比特的 RGB 颜色表示，即 8 比特红色分量，8 比特绿色分量和 8 比特蓝色分量。然而，本发明并不局限于这个特定例子，而是可以应用到图像元素的任何颜色表示。平均颜色 $(\bar{R}, \bar{G}, \bar{B})$ 然后被确定为：

$$\begin{aligned}\bar{R} &= \frac{1}{N} \sum_{i=1}^N R_i \\ \bar{G} &= \frac{1}{N} \sum_{i=1}^N G_i \\ \bar{B} &= \frac{1}{N} \sum_{i=1}^N B_i,\end{aligned}\quad (1)$$

其中 R_i 、 G_i 、 B_i 为图像元素 i 的 R、G、B 分量，并且 N 为图像块中

的图像元素的总数。

一旦在步骤 S10 中确定了平均颜色 ($\overline{R}, \overline{G}, \overline{B}$)，下一个步骤 S11 量化平均颜色为将被用作这个实施例中的颜色码字的量化颜色。在该颜色量化中，24 比特平均颜色可以被量化成例如 9 比特、12 比特或 15 比特序列，但优选地量化为 18 比特序列。在这样的情况中，每个 8 比特平均颜色分量被量化成一个 6 比特平均分量（或对于总共 9、12、15 比特来说为 3 比特、4 比特、5 比特）。此外还应当看到，在颜色码字中的量化颜色分量的比特数量方面，各个大小可以不同。例如，大小 (G) \geq (大小 (R) 或大小 (B))。

如果平均颜色 $\overline{R}, \overline{G}, \overline{B}$ 被计算为：

$$\begin{bmatrix} 178 \\ 88 \\ 21 \end{bmatrix} = \begin{bmatrix} 10110010 \\ 01011000 \\ 00010101 \end{bmatrix}_{\text{bin}},$$

一个 6-比特版本 $\hat{R}, \hat{G}, \hat{B}$ 可以从下面生成：

$$\begin{bmatrix} 178 \\ 89 \\ 20 \end{bmatrix} = \begin{bmatrix} 10110010 \\ 01011001 \\ 00010100 \end{bmatrix}_{\text{bin}},$$

即 $[1011100, 010110, 000101]_{\text{bin}}$ 可以被用作 (12 比特) 颜色码字：然而应当注意，在其中原始图像元素颜色包括仅一个分量例如灰度级 (L) 的情况中，根据公式 2 在步骤 S10 中计算相应的平均灰度级 \bar{L} ：

$$\bar{L} = \frac{1}{N} \sum_{i=1}^N L_i \quad (2)$$

其中 L_i 为图像元素 i 的灰度级，并且 N 按照公式 2 定义。这个单平均灰度级然后可以在步骤 S11 中被量化并且用作颜色码字。在这样的情况中，颜色码字优选地包括量化的平均灰度级。

下一个步骤 S12 为图像元素确定强度或颜色修正码字并且选择强度或颜色修正因子索引。相应于阿尔法修正码字，这个颜色修正码字表示一组多个颜色修正因子，用于（在解码期间）修正图像块中的图像元素的颜色表示（强度）。

在本发明的优选实施例中，颜色修正码字为允许识别颜色修正因子组的索引或表示。这个索引然后可以在包含几个不同颜色修正因子组的

颜色或强度表格或密码本中识别或指向该组。这个表格可以以类似于表 1 的方式组织，即包括 16 个不同的修正因子组，修正因子组包括对称的修正因子值并且其中修正因子组的第一部分为第二部分中的修正因子组的 2 的因子。

在这样的颜色表格中实际的颜色修正因子值可以不同于阿尔法表格中的相应阿尔法修正因子值。然而，在一些应用中，可能使用同样的颜色修正因子值作为阿尔法修正因子值。为了节省存储区域，于是可以使用单个组合的阿尔法和颜色表格，例如根据表 1。如果最大和最小阿尔法值与最大和最小颜色（分量）值，例如 255 和 0，相同时，这尤其为正确。即使使用同样的阿尔法和颜色表格，阿尔法修正码字优选地独立于颜色修正码字。换句话说，可能这两个修正码字对于一些图像块包括同样的位组合（表格索引），但对于大多数的图像块，希望这两个修正码字将不同。在任何一种情况中，阿尔法修正码字和颜色修正码字可以被一起组织或编组成压缩图像块中的通用修正码字。然后这个通用修正码字将包括阿尔法修正（子）码字和颜色修正（子）码字。

在其他的应用中，阿尔法修正码字可以具有组合的阿尔法修正码字和颜色修正功能性。因此，例如从表格中使用这个阿尔法修正码字识别的修正因子组将被使用来修正由阿尔法码字表示的阿尔法值和修正由颜色码字表示的颜色分量。阿尔法码字的这一组合修正功能性节省了额外的比特，否则这些比特将被用于专用颜色修正码字。然而，这些额外的比特典型地以较差的颜色（或阿尔法）编码为代价而得到。

一旦已经选择了颜色修正因子组并且确定了颜色修正因子码字，步骤 S12 继续为图像块中的图像元素选择颜色修正因子索引或表示。每个这样的颜色修正因子索引与来自所选择的颜色修正因子组的一个颜色修正因子值相关。换句话说，颜色修正因子索引允许识别该组的哪个颜色修正因子被使用于块中的特定图像元素。

步骤 S12 优选地为块中的图像元素重复执行，如线 L3 所示意的。该方法然后继续到图 1 的步骤 S3。

使用颜色管理和码字生成的这个实施例来压缩的图像块 700 如图 3B 所示。表示 700 包含了有关图 3A 的压缩块的颜色码字 710、阿尔法码字 720、阿尔法修正码字 730 和阿尔法修正因子索引序列 740。而且，压缩块 700 还包括颜色修正因子码字 750 和颜色修正因子索引序列 760。记

住一些实施例中阿尔法修正码字 730 和颜色修正码字 750 可以被组织到通用修正码字中，而在一些其他的实现中，颜色修正码字 750 可以被省略，这样阿尔法和颜色修正功能由阿尔法修正因子码字 730 编码。应当看到，编码图像块 700 的颜色码字 710、阿尔法码字 720、阿尔法修正码字 730、颜色修正码字 750、阿尔法修正因子索引序列 740 和颜色修正因子索引序列 760 的相互间次序可以与该图中所示出的有所不同。还应当看到，码字和索引序列不必被连续布置。

如果图像块包含八个图像元素（见例如图 2A 和 2B）并且每个阿尔法和颜色修正因子索引为 2 比特，两个修正因子索引序列 740、760 将各为 16 比特。而且，假设阿尔法码字 720、阿尔法修正码字 730 和颜色修正码字 750 的相应大小分别为 6 比特、4 比特和 4 比特。如果颜色码字包括三个 6 比特颜色分量，图像块的编码表示 700 的整体大小为 64 比特并且获得 8bpp 的压缩率。

图 5 示例了图 1 的颜色确定步骤的另一个实施例的流程图。该方法从图 1 的步骤 S1 继续进行。在下一个步骤 S20 中，确定第一和第二颜色子码字。这些第一和第二子码字表示图像块中的图像元素的颜色，优选地分别表示块中的图像元素的第一和第二部分。

根据图 4 的颜色处理特别适合于压缩具有类似色度（色调）但不同亮度的图像元素的图像块，并且保持亮度分量好于色度分量。根据图 5 的该颜色处理对于压缩图像块特别有效，图像块的图像元素被分布到颜色空间中的两个簇。此外，假定图 4 中使用的颜色表格仅包含非零颜色修正因子。在这样的情况中，图 4 的颜色处理并不特别地适合于压缩图像块，在该图像块中，颜色码字对于所有的图像元素来说确切地表示了想要的颜色。由于所有的颜色修正因子具有非零值，当颜色修正因子值应当为每个图像元素加到颜色表示中时，这个想要的颜色不能在解码期间获得。然而，在图 5 的颜色处理中，两个颜色子码字中的至少一个可以准确地表示这个想要的颜色，允许图像块的更精确的压缩表示。

指向或与两个颜色子码字之一相关的颜色索引在下一个步骤 S21 中被选择。在第一实施例中，块中的每个图像元素与各个颜色索引相关。然而，在本发明的优选实施例中，结果得到压缩块表示的更小的整体大小，块中的图像元素的子集与从第一或第二子码字选择的预定义的颜色子码字相关。结果，不必为这个（这些）图像元素执行颜色索引的选择

或指配。例如，第一（最后）图像元素可以总是与第一（或第二）颜色子码字相关。颜色索引序列然后不必为这个第一（最后）图像元素包含颜色索引。结果，在图像块具有总共 8 个图像元素的情况中，序列将只包含 7 个颜色索引。本发明预见到多于一个的图像元素可以与颜色子码字预相关。

步骤 S21 优选地为块中的所有图像元素重复执行，为该块应当生成颜色索引，如线 L4 所示意的。该方法然后继续到图 5 的步骤 S3。

使用颜色管理和码字生成的这个实施例压缩的图像块 700 如图 3C 所示。表示 700 包含了颜色码字，除了阿尔法码字 720、阿尔法修正码字 730 和阿尔法修正因子索引序列 740 外，该颜色码字包括第一颜色子码字 710A 和第二颜色子码字 710B 以及颜色索引的序列或位图 770（对于块中的图像元素的仅一个子集优选地为一个强度索引）。应当看到，压缩图像块 700 的第一颜色子码字 710A、第二颜色子码字 710B、阿尔法码字 720、阿尔法修正码字 730、颜色索引序列 770 和阿尔法修正因子索引序列 740 的相互间次序可以与该图中所示出的有所不同。还应当看到，码字和索引序列不必被连续地布置。

图 5 中确定的两个颜色子码字可以通过首先确定平均颜色来生成或者确定图像块中的图像元素的重心。然后，计算穿过颜色空间中的平均颜色点的最佳线。这个最佳线可以使用主要分量分析来确定，这是本领域的技术人员所熟知的。简单地说，这个最佳线被选择作为颜色空间中的穿过平均颜色点的直线，这样它最小化“惯性矩”（均方差）。在这条线上的两点被选择、量化并用作颜色子码字。在第一实施例中，计算图像元素颜色在最佳线上的投影方差。识别在这条线上的两点，这两点处在距平均颜色点约 $3/4$ 方差的距离处。这两个选择点的颜色分量（沿着该线的平均颜色点的任一侧一个）然后被量化。这些量化值然后被用作颜色子码字。

在颜色子码字确定的可选实现中，块中的图像元素的颜色值被量化。然后在量化的颜色值中间执行穷尽搜索。该搜索的结果是两个量化的颜色值（然而应当看到这两个颜色值可以相等），其最小化与采用这两个选择的颜色值作为用于该块的颜色子码字相关的误差值。这两个最佳的量化颜色值然后可以直接被用作颜色子码字。然而，任选地，在颜色点之间执行适合的子码字的穷尽搜索，所述颜色点在距各个选择的颜

色值预定义的距离之内。在搜索中找到的最佳的量化颜色值（在最小化相关误差值的意义上）被用作颜色子码字。

在图 5 的颜色处理的扩充中，第一和第二颜色子码字在步骤 S20 中例如根据上面提到的实施例之一来确定。然而，在这个实施例中，至少三个颜色表示可以使用这两个颜色子码字生成。在这样的情况中，第一颜色表示可以通过扩展第一颜色子码字来生成，第二颜色通过扩展第二颜色子码字来生成，并且至少第三颜色表示通过扩展第一和第二颜色子码字的线性组合来生成。这样的线性组合可以是 $w \times$ 第一子码字 + $(1-w) \times$ 第二子码字，其中 $0 < w < 1$ 。

应当看到，这两个颜色子码字典典型地为同样大小的码字。然而，本发明并不局限于此。例如，第一颜色子码字可以被表示为比第二子码字更高的分辨率，因此它包含比第二颜色子码字更多的比特。

图 4 和 5 中公开的两个不同的颜色处理可以被视作两个可获得的压缩模式。因此，根据第一压缩模式即根据图 1 和 4 压缩图像块，得到了图 3B 的压缩块表示，而根据第二压缩模式即根据图 1 和 5 压缩块，得到了图 3C 的压缩块。于是可有利地根据第一压缩模式压缩给定阿尔法图像的一些图像块并且根据第二压缩模式压缩图像的其它块。在这样的情况中，在所得到的编码图像中的块的一部分将典型地为按照图 3B 而剩余的块然后为按照图 3C。

两个不同的压缩模式的使用通过能够更适合于图像块的各自特性而给编码方法提供了灵活性。作为具有在两个不同的压缩模式之间按块选择的结果，处理的阿尔法图像的图像质量与具有单一（固定）编码的现有技术编码方案相比有所提高。

为了确定对于一个给定的图像块应用这两个操作模式中的哪一个模式，块可以根据两个压缩模式被压缩。然后要使用的压缩模式和压缩块表示优选地基于与这两个编码块相关的误差表示的比较加以选择。这些误差表示为指示与根据第一或第二模式来压缩图像块有关的图像质量表示的度量。因此，将选择引起或与最小的误差表示有关的压缩模式和被压缩图像块。

这个误差表示生成基本上包括使用根据第一压缩模式压缩的图像块表示（图 3B 中的 700）来生成图像块的第一解压缩或解码表示。相应地，图像块的第二解压缩表示使用根据第二压缩模式压缩的图像块表示（图

3C 中的 700) 来生成。解压缩块的程序将在下面的解码论述中进行更为详细地描述。

计算指示以第一和第二解压缩块表示来表示该原始图像块的误差值。令 $R_0^i (G_0^i, B_0^i)$ 代表原始 (未处理的) 图像块中的图像元素 i 的红 (绿、兰) 颜色分量的值, 并且 $R_d^i (G_d^i, B_d^i)$ 代表第一解压缩块表示中的图像元素 i 的相应得到的红 (绿、兰) 颜色分量。于是误差值可以被计算为:

$$\varepsilon^2 = \sum_{i=1}^N \left[(R_d^i - R_0^i)^2 + (G_d^i - G_0^i)^2 + (B_d^i - B_0^i)^2 \right], \quad (3)$$

其中, N 为块中的图像元素的总数。任选地, 分量特定的权重 w_R 、 w_G 、 w_B 可以用于公式 (3)。也可以为第二压缩块和模式计算相应的误差值。两个得到的误差值然后被比较并且识别出最小的。与这个被识别的误差值有关的压缩模式和压缩块将被用于当前块。

可选地, 在图像块的实际处理 (压缩) 之前执行压缩模式的选择。结果, 每图像块必须生成仅单个压缩块表示。这个早的模式选择优选地是在块中图像元素的特性的研究基础上做出。在优选的实施例中, 该选择至少部分地基于相关颜色空间中图像元素的颜色分布。

如上面已经提到的, 这两个压缩模式适合于有点不同的图像块。例如, 当块中图像元素的颜色在线上或至少靠近线时, 第一压缩模式高度适合于使用:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} t + \begin{bmatrix} \overline{R} \\ \overline{G} \\ \overline{B} \end{bmatrix}, \quad (4)$$

其中, \overline{RGB} 为块中的图像元素的平均颜色并且 $t \in [-\infty, \infty]$ 。因此, 第一压缩模式适合于压缩图像块, 其图像元素在通过平均颜色点的线 (4) 上或靠近该线 (4), 并且在 RG 、 RB 和 GB 平面中具有 1 的斜率。然而, 如果图像块包括具有远离上面识别线的颜色值并且特别地被分成两簇的图像元素, 则第二压缩模式更适合。

基于颜色值分布的辨别可以通过比较平行于在与图 5 相关讨论的最佳线的矢量 (v_1) 与平行于上面讨论的线 (4) 的矢量 (v_2) 来实现。如果这两个矢量之间的角度 (θ) 小于阈值角度 (φ), 即

$$\theta = \arccos\left(\frac{v1 \cdot v2}{|v1||v2|}\right) < \varphi, \quad (5)$$

则第一压缩模式被选择，否则第二模式被选择。一旦要使用的压缩模式被选择，当前的图像块就根据这个选择的模式被压缩。

为了使得能够识别哪个压缩模式已经被用于给定图像块，这在解压缩期间非常重要，在图 3B 和 3C 中的压缩图像块表示 700 可以包括模式索引。这个模式索引可以是 1 比特的索引，其中 0_{bin} (1_{bin}) 根据图 3B (图 3C) 被用于压缩块表示。如果图 3B 中的压缩块 700 的整体大小在比特数量方面要在增加这个模式索引时保持不变，则例如阿尔法码字 720 的大小可以被减少一比特，因此优选地现在为 5 比特。

在本发明的优选实现中，所得到的压缩块表示的大小（比特数量）是相同的，与所使用的压缩模式无关。在这样的情况中，图 3C 中的压缩块表示的整体大小优选地也为 64 比特。这可以通过使用一个 1 比特模式索引、两个 15 比特颜色子码字 710A、710B、一个 6 比特阿尔法码字 720、4 比特阿尔法修正码字 730、7 比特颜色索引序列（对于块中的除了第一图像元素之外的每个图像元素，一个 1 比特颜色索引，其预定义与第一颜色子码字 710A 有关）以及 16 比特阿尔法修正因子索引序列 740 来获得。

可以期望多数阿尔法图像的（很）大部分图像块当使用第一和第二模式之间的选择时将根据第一压缩模式被压缩。在上面讨论的两个模式的优选实现中，两个模式均可以访问 63 个“有用”比特，其可被用于压缩图像块。剩余的单个比特然后被用作模式索引。在本发明的可选实施例中，使用了与单个比特不同类型的模式索引。那个模式索引然后将使得能够对两个压缩模式有不同量的分配的“有用”比特，但是仍旧保持压缩图像块的整体大小。

在这个实施例中，单一模式比特被省略，这释放了 1 比特。颜色表格的条目将被用于在两个模式之间进行区别。结果，仅一部分颜色表格条目，例如这些 16 个条目中的 12 个条目，将被用作颜色修正因子组。剩余的（四个）表格条目为模式索引，用信号通知第二压缩模式。例如， 0001_{bin} - 0011_{bin} 、 0101_{bin} - 0111_{bin} 、 1001_{bin} - 1011_{bin} 和 1101_{bin} - 1111_{bin} 中的一个 4 比特字可以表示一个颜色修正码字并且用信号通知第一压缩模式将被使用。因此，在第一压缩模式中，模式索引和颜色修正码字

共享这同样的四比特。

在这个实施例中，修正的颜色表格可以用于增强第一模式下图像块的亮度分辨率。乍一看，第一模式的这个分辨率改进是以对于第二压缩模式的少三个有用比特（由于模式索引，0000_{bin}、01000_{bin}、1000_{bin}或1100_{bin}而少四个有用比特，但是由于省略该单一模式比特而有一个额外的有用比特）为代价而得到的。然而，通过仔细研究建议的“模式2”特定的模式索引，我们注意到这两个 LSB 对于所有的这些模式索引是相同的（00_{bin}）。结果，这两个 MSB 比特可以被使用来表示“有用”数据，例如颜色子码字、阿尔法码字、阿尔法修正码字、颜色索引或阿尔法修正因子索引数据。结果，对于第一压缩模式来说，额外的 1 比特改进仅以对第二压缩模式来说少一个有用比特为代价而得到。这个单个比特优选地取自第二颜色子码字，更优选地来自第二子码字的兰色分量。

图 6 更详细地示例了图 1 的确定阿尔法码字、确定阿尔法修正码字以及选择阿尔法修正因子索引的步骤的流程图。该方法从图 1 的步骤 S2 继续进行。在下一个步骤 S30 中，确定图像块中的图像元素的平均阿尔法值。在下文中，假定图像元素的阿尔法值由 8 比特表示，即可能从 0 到 255。平均阿尔法 ($\bar{\alpha}$) 然后被确定为：

$$\bar{\alpha} = \frac{1}{N} \sum_{i=1}^N \alpha_i, \quad (6)$$

其中，阿尔法_i为图像元素 i 的阿尔法值并且 N 为图像块中的图像元素的总数。

一旦在步骤 S30 中确定了平均阿尔法值 ($\bar{\alpha}$)，下一个步骤 S31 量化该平均颜色为一个量化的阿尔法值，其将用作这个实施例中的阿尔法码字。量化的阿尔法值的大小可能是例如 3 比特、4 比特或优选地为 5 比特或 6 比特。

下一个步骤 S32 研究阿尔法表格的不同的阿尔法修正因子组以及该组的不同的阿尔法修正因子值，并且为每个这样的修正因子组和修正因子值测试计算误差值。基于这些误差值，一个修正因子组和引起最小误差值的组的阿尔法修正因子值在步骤 S32 中被选择。该方法然后继续到图 1 中的步骤 S2 或结束。

解码

图 7 示例了根据本发明的解码被编码图像或原始图像的被编码版本

的方法的流程图。编码图像基本上包括图像块的若干编码表示，比如图 3A、3B 或 3C 的表示 700。这些编码的块表示优选地由上文结合图 1 讨论的图像编码方法生成。

该方法通常通过识别被编码的图像块开始来解码。有可能一个编码图像的所有编码图像块应当被解码来生成原始图像的解码表示。可选地，原始图像的仅一部分将被存取。结果，仅选定数量的图像块必须被解码（或更精确地，一定图像块的选定数量的图像元素必须被解码）。

一旦识别出正确的编码图像块，步骤 S40 提供阿尔法修正因子组。这个修正因子组在压缩块表示中的阿尔法修正码字的基础上提供。该组供应优选地通过依靠阿尔法修正码字识别来自上面包含多个修正因子组的表格，例如表 1 的阿尔法修正因子组来执行。然而，在一些应用中，有可能阿尔法修正码字本身包含修正因子组并且不需要任何表格查找。

在下一个步骤 S41 中，为图像块的至少一个图像元素（即为应被解码的该图像元素）生成颜色表示。这个颜色生成在压缩块中的颜色码字的基础上执行。

下一个步骤 S42 基于阿尔法码字为至少一个图像元素生成一个阿尔法表示。在步骤 S43 中，选择将被用于应当被解码的图像元素的阿尔法修正因子。该修正因子值从在步骤 S40 中提供的修正因子组中选择，该选择是基于与该图像元素相关并且在压缩的块表示的阿尔法修正因子索引序列中找到的阿尔法修正因子索引。在步骤 S43 中，一旦正确的强度修正因子值被选择，图像元素的阿尔法表示在步骤 S44 中使用该值被修正或调制。根据本发明，阿尔法修正指的是用一个阿尔法修正因子值修正阿尔法表示，例如相加或相乘。

步骤 S42 到 S44 可以为图像块中的几个图像元素来执行（如线 L5 所示意的）。本发明预见到：在一些应用中，仅单个图像元素从特定图像块中解码，特定图像块的多个图像元素被解码和/或特定块的所有图像元素被解码。

步骤 S40 到 S44 然后优选地为包含应当被解码的图像元素的所有图像块重复执行（如线 L6 所示意的）。这意味着步骤 S40 到 S44 的循环可以被执行一次，但很多时间为不同的编码图像块执行若干次和/或为特定的编码图像块执行若干次。

在任选步骤 S45 中，原始图像的解码表示或其一部分，基于解码图

像元素和块来生成。应当看到，在一些应用中，若干图像元素必须被解码，从而来再现解码表示的单个像素、纹理元或体元。例如，在三线内插期间，八个图像元素被解码，并且对于二线内插相应的数量为四个图像元素。该方法然后结束。

图 8 示例了图 7 的颜色表示生成步骤的实施例的流程图。颜色解码的这个实施例表示第一解码或解压缩模式并且特别适合于解压缩根据第一压缩模式压缩的图像块（见图 3B）。该方法从图 7 的步骤 S40 继续。在下一个步骤 S50 中，颜色码字的量化颜色，优选地为 15、16 或 18 比特，被扩展或扩充为优选 24 比特。对于具有 RGB 颜色的情况，颜色码字的每个量化的 5 比特或 6 比特颜色分量然后被扩展为 8 比特颜色分量。换句话说，一个 5 比特颜色分量 10110_{bin} 的被扩展为 $1011\ 0101_{bin}$ 。这个扩展的颜色然后被指配给将被解码的图像块的图像元素。

在随后的步骤 S51 中，正确的颜色修正因子组被识别并且从颜色表格（专用颜色表格或共有颜色和阿尔法表格）中根据颜色修正码字来选择。在其中共有修正码字使用于颜色和阿尔法的情况中，这个共有码字将被用于识别正确的颜色修正因子组。如果存储在颜色表格中的颜色修正因子组包含了第一子组的修正因子值，例如 $[a, b]$ ，第二子组的颜色修正因子值可以从第一子组的这些值中确定，例如 $[-a, -b]$ 。在步骤 S52 中，识别和选择的颜色修正因子值被加到这个扩展的颜色。因此，这个修正因子值被加到图像元素的颜色所有分量上（对于 RGB 颜色共三个或者对于灰度级颜色为单个分量）。这可能实现为将修正因子值简单加到所有的颜色分量上。然而，在一些应用中，可能优选地在将其加到分量上之前加权修正因子值。在这样的情况中，可以为不同的颜色分量使用不同的权重。在可选的实施例中，除了简单加法之外可以采用另一种类型的修正，例如乘法或异或。

在下一个任选步骤 S53 中，所得到的修正的颜色分量值被箝位在最小的颜色阈值和最大的颜色阈值之间。例如，如果在将（可能的加权的）颜色修正因子值加到颜色分量之后，所得到的值小于最小阈值，该值被箝位到这个阈值的值上。相应地，如果所得到的值大于最大阈值，该阈值的值应当作为替代被用于那个分量。最小和最大阈值的非限制性例子对于具有 256 个颜色分量级别的情况分别为 0 和 255。步骤 S50 到 S53 优选地为应当被解码的压缩块的编码图像元素重复执行，如线 L7 所示意

的。该方法然后继续到图 7 的步骤 S42。

图 9 示例了图 7 的颜色表示生成步骤的另一个实施例的流程图。颜色解码的这个实施例表示了第二解码或解压缩模式，并且特别地适合于解压缩根据第二压缩模式压缩的图像块（见图 3C）。该方法从图 7 的步骤 S40 继续。在下一个步骤 S60 中，颜色子码字被从第一或第二颜色子码字中、使用与相关的图像元素关联并且在颜色索引序列中找到的颜色索引加以选择。

然而，如果相关的图像元素属于图像元素的子集，其不具有相关的颜色索引而相反具有预定义的相关的颜色子码字，该预定义的颜色子码字在步骤 S60 中被识别。在下一个步骤 S61 中，图像元素的颜色表示使用选择的或预定义的颜色码字通过扩展选择的颜色子码字的量化颜色来生成。该颜色扩展类似于步骤 S50。步骤 S60 和 S61 可以为图像块中的若干图像元素执行（如线 L8 所示意的）。

可选地，特别地当解压缩一个压缩的块表示中的多个图像元素时，第一颜色表示使用第一颜色子码字来生成，并且第二相应颜色表示使用第二颜色子码字或者第一和第二子码字的组合来生成。然后为图像元素的选择是使用颜色索引在第一和第二颜色表示之间执行的，或者第一或第二颜色表示之一与块中的一个或一些图像元素预相关。

该方法然后继续到图 7 的步骤 S42。

用于当前图像块的相关解压缩模式优选地基于被包括在压缩图像块表示中的专用模式索引来确定。如果替代地颜色修正码字，或颜色修正码字的一些比特，被用作模式索引，则这些比特在生成该图像块的颜色表示之前被研究。

图 10 示例了图 7 的生成阿尔法表示、选择阿尔法修正因子以及修正阿尔法表示的步骤的流程图。该方法从图 7 的步骤 S41 继续。在下一个步骤 S70 中，阿尔法码字的量化的阿尔法值，优选地 5 或 6 比特，被扩展或扩充为优选地 8 比特。这个扩展的阿尔法值然后被指配给将被解码的图像块的图像元素。

在后面的步骤 S71 中，正确的阿尔法修正因子组被识别并且从阿尔法表格（专用阿尔法表格或共有阿尔法和颜色表格）中根据阿尔法修正码字选择。在其中一个共有的修正码字用于阿尔法和颜色的情况中，这个共有码字将被用于识别正确的阿尔法修正因子组。如果存储在阿尔法

表格中的该阿尔法修正因子组包含第一子组的修正因子值，例如 [a, b]，第二子组的颜色修正因子值可以从第一子组的值中确定，例如 [-a, -b]。在步骤 S72 中，被识别和选择的阿尔法修正因子值被加到扩展的阿尔法值上。这可以实现为将修正因子值简单加到阿尔法值。在可选的实施例中，除了简单加法之外可以采用另一种类型的修正，例如乘法或异或。

在下一个任选步骤 S73 中，所得到的修正的阿尔法值被箝位在最小的阿尔法阈值和最大的阿尔法阈值之间。例如，如果在将阿尔法修正因子值加到阿尔法表示之后，所得到的值小于最小阈值，该值被箝位到这个阈值的值上。相应地，如果所得到的值大于最大阈值，该阈值的值应当作为替代被用于那个分量。最小和最大阈值的非限制性例子对于具有 256 级别的阿尔法的情况分别为 0 和 255。步骤 S70 到 S73 优选地为应当被解码的压缩块的编码图像元素重复，如线 L9 所示意的。该方法然后继续到图 7 的步骤 S40 或 S45。

解码被编码的图像块还将通过下面的三个例子被示例。根据表 1 的阿尔法表格将被使用。

解码例子 1

图像块的编码表示为根据图 3A 并且包含下面的数据比特 101001101100 0111 11 01 10 00 10 01 00 00_{bin}，其中比特 0-5 为颜色码字，比特 6-11 为阿尔法码字，比特 12-15 为阿尔法修正码字并且比特 16-31 为阿尔法修正因子索引序列。在这个解码例子 1 中，图像块的原始颜色为灰度级。

颜色码字然后被解码（扩展）来生成图像块的颜色表示。6 比特的位组合格式被重复来生成一个 8 比特字。这个 8 比特字将包括颜色码字作为它的六个 MSB 以及该颜色码字的两个 MSB 作为它的两个 LSB。

颜色表示：10100110_{bin} ⇔ 166

这个扩展的颜色被指配给图像块的图像元素，给定：

(166)	(166)	(166)	(166)
(166)	(166)	(166)	(166)

(6 比特) 阿尔法码字被以类似方式解码（扩展）成一个 (8 比特)

阿尔法表示：

阿尔法表示：10110010_{bin} ⇔ 178

这个生成的阿尔法值被指配给图像块的图像元素，给定：

(166; 178)	(166; 178)	(166; 178)	(166; 178)
(166; 178)	(166; 178)	(166; 178)	(166; 178)

将被使用的正确的阿尔法修正因子组从表 1 中基于强度码字选择。

如表 1 所示，阿尔法修正码字 0111_{bin} 对应阿尔法修正因子 [-127, -42, 42, 127]。

阿尔法修正因子索引序列使得能够根据下式识别这四个修正因子值中的哪一个被用于不同图像元素：

$$\begin{bmatrix} 11_{bin} \\ 10_{bin} \\ 00_{bin} \\ 01_{bin} \end{bmatrix} = \begin{bmatrix} -127 \\ -42 \\ 42 \\ 127 \end{bmatrix}$$

第一阿尔法修正因子索引为 11_{bin}，其意味着第一阿尔法修正因子值，-127，应当被加到第一图像元素的阿尔法值上，178-127=51。因为这个值 51 在 0 和 255 的可接受间隔之间，不需要对其进行任何箝位。这个过程对第二图像元素索引 01_{bin} 继续，其对应于修正因子值 127。该值被加到 178 上得到 305。这个结果然后被箝位在 0 和 255 之间，因此给定 255。为块中的所有图像元素重复该过程，将创建最终的解码图像块，如下所示：

(166; 51)	(166; 255)	(166; 136)	(166; 220)
(166; 136)	(166; 255)	(166; 220)	(166; 220)

解码例子 2

图像块的编码表示为根据图 3B，加有单个模式索引，并且包含下面的数据比特 0101001 101100 001100 0111 10011 0010 11 01 10 00 10 01 00 00 10 01 11 00 10 01 11 11_{bin}，其中比特 0 为模式索引，比特 1-6 为颜色码字的红色分量，比特 7-12 为颜色码字的绿色分量，比特 13-18 为颜色码字的蓝色分量，比特 19-22 为颜色修正码字，比特 23-27 为阿尔法码字，比特 28-31 为阿尔法修正码字，比特 32-47 为颜色修正因子索引序列并且比特 48-63 为阿尔法修正因子索引序列。模式比特值 0_{bin} 表示第一解压缩模式，应当被使用于当前压缩块。在这个例子中，表 1 被用作组合的阿尔法和颜色表格。

颜色码字的每个颜色分量被扩展为 8 比特颜色分量表示，类似于解码例子 1：

红: $10100110_{bin} \leftrightarrow 166$

绿: $10110010_{bin} \leftrightarrow 178$

兰: $00110000_{bin} \leftrightarrow 48$

这些扩展的颜色分量被指配给图像块的图像元素, 给定:

(166, 178, 48)	(166, 178, 48)	(166, 178, 48)	(166, 178, 48)
(166, 178, 48)	(166, 178, 48)	(166, 178, 48)	(166, 178, 48)

将被使用的正确的颜色修正因子组从表 1 中基于颜色修正码字选择。如表 1 所示, 颜色修正码字 0111_{bin} 对应颜色修正因子 $[-127, -42, 42, 127]$ 。第一颜色修正因子索引为 11_{bin} , 其代表 -127 (与解码例子 1 成对比)。这个颜色修正因子值被加到第一图像元素的所有三个颜色分量上:

$$\begin{bmatrix} 166 \\ 178 \\ 48 \end{bmatrix} + \begin{bmatrix} -127 \\ -127 \\ -127 \end{bmatrix} = \begin{bmatrix} 39 \\ -51 \\ -79 \end{bmatrix}$$

所得到的分量然后被箝位在 0 和 255 之间, 因此给定 $(39, 0, 0)$ 。为剩余的图像块继续这个程序, 结果得到部分解码的图像块:

(39, 00, 00)	(255, 255, 175)	(124, 156, 6)	(208, 220, 90)
(124, 156, 6)	(255, 255, 175)	(208, 220, 90)	(208, 220, 90)

这个 (5 比特) 阿尔法码字被以类似方式解码 (扩展) 成一个 (8 比特) 阿尔法表示:

阿尔法表示: $10011100_{bin} \leftrightarrow 156$

将被使用的正确的阿尔法修正因子组从表 1 中基于强度码字选择。如表 1 所示, 阿尔法修正码字 0010_{bin} 对应阿尔法修正因子 $[-31, -6, 6, 31]$ 。阿尔法表示 156 使用阿尔法修正因子索引来修正。这个阿尔法修正以类似于颜色修正的方式被执行。最后的解码图像块如下所示:

(39, 00, 00; 150)	(255, 255, 175; 187)	(124, 156, 6; 125)	(208, 220, 90; 162)
(124, 156, 6; 150)	(255, 255, 175; 187)	(208, 220, 90; 125)	(208, 220, 90; 125)

解码例子 3

图像块的编码表示为根据图 3C, 加有单个模式索引, 并且包含下面的数据比特 $110100\ 10110\ 00110\ 01101\ 11110\ 00000\ 011110\ 0010$

1100100 10 01 11 00 10 01 11 11_{bin}, 其中比特 0 为模式索引, 比特 1-5 为第一颜色子码字的红色分量, 比特 6-10 为第一颜色子码字的绿色分量, 比特 11-15 为第一颜色子码字的蓝色分量, 比特 16-20 为第二颜色子码字的红色分量, 比特 21-25 为第二颜色子码字的绿色分量, 比特 26-30 为第二颜色子码字的蓝色分量, 比特 31-36 为阿尔法码字, 比特 37-40 为阿尔法修正码字, 比特 41-47 为颜色索引序列并且比特 48-63 为阿尔法修正因子索引序列。模式比特值 1_{bin} 指示第二解压缩模式应当被使用于当前压缩块。

颜色子码字被解码(被扩展)以生成两个颜色表示。在颜色子码字中的每个颜色分量被扩展为 8 比特:

第一颜色表示

红: 10100101_{bin} ↔ 165

绿: 10110101_{bin} ↔ 181

蓝: 00110001_{bin} ↔ 49

第二颜色表示

红: 01101011_{bin} ↔ 107

绿: 11110111_{bin} ↔ 247

蓝: 00000000_{bin} ↔ 0

第一图像元素在这个例子中总是与第一颜色子码字和表示相关, 并且因此被指配颜色(165, 181, 49)。对于第二图像元素颜色索引为 1_{bin}, 意味着第二颜色表示(107, 247, 0)应当被指配给这个图像元素。继续这个程序得到部分解压缩的块表示:

(165, 181, 49)	(107, 247, 0)	(107, 247, 0)	(165, 181, 49)
(165, 181, 49)	(107, 247, 0)	(165, 181, 49)	(165, 181, 49)

这个(6 比特)阿尔法码字被以类似方式解码(扩展)成一个(8 比特)的阿尔法表示:

阿尔法表示: 01111001_{bin} ↔ 121

将被使用的正确的阿尔法修正因子组从表 1 中基于强度码字选择。如表 1 所示, 阿尔法修正码字 0010_{bin} 对应阿尔法修正因子 [-31, -6, 6, 31]。阿尔法表示 121 使用阿尔法修正因子索引来修正。这个阿尔法修正以类似于颜色修正的方式被执行。最后的解码图像块如下所示:

(165, 181, 49; 115)	(107, 247, 0; 152)	(107, 247, 0; 90)	(165, 181, 49; 127)
(165, 181, 49; 115)	(107, 247, 0; 152)	(165, 181, 49; 90)	(165, 181, 49; 90)

实现讨论

根据本发明的图像编码(图像块编码)和图像解码(图像块解码)

方案可被提供在一般的数据处理系统中，例如在被配置用于处理和/或再现图像的用户终端或其它单元中。这样的终端可以是计算机或瘦客户端（thin client），比如个人数字助理（PDA）、移动单元或移动电话。由于根据本发明的编码和解码可以很容易地以硬件、软件或硬件和软件的组合的形式来实现，本发明可以有利地应用于瘦客户端。

图像处理终端

图 11 示例了由移动单元代表的图像处理终端 100。然而，本发明并不局限于移动单元，而是可以被实现在其它的终端中以及数据处理单元中。在图中仅示出直接涉及本发明的移动单元 100 中的装置和部件。

移动单元 100 包含用于处理移动单元 100 中的数据的（中央）处理单元（CPU）200，该数据包括图像数据。图形系统 130 被提供在移动单元 100 中，用于管理图像和图形数据。特别地，图形系统 130 适合于在连接的屏幕 120 或其它显示单元上再现或显示图像。移动单元 100 还包含存储装置或存储器 140，用于在其中存储数据。在这个存储器 140 中，可以存储图像数据，特别是根据本发明的编码的图像数据（压缩的图像块）。

根据本发明的图像编码器 210 被提供在移动单元 100 中。这个编码器 210 被配置来编码图像或纹理成该图像（或纹理）的编码表示。如上所讨论的，这样的编码表示包含多个压缩图像块的序列或文件。这个图像编码器 210 可以被提供作为运行在 CPU 200 上的软件，如图中所示出的。替换地，或者附加地，编码器 210 可以被安排在图形系统 130 中或移动单元 100 中的其它地方。

来自块编码器 210 的图像的编码表示可以通过（存储器）总线 150 提供给存储器 140，用于存储在那里直到后来的图像再现。替换地，或者附加地，编码图像数据可以被转发到输入和输出（I/O）单元 110，用于（无线或有线）传输到其它的外部终端或单元。这个 I/O 单元 110 还可以适合于接收来自外部单元的图像数据。这个图像数据可以为应当由图像编码器 210 编码的图像或者应当被解码的编码图像数据。也有可能将编码图像表示存储在例如图形系统 130 中提供的专用纹理存储器中。而且，部分编码图像还可以或者替换地被（暂时）存储在例如图形系统 130 的（纹理）高速缓存存储器中。本发明的便宜（在复杂性方面）和快速解压缩的巨大优势是压缩的图像块可以至少暂时存储在高速缓存中，以

便快速和容易地存取。这进一步由于高压缩率而变得容易，其允许数量四倍于未压缩的 (RGBA8888) 块数据的图像块数据同时存储在高速缓存中。

根据本发明的图像解码器 220 被提供在移动单元 100 中，用于解码编码图像，以便生成解码图像表示。这个解码表示可以对应于整个原始图像或其一部分。图像解码器 220 提供解码图像数据给图形系统 130，该系统反过来典型地在于屏幕 120 上再现或显示数据之前处理该数据。图像解码器 220 可以被安排在图形系统 130 中，如图中所示的。替换地，或者附加地，解码器 200 可以被提供作为软件运行在 CPU 200 或移动单元 100 中的其它地方上。

移动单元 100 可以被配备图像编码器 210 和图像解码器 220，如图中所示的。然而，对于某些终端 100，它可能仅包括图像编码器 210。在这样的情况中，编码图像数据可以被传输到执行图像解码和可能的再现的另一个终端。相应地，终端 100 可能仅包括图像解码器 220，即无编码器。这样的终端 100 然后接收包含来自另一个终端的编码图像数据的信号并且将它解码来生成解码图像表示。因此，编码图像信号可能在终端之间使用无线发送机和接收机来无线传送。可选地，用于在终端之间分发图像和编码图像表示的其它技术可以被采用，比如使用 IR 端口的 IR 技术、蓝牙和终端间的图像数据的有线传输。可以在终端间被连接和互换的存储器卡或芯片也可以用于该图像数据终端间分发。

移动单元 100 的单元 110、130、200、210 和 220 可以作为软件、硬件或其组合的形式来提供。

编码器

图 12 示例了根据本发明的图像编码器 210 的实施例的框图。编码器 210 典型地包含图像分解器 215，用于将输入图像分解或分割成若干图像块。分解器 215 优选地被配置用于分解图像成包含八个图像元素的图像块。这个分解器 215 可以适合于分解不同的输入图像成具有不同大小的图像块。在这样的情况中，分解器 215 优选地接收输入信息，使得能够识别哪个图像块格式将使用于给定图像。

图像块编码器 210 还包含块编码器 300。这个块编码器 300 编码从图像分解器接收的图像块，以生成编码的块表示。这样的图像块表示包含至少一个颜色码字、阿尔法码字、阿尔法修正码字和阿尔法修正因子

索引序列。块表示的整体大小比未编码的图像块的相应大小小得多。块编码器 300 优选地被配置来顺序处理（编码）来自分解器 215 的每个图像块。

块编码器 300 优选地包含或可存取包含多个阿尔法修正因子组的阿尔法表格 500。阿尔法表格 500 的修正因子组被用于生成阿尔法修正码字。阿尔法表格 500 可以被安排在块编码器 300 中或图像编码器 210 中的其它地方。

这个阿尔法表格 500 可以为专用阿尔法表格或共有阿尔法和颜色表格。在前一种情况中，块编码器 300 然后可能附加地包括颜色表格。也可能在编码器 210 中实现若干不同的表格，其中表格的阿尔法修正因子适合于不同图像类型的区别或适合于各个特定图像。当被配置以根据多个压缩模式工作时，同样的阿尔法表格 500 可以被使用在所有的模式中，然而，在一些应用中，模式特定的阿尔法表格 500 可以替代地被使用。

尽管图 12 的图像编码器 210 仅包括单个块编码器 300，从处理时间的角度看，在图像编码器 210 中实现多个块编码器 300 可以更有利。通过提供多个块编码器 300。来自分解器 215 的多个图像块可以被并行处理（编码），这减少了整个图像编码时间。可选地，第一子组块编码器可以按照第一压缩模式工作来压缩图像块，并且剩余子组的编码器可以按照第二压缩模式来工作。

在多个块编码器的情况中，每个块编码器 300 可以包含阿尔法表格 500。在不同编码器 300 中的阿尔法表格 500 可以都包含同样的阿尔法修正因子值。可选地，模式特定的阿尔法表格可以由编码器 300 来使用。在可选的实现中，单个阿尔法表格 500 被安排在图像编码器 210 中并且连接到所有的块编码器 300 中。

图像编码器 210 的单元 215 和 300 可以以软件、硬件或其组合的形式提供。单元 215、300 和 500 可以一起实现在图像编码器 210 中。可选地，利用在用户终端中的其它地方提供的一些单元，还可能实现分布式实现。

图 13 示例了根据本发明的块编码器 300 的实施例的框图，比如图 12 中的图像编码器的块编码器。编码器 300 包含颜色量化器 310，其生成表示图像块中图像元素的颜色颜色码字。阿尔法量化器 320 被提供在块编码器 300 中用于生成阿尔法码字，该阿尔法码字表示块中的图像元

素的阿尔法值。

阿尔法修正因子提供器 330 被实现在块编码器 300 中, 以便识别用于当前图像块的阿尔法修正因子组。提供器 330 优选地被配置用于从相关的阿尔法表格 500 中选择这个修正因子组。该修正因子提供器 330 然后生成与所选择的阿尔法修正因子组相关的阿尔法修正因子码字。

编码器 300 还包括索引选择器 340, 用来为块中的图像元素选择图像元素相关的阿尔法修正因子索引。每个这样的阿尔法修正因子索引与来自阿尔法修正因子提供器 330 识别的阿尔法修正因子组的阿尔法修正因子之一相关。

块编码器 300 任选地可以包括误差估计器 350, 以便估计误差值用于选择与图像块的索引相关的码字和图像元素。最小化相关误差值的码字和索引的选择然后可以为相关的压缩图像块版本来选择。

块编码器 300 的单元 310 至 350 可以作为软件、硬件或其组合的形式来提供。单元 310 至 350 以及 500 可以一起实现在块编码器 300 中。可选地, 利用在图像编码器中的其它地方提供的一些单元, 还可能实现分布式实现。

图 14 示例了根据本发明的块编码器 300 的另一个实施例的框图。这个块编码器实施例能够按照两个以前描述的压缩模式工作。阿尔法量化器 320、阿尔法修正因子提供器 330 和误差估计器 350 的工作已经参照图 13 进行了描述, 并且这里不再重复。

颜色量化器 310 为将被压缩的图像块确定颜色码字。当按照第一压缩模式工作时, 这个确定的颜色码字为块中图像元素的颜色单一表示。然而, 当颜色量化器 310 按照第二模式工作时, 量化器 310 生成包括第一颜色子码字和第二颜色子码字的颜色码字。第一和第二颜色子码字然后分别表示图像元素的第一和第二部分的颜色。

颜色修正因子提供器 360 被实现在块编码器 300 中, 并且可工作在第一压缩模式下来识别颜色修正因子组以用于当前图像块。提供器 360 优选地被配置用于从相关的颜色表格 550 中选择这个修正因子组。修正因子提供器 360 然后生成与所选择的颜色修正因子组相关的颜色修正因子码字。

在第一模式下, 索引选择器 340 为块中每个图像元素从由颜色修正因子提供器 360 识别的颜色修正因子组选择颜色修正因子。选择器 340

然后生成与所选择的颜色修正因子相关的各个颜色修正因子索引。然而，当按照第二压缩模式工作时，索引选择器 340 为块中图像元素的子组中的每个图像元素生成一个与由颜色量化器 310 确定的颜色子码字之一相关的颜色索引。图像元素的剩余子组具有与颜色码字之一预定义的关联并且因此不需要任何颜色索引。

可以使这个编码器实施例总是按照第一压缩模式或按照第二压缩模式工作。然而，为了给阿尔法图像处理提供灵活性并且提高所得到的压缩阿尔法图像的图像质量，块编码器 300 可以优选地按照两个模式工作。在这样的情况中，编码器 300 优选地包括压缩模式选择器 370，用于为将被压缩的每个图像块选择一个压缩模式。这个选择器 370 可以工作用来基于块中的图像元素的输入颜色值（即在压缩块之前）选择压缩模式。可选地，执行该模式选择以响应于输入误差表示或输入压缩块表示（即在压缩块之后或期间）。与按照第一模式和第二模式压缩图像块相关的这些输入误差表示，可以由模式选择器 370 自己来生成或从误差估计器 350 接收。

模式选择器 370 还为块生成模式索引并且将其插入到压缩的图像块表示中，该模式索引代表所选择的压缩模式。在其中模式表示由颜色修正码字执行的情况中，无额外的模式索引必须由选择器 370 来生成。该模式选择器 370 优选地还生成和转发模式表示信号或命令给块编码器 300 中的至少某些单元。那个模式信号然后将使各个单元按照第一或第二压缩模式工作。

块编码器 300 的单元 310 至 370 可以作为软件、硬件或其组合的形式提供。单元 310 至 370，500 以及 600 可以被一起实现在块编码器 300 中。可选地，利用在图像编码器中的其它地方提供的一些单元，还可能实现分布式实现。

在图 15 的框图中，示例了根据本发明的颜色量化器 310 的优选实现。量化器 310 包含装置 312，被配置来确定图像块中图像元素的颜色平均值。

该确定的平均颜色然后被提供给量化装置 314，该装置可以直接量化平均颜色并且基于这个量化的平均颜色生成颜色码字。可选地，量化装置 314 或颜色平均器 312 确定穿过颜色空间中的平均颜色点的最佳线。在这条线上的两点然后由量化器 314 选择和量化成两个颜色子码字。选

择的点可以按照结合图 5 的上述讨论来确定。

颜色量化器 310 的单元 312 和 314 可以作为软件、硬件或其组合的形式来提供。单元 312 和 314 可以被一起实现在颜色量化器 330 中。可选地，利用在块编码器中的其它地方提供的一些单元，还可能实现分布式实现。

在图 16 的框图中，示例了按照本发明的阿尔法量化器 320 的优选实现。量化器 320 包含装置 322，被配置用于确定图像块中图像元素的阿尔法的平均。这个平均阿尔法值然后被提供给量化装置 324，该装置量化平均阿尔法值并且从这个量化的平均阿尔法值生成阿尔法码字。

阿尔法量化器 320 的单元 322 和 324 可以作为软件、硬件或其组合的形式来提供。单元 322 和 324 可以被一起实现在颜色量化器 310 中。可选地，利用在块编码器中的其它地方提供的一些单元，还可能实现分布式实现。

在图 17 的框图中，示例了按照本发明的索引选择器 340 的优选实现。这个索引选择器 340 包含阿尔法修正因子选择器 342，用于为块中的每个图像元素选择阿尔法修正因子。与这些被选择的阿尔法修正因子有关的阿尔法修正因子索引然后被生成并且组成阿尔法索引序列。

索引选择器 340 还优选地包含工作在第一压缩模式下的颜色修正因子选择器 344。对应于阿尔法修正因子选择器 342，这个颜色修正因子选择器 344 为块中的每个图像元素选择颜色修正因子。与这些被选择的阿尔法修正因子有关的颜色修正因子索引然后被生成并且组成颜色索引序列。

颜色选择器 346 优选地在索引选择器 340 中实现，用于在第二压缩模式下且为图像块的一部分中的每个图像元素选择与第一或第二颜色子码字相关的颜色索引。块的剩余部分中的图像元素缺少颜色索引并且替代地具有预定义的子码字关联。

索引选择器 340 的单元 342 至 346 可以作为软件、硬件或其组合的形式来提供。单元 342 至 346 可以被一起实现在索引选择器 350 中。可选地，利用在块编码器中的其它地方提供的一些单元，还可能实现分布式实现。

解码器

图 18 示例了根据本发明的图像解码器 220 的实施例的框图。图像解

码器 220 优选地包括块选择器 222, 以便例如从存储器中选择应当被提供给块解码器 400 用于解码的那个(些)编码图像块。块选择器 222 优选地接收例如来自报头或再现引擎的与编码的图像数据有关的输入信息。具有想要的图像元素的被压缩图像块的地址被基于输入信息计算。这个计算的地址优选地取决于图像之内的图像元素(像素、纹理元或体元)的坐标。使用该地址, 块选择器 222 识别例如来自存储器或高速缓存的编码图像块。这个被识别的编码图像块然后被从存储装置中取装并且被提供给块解码器 400。

对于图像块的图像元素的(随机)存取有利地使得能够选择性取装和解码图像的仅仅那些需要的部分。而且, 图像可以以数据被要求的任何顺序来解码。例如, 在纹理映射中, 仅部分纹理可以被需要并且这些部分将通常被要求以非连续的顺序。因此, 本发明的图像解码可以通过被应用去处理图像的仅一部分或片段而具有益处。

选择的编码图像块被转发到图像解码器 400 中。除了图像块之外, 解码器 400 还优选地接收指定块的哪些图像元素应当被解码的信息。该信息能够指定整个图像块即其中的所有图像元素应当被解码。然而, 所接收的信息可以识别出仅单个或一些图像元素应当被解码。块解码器 400 然后生成块中的图像元素的解压缩表示。这个解码表示优选地为 S 比特颜色, 其中 S 为原始图像中的每个图像元素的比特数量, 例如 24 比特 RGB 颜色、与 T 比特阿尔法值, 其中 T 为原始图像中每个图像元素的比特数量, 例如 8 比特阿尔法值。块解码器 400 优选地包含在解码程序期间使用的阿尔法表格 500。可选地, 这个阿尔法表格 500 可以被提供在图像解码器 220 中的其它地方。上面结合图 12 所讨论的, 对于不同的工作模式和/或图像类型使用不同的阿尔法表格, 也适用于图像解码器 220。

任选的图像组合器 224 可以提供在解码器 220 中。这个组合器接收来自块解码器 400 的解码图像元素并且将它们组合起来以生成可以在屏幕上再现或显示的像素、纹理元或体元。组合器 224 可以需要若干输入图像元素来生成单个像素、纹理元或体元。这个图像组合器 224 可以可选地被提供在图形系统中。

尽管图 18 的图像解码器 220 只包含单个块解码器 400, 从处理时间的角度看, 在图像解码器 220 中实现多个块解码器 400 可以是有利的。通过提供多个块解码器 400, 来自分解器 215 的多个图像块可以被并行

处理（解码），这减少了整个的图像解码时间。可选地，块解码器的第一子组可以按照第一解压缩模式工作来解压缩图像块，并且块解码器的剩余子组按照第二解压缩模式工作。

图像解码器 220 的单元 222、224 和 400 可以作为软件、硬件或其组合的形式来提供。单元 222、224、400 和 500 可以被一起实现在图像解码器 220 中。可选地，利用在用户终端中的其它地方提供的一些单元，还可能实现分布式实现。

图 19 为根据本发明的块解码器 400 的实施例的示例。块解码器 400 包含颜色生成器 410，用来基于颜色码字为图像块中的图像元素生成颜色表示。这个生成器 410 优选地扩展颜色码字的量化颜色到颜色表示。

阿尔法生成器 420 被实现在块解码器 400 中，用来基于阿尔法码字为块中的图像元素生成阿尔法值或表示。这个阿尔法生成器 420 优选地扩展阿尔法码字的量化阿尔法值到阿尔法表示。

块解码器 400 还包含装置 430，用于使用阿尔法修正码字提供阿尔法修正因子组。这个阿尔法修正因子组优选地从相关的阿尔法表格 500 中提供。这个提供器 430 可以被配置用来从阿尔法表格 500 中取装第一子组的阿尔法修正因子值，并且用来基于第一子组确定第二子组的修正因子。

索引或修正因子选择器 440 被安排用于从由提供装置 430 获得的修正因子组中选择阿尔法修正因子值之一。该修正因子选择器 440 被配置来基于阿尔法修正索引序列而为压缩图像块中的图像元素选择正确的阿尔法修正因子值。

来自阿尔法生成器 420 的扩展阿尔法值和来自修正因子选择器 440 的修正因子值被转发到使用修正因子值修正扩展的阿尔法值的阿尔法调制器或修正器 450。而且，一旦阿尔法值已经被修正，修正器 450 优选地将所得到的阿尔法表示籍位在最大和最小阈值之间，例如在 0 和 255 之间。

块解码器 400 的单元 410 至 450 可以作为软件、硬件或其组合的形式来提供。单元 410 至 450 和 500 可以被一起实现在块解码器 400 中。可选地，利用在用户终端中的其它地方提供的一些单元，还可能实现分布式实现。

图 20 示例了根据本发明的块解码器 400 的另一个实施例的框图。该

块解码器实施例能够根据两个前面描述的解压缩模式来工作。阿尔法生成器 420、阿尔法修正因子组提供器 430 以及阿尔法修正器 450 的工作已经参考图 19 进行了描述，并且这里不再重复。

块解码器 400 包含解压缩模式选择器 480，用来选择一模式，一个当前的压缩图像块应当根据该模式被解码。这个模式选择优选地基于被包括在压缩块表示中的模式索引。可选地，该模式选择可以通过研究（至少一部分）组合的颜色修正和模式表示码字来执行。模式选择器 480 生成表示模式索引和当前解压缩模式的模式信号或命令。这个模式信号被转发到块解码器 400 中的其它单元，其响应于所接收的模式信号来调整它们各自的工作。

当工作在第一解压缩模式下时，颜色生成器 410 基于颜色码字为图像块中的所有图像元素生成单个颜色表示。这个生成器 410 优选地扩展颜色码字的量化颜色到颜色表示。

块解码器 400 还包括装置 460，用于当接收来自模式选择器 480 的表示模式信号的“模式 1”时使用颜色修正码字提供颜色修正因子组。这个颜色修正因子组优选地从相关的颜色表格 550 中提供。这个提供器 460 可以被配置用于从颜色表格 550 中取装第一子组的颜色修正因子值，并且基于第一子组确定第二子组的修正因子。

索引选择器 440 被安排用来在第一模式下从由提供装置 460 获得的颜色修正因子组选择颜色修正因子值之一。选择器 440 被配置来基于颜色修正索引序列为压缩图像块中的图像元素选择正确的颜色修正因子值。然而，如果来自模式选择器 480 的模式命令改而发信号通知第二模式，索引选择器 440 使用颜色索引序列来选择颜色子码字或表示以用于图像元素的子组。当剩余子组的图像元素具有预定义的相关颜色表示（子码字）时，那个表示可以简单地被指配给这个（这些）图像元素而不使用颜色索引序列。

当解码器根据第一模式工作时，来自颜色生成器 410 的扩展颜色和颜色修正因子值被转发给颜色调制器或修正器 470，其使用颜色修正因子值修正扩展颜色的颜色分量（的强度）。修正器 470 可以使用加权的颜色修正因子值，对于不同颜色分量具有不同的权重。而且，一旦颜色分量已经被修正，修正器 470 优选地将该分量箝位在最大和最小阈值之间，例如在 0 和 255 之间。

块解码器 400 的单元 410 至 480 可以作为软件、硬件或其组合的形式来提供。单元 410 至 480、500 和 550 可以被一起实现在块解码器 400 中。可选地，利用在用户终端中的其它地方提供的一些单元，还可能实现分布式实现。

图 21 为示意性示例了根据本发明的块解码器的可能的硬件实现。按照图 3B，到块解码器的输入为编码块表示 700，该编码块表示包含了 18 比特颜色码字 710（红、绿和兰分量每个分量 6 比特）、4 比特颜色修正码字 750、16 比特颜色修正因子索引序列 760、16 比特阿尔法修正因子索引序列 740、4 比特阿尔法修正码字 730 和 6 比特阿尔法码字 720。在这两个索引序列 740、760 中的修正因子索引被组织以便八个颜色或阿尔法修正因子索引的 MSB 在八个 LSB 之前。因此，索引序列 10110001 10000110_{bin} 表示下面的修正因子索引 11_{bin}、00_{bin}、10_{bin}、10_{bin}、00_{bin}、01_{bin}、01_{bin} 和 10_{bin}。

颜色码字 710 被提供给由三个比特扩充器 412 到 416 实现的颜色生成器 410。第一个比特扩充器 412 接收 6 比特红色分量，第二 414 和第三 416 扩充器分别接收 6 比特绿色和兰色分量。来自各个扩充器 412 到 416 的输出为 8 比特颜色分量。

修正因子或索引选择器 440 被实现为四个多路复用器 442-448。3 比特地址索引被输入到这些多路复用器 442-448。第一多路复用器 442 响应于地址索引，为被寻址的图像元素输出颜色修正因子索引的 MSB。颜色修正因子索引的 LSB 由第二多路复用器 444 提供。相应地，第三 446 和第四 448 多路复用器分别选择和输出阿尔法修正因子索引的 MSB 和 LSB。

与所选择的图像元素相关的 2 比特颜色修正因子索引然后被转发到表格查找 465。这个表格查找 465 对应于图 20 中的颜色修正因子组提供者 460 和颜色表格 550。使用输入颜色修正码字 750 和颜色修正因子索引，查询 465 从表格中的修正因子组之一中取装正确的颜色修正因子值。这个 9 比特有符号的（正或负）修正因子值然后被提供给颜色修正器 470。在这个硬件实现中，颜色修正器 470 包含三个加法器 471 至 473 以及三个籍位器 474 至 476。修正因子值被输入到各个加法器 471 到 473。第一加法器 471 将颜色修正因子值加到来自比特扩充器 412 的 8 比特红色分量上。相应地，加法器 472 和加法器 473 分别将修正因子值加

到来自比特扩充器 414 和 416 的 8 比特绿色和蓝色分量上。在可选实现中，加法器 471 至 473 可以采用其它修正元件来代替，例如乘法器或异或门。加法器 471 至 473 的输出被转发到箝位器 474 至 476，其将修正的颜色分量箝位在 0 和 255 之间。

6 比特阿尔法码字 720 被提供给阿尔法生成器 420，其由一个比特扩展器 425 实现。来自该扩展器 425 的输出是 8 比特阿尔法值。

与选择的图像元素有关的 2 比特阿尔法修正因子索引被从多路复用器 446 和 448 转发到表格查找 435。这个表格查找 435 对应阿尔法修正因子组提供器 430 和图 19 或 20 中的阿尔法表格 700。使用输入的阿尔法修正码字 730 和阿尔法修正因子索引，查找 435 从表格中的修正因子组之一取装正确的阿尔法修正因子值。这个 9 比特有符号的（正或负）修正因子值连带来自比特扩充器 425 的被扩充的阿尔法值一起被提供给阿尔法修正器 450。

在这个硬件实现中，阿尔法修正器 450 包含加法器 452 和箝位器 454。阿尔法修正因子值被输入到将其加到来自扩充器 425 的 8 比特阿尔法值的加法器 452。由本发明预见到：加法器 452 可以由例如乘法器或异或门来替代。来自加法器 452 的输出被转发到将修正的阿尔法箝位在 0 和 255 之间的箝位器 454。

来自颜色箝位器 474 至 476 和阿尔法箝位器 454 的输出为解压缩或解码的 32 比特 RGBA 图像元素。

尽管图 21 的解码器实现包含两个独立的查找表格 435、465，一个用于管理颜色以及一个用于阿尔法，但可以采用一单个表格查找来解码这些图像元素特性。

图 22 示意性示例了图 21 的比特扩充器 412、414、416 和 425 的可能硬件实现。这些扩充器接收 6 比特颜色分量或阿尔法码字并且输出扩充的相应 8 比特颜色分量或阿尔法值。在该输出中，六个 MSB 组成输入 6 比特字，“第七 MSB”对应于输入的 MSB 并且 LSB 对应输入字的“第二 MSB”。

图 23 示意性示例了图 21 的箝位器 454、474-476 的可能的硬件实现。到箝位器 454、474-476 的输入为 10 比特的修正颜色分量值或阿尔法值。这个输入值的八个 LSB 被提供给多路复用器 456。其它的到多路复用器 456 的输入为最大的阈值（255；8 比特）。多路复用器 445 基于

输入字的第二 MSB 选择 8 比特输入值或最大阈值。换句话说, 如果这个第二 MSB 等于一, 多路复用器 456 输出这个阈值, 否则 8 比特输入值被转发到第二多路复用器 458。这个第二多路复用器 448 基于颜色分量的 MSB 选择来自第一多路复用器 456 的输出或最小阈值 (0; 8 比特)。如果这个 MSB 或符号比特等于一, 来自第一多路复用器 456 的输出为负并且该最小阈值应当由第二多路复用器 458 来选择。然而, 如果该符号比特为零, 来自第一多路复用器 456 的输出也应当为来自第二多路复用器 458 的输出。

图 24 示意性示例了图 21 的表格查找 435、465 的可能的硬件实现。这个 4 比特输入修正码字的三个 LSB 被输入到两个多路复用器 431 和 432, 用于从八个可能的修正因子值中为每个多路复用器 431 和 432 选择一个 7 比特阿尔法或颜色修正因子值。如果采用按照表 1 的阿尔法/颜色表格, 来自这些 16 个强度修正因子的剩余的 48 个值可以被计算。来自多路复用器 431 和 432 的被选择的修正因子值被输入到另一个多路复用器 433, 该复用器 443 基于 1 比特输入数据 (2 比特阿尔法或颜色修正因子索引中的 1 比特) 从图 21 中的多路复用器 442、444 或 446、448 中选择这些值之一。所选择的修正因子值然后被转发到多路复用器 436 和用来对修正因子值求反的非装置 434。这个被求反的值也被转发到多路复用器 436。这个多路复用器 436 基于来自图 21 中的多路复用器 442、444 或 446、448 的修正因子索引的剩余比特, 来选择正的 7 比特修正因子值或被求反的值。所选择的 (8 比特) 修正因子值被带给多路复用器 438 和比特移位器 437, 该比特移位器 437 将修正因子值向左移位一比特从而得到 9 比特修正因子值 (对应于以十为基数的该值乘以二)。多路复用器 438 基于来自输入修正码字的 MSB 选择该 8 比特修正因子值或 9 比特修正因子值。该选择的结果为 64 个可能修正因子值中的一 9 比特修正因子值, 用于特定图像元素。

图 21 中的块编码器的硬件方案非常简单, 如果表格查找 435、465 和籍位器 454、474-476 分别按照图 23 和图 24 实现, 它基本上包括仅四个加法器、一个求反装置、以及 22 个多路复用器。

本领域的技术人员应当理解, 可以对本发明做出各种修改和变化而不背离其由所附的权利要求所定义的范围。

参考文献

[1] 美国专利 No. 5, 956, 431

[2] Y. Linde, A. Buzo and R. Gray, "An algorithm for vector quantizer design(用于矢量量化器设计的算法)", IEEE Transactions on Communications, Vol. 28, 第 84-94 页, 1980 年 1 月

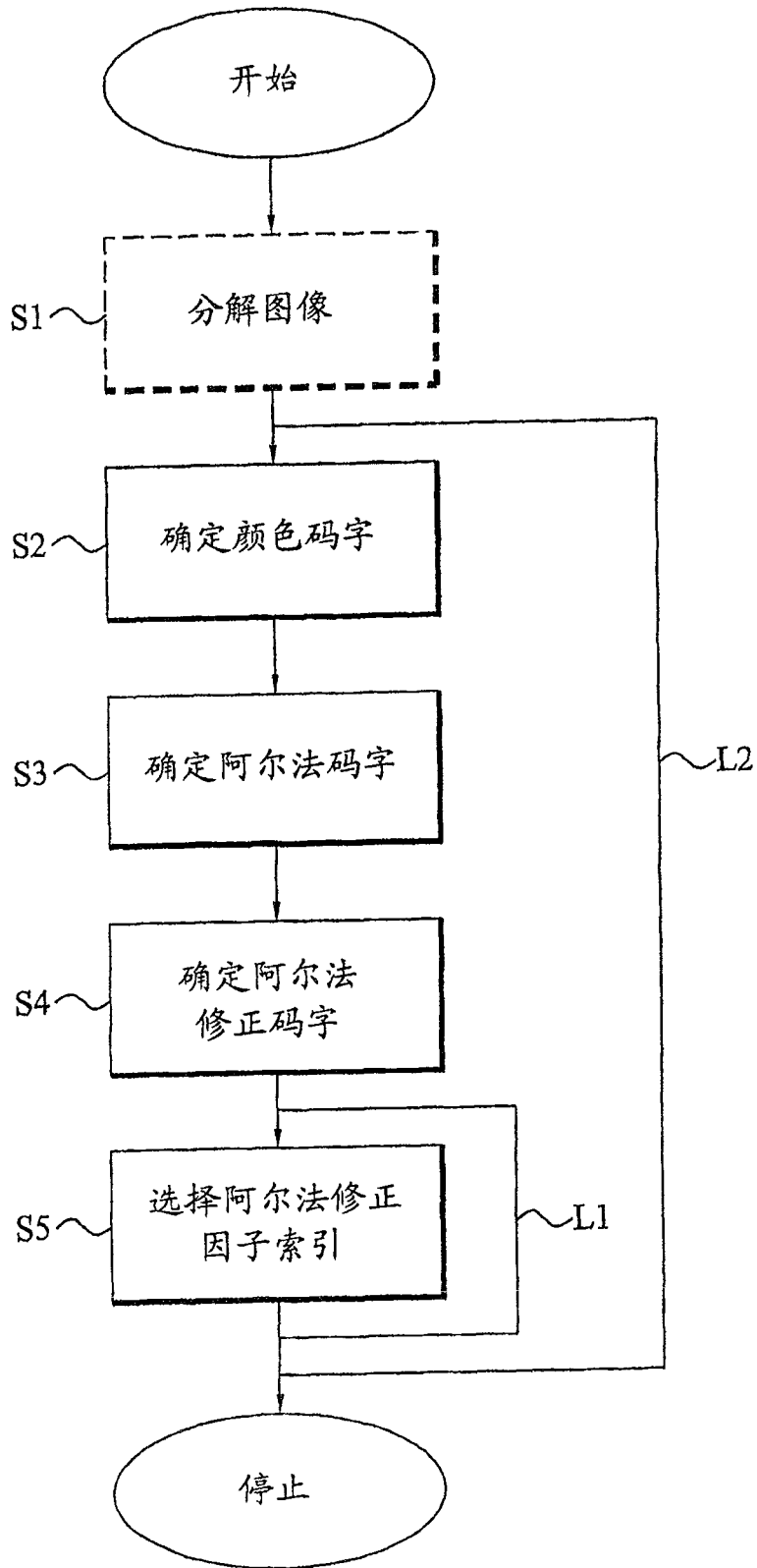


图 1

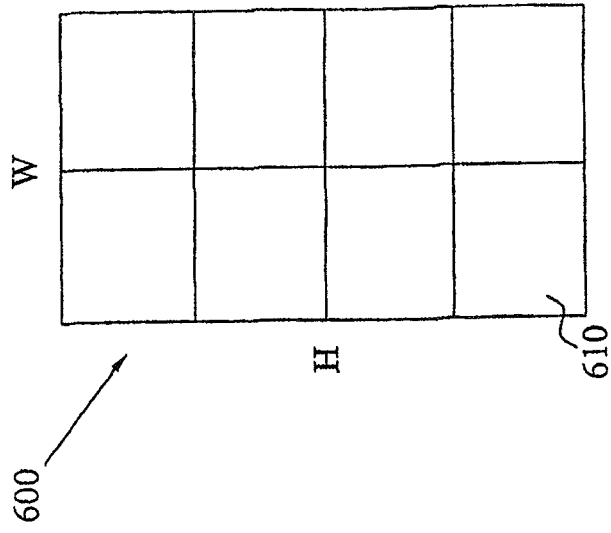


图 2B

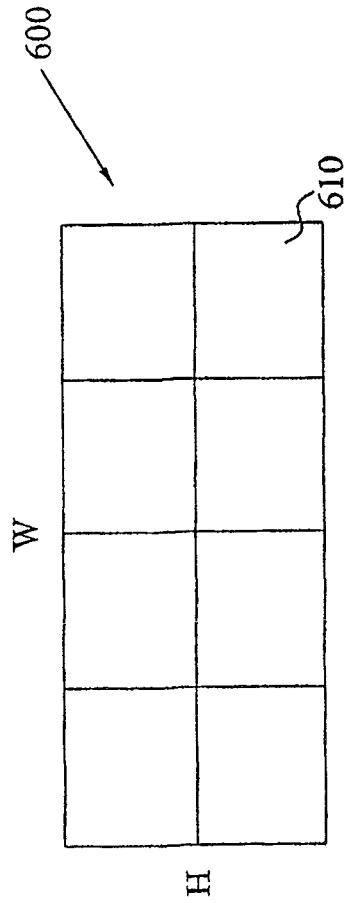


图 2A

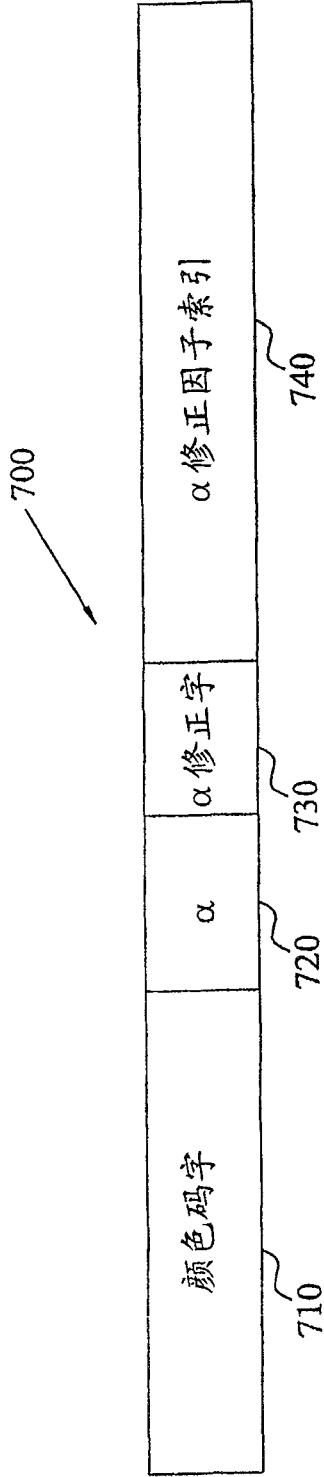


图 3A

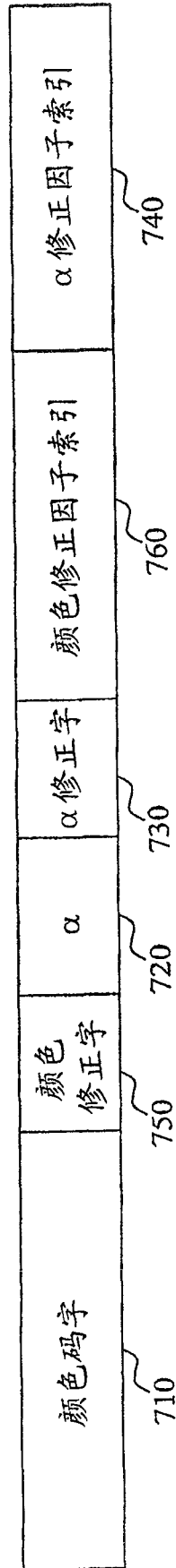


图 3B

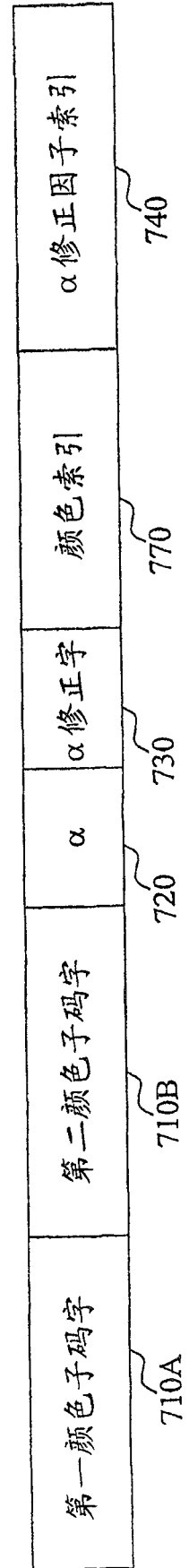


图 3C

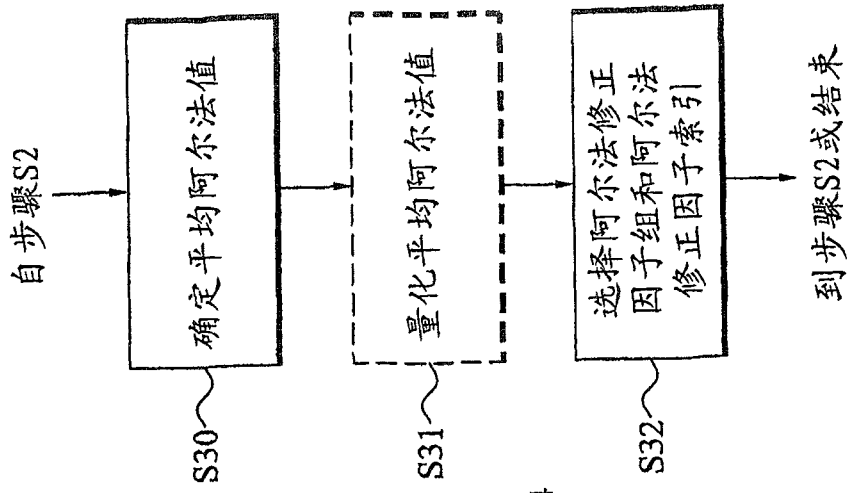


图 6

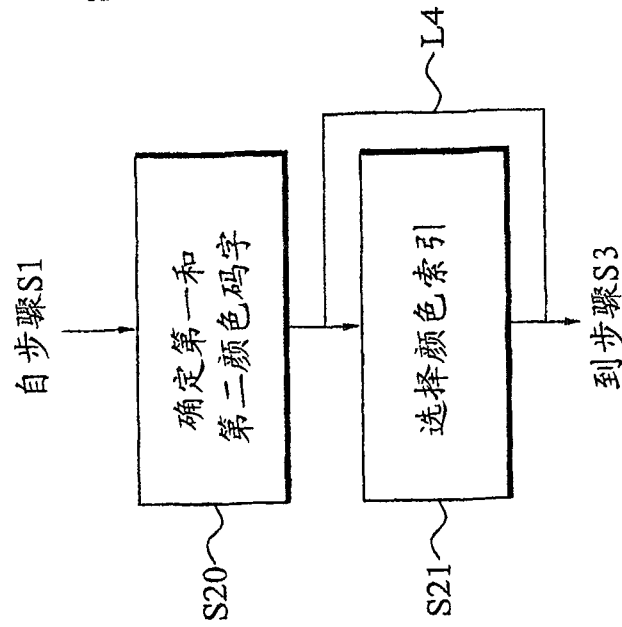


图 5

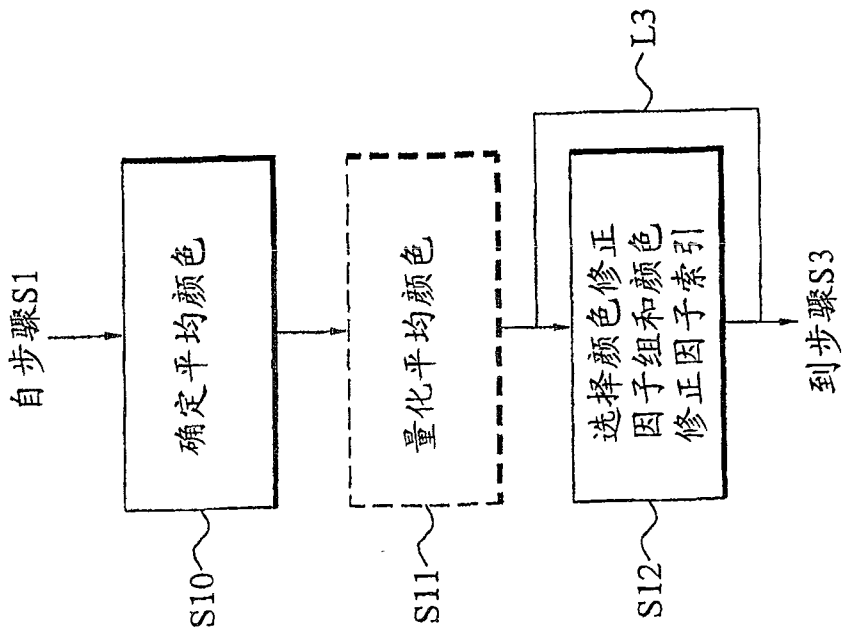


图 4

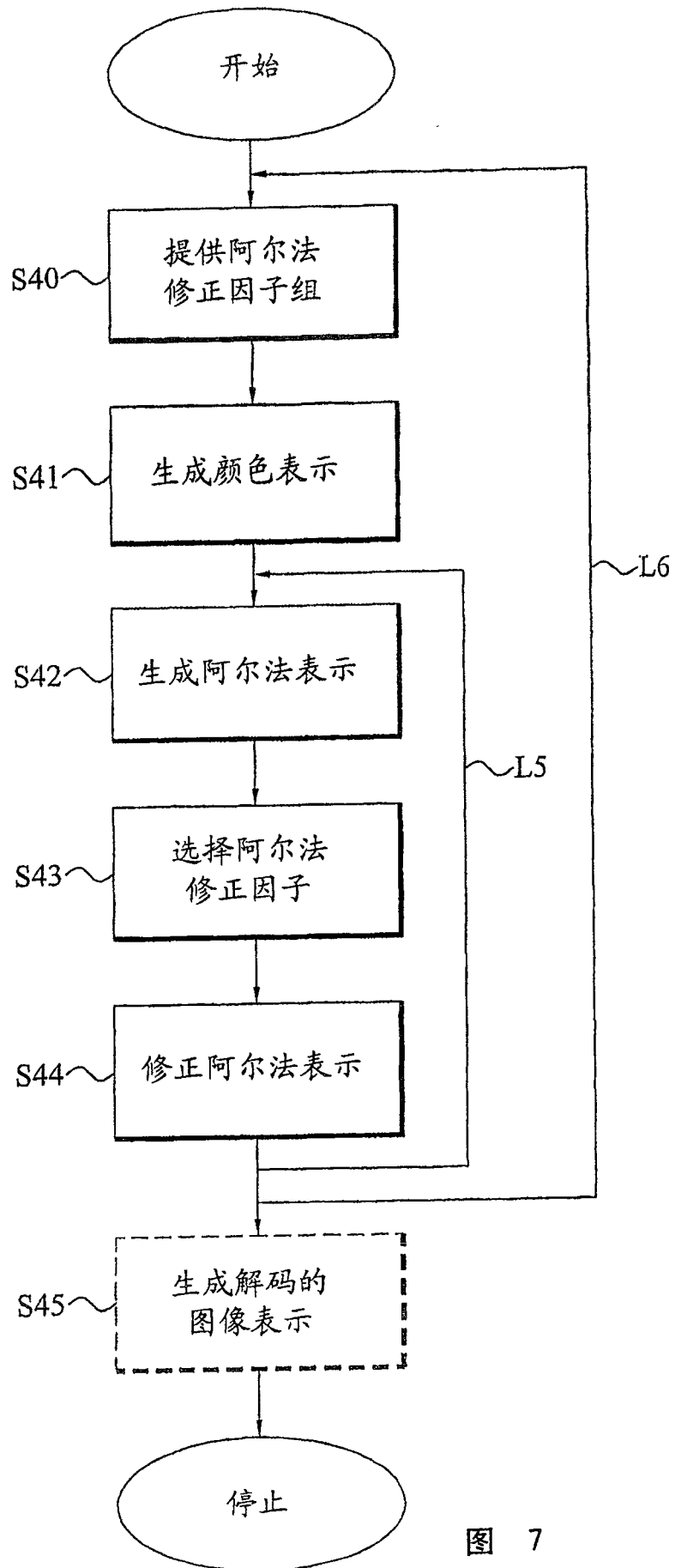


图 7

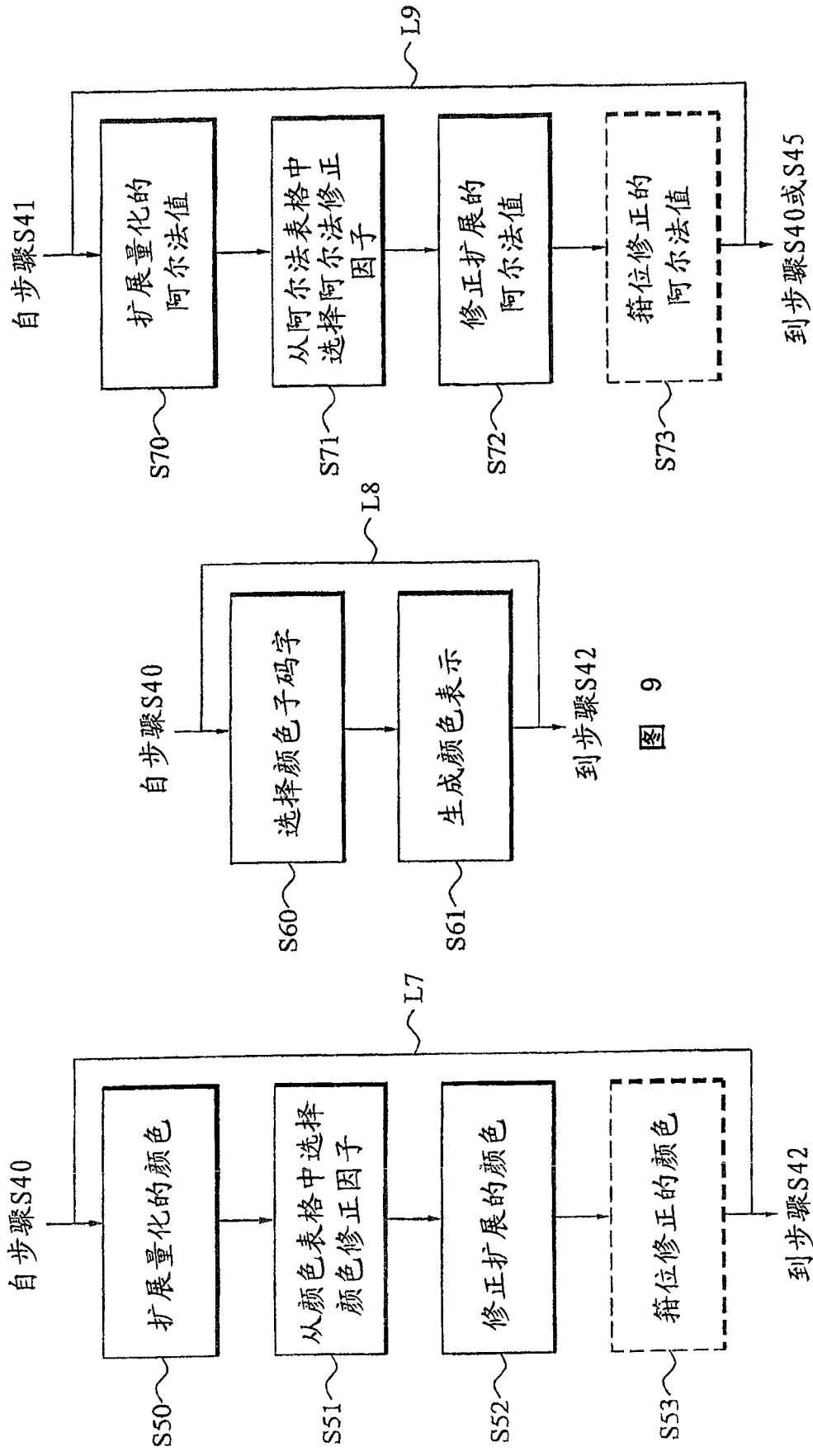


图 8

图 9

图 10

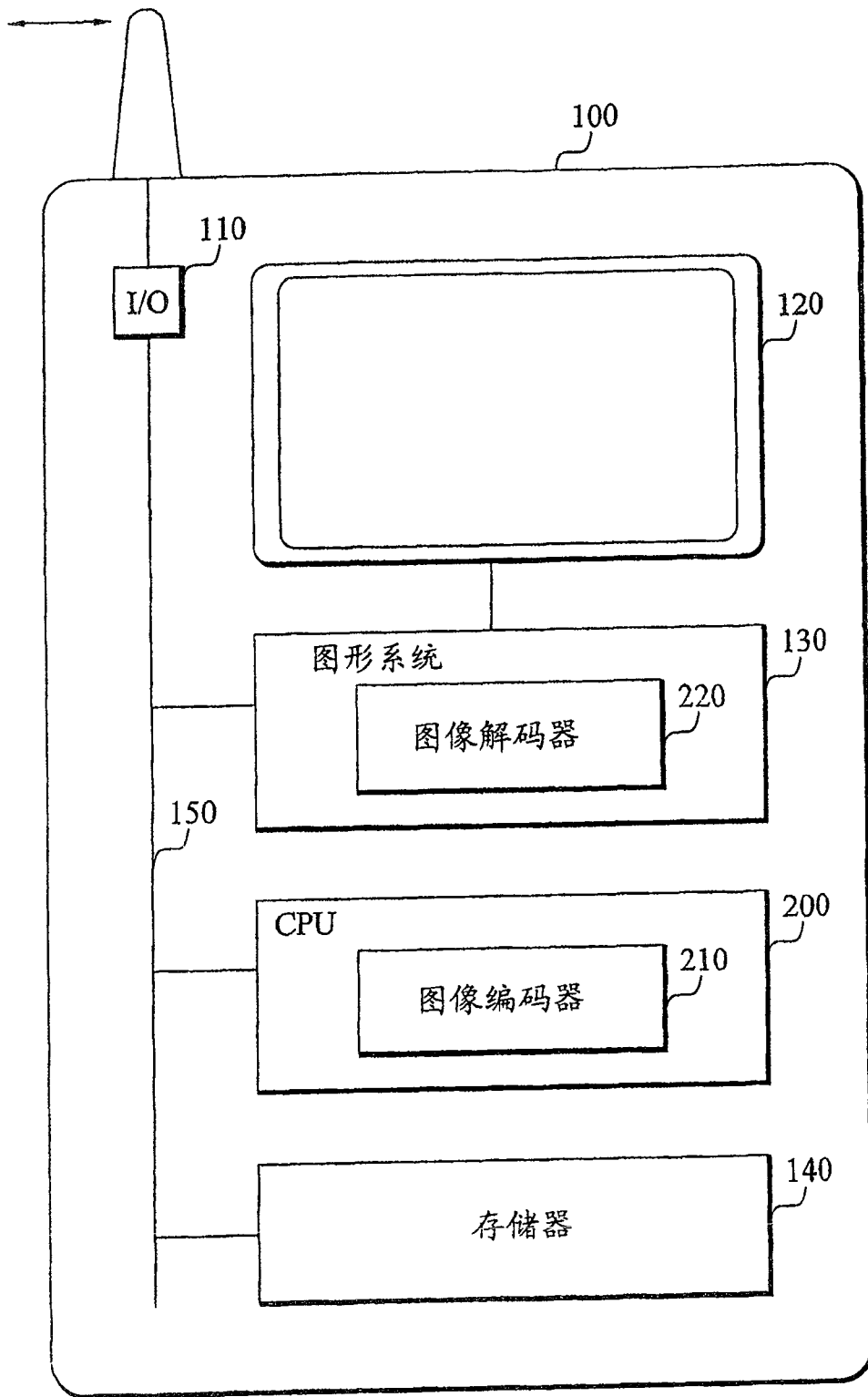


图 11

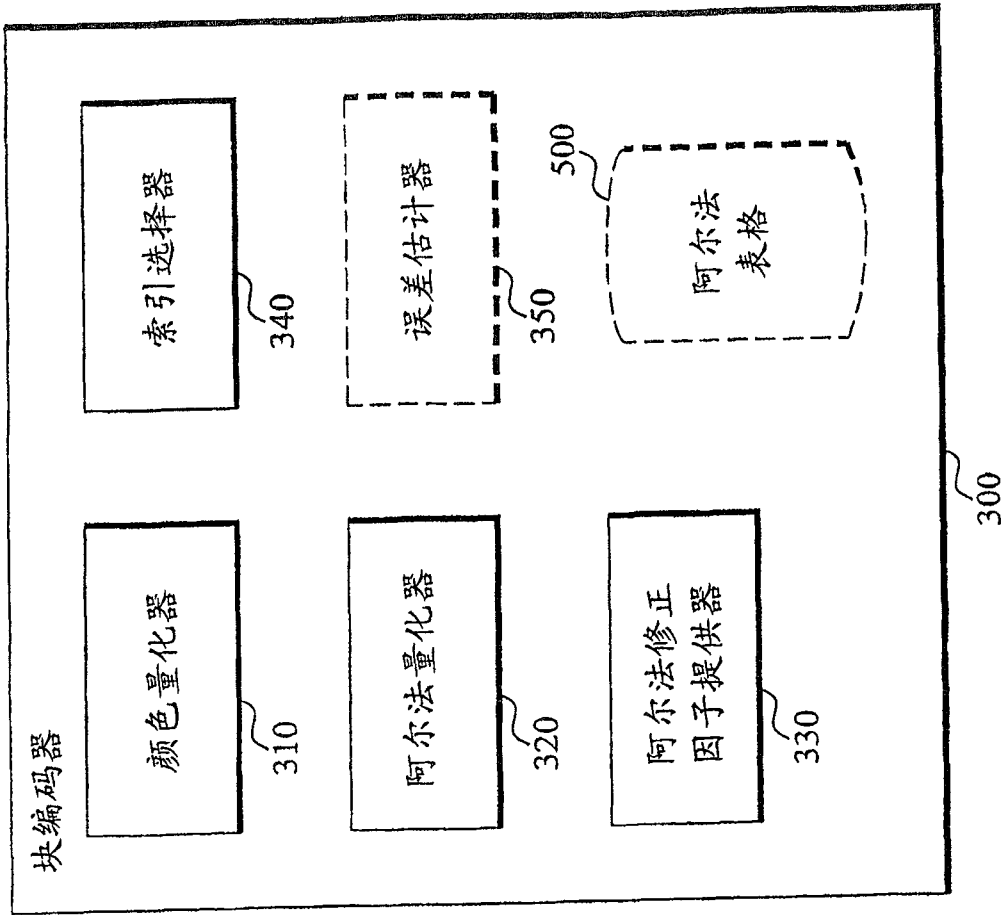


图 13

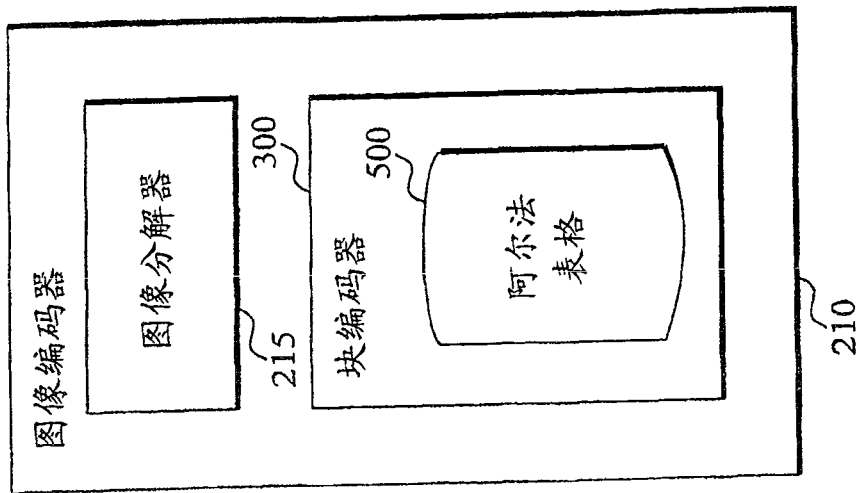


图 12

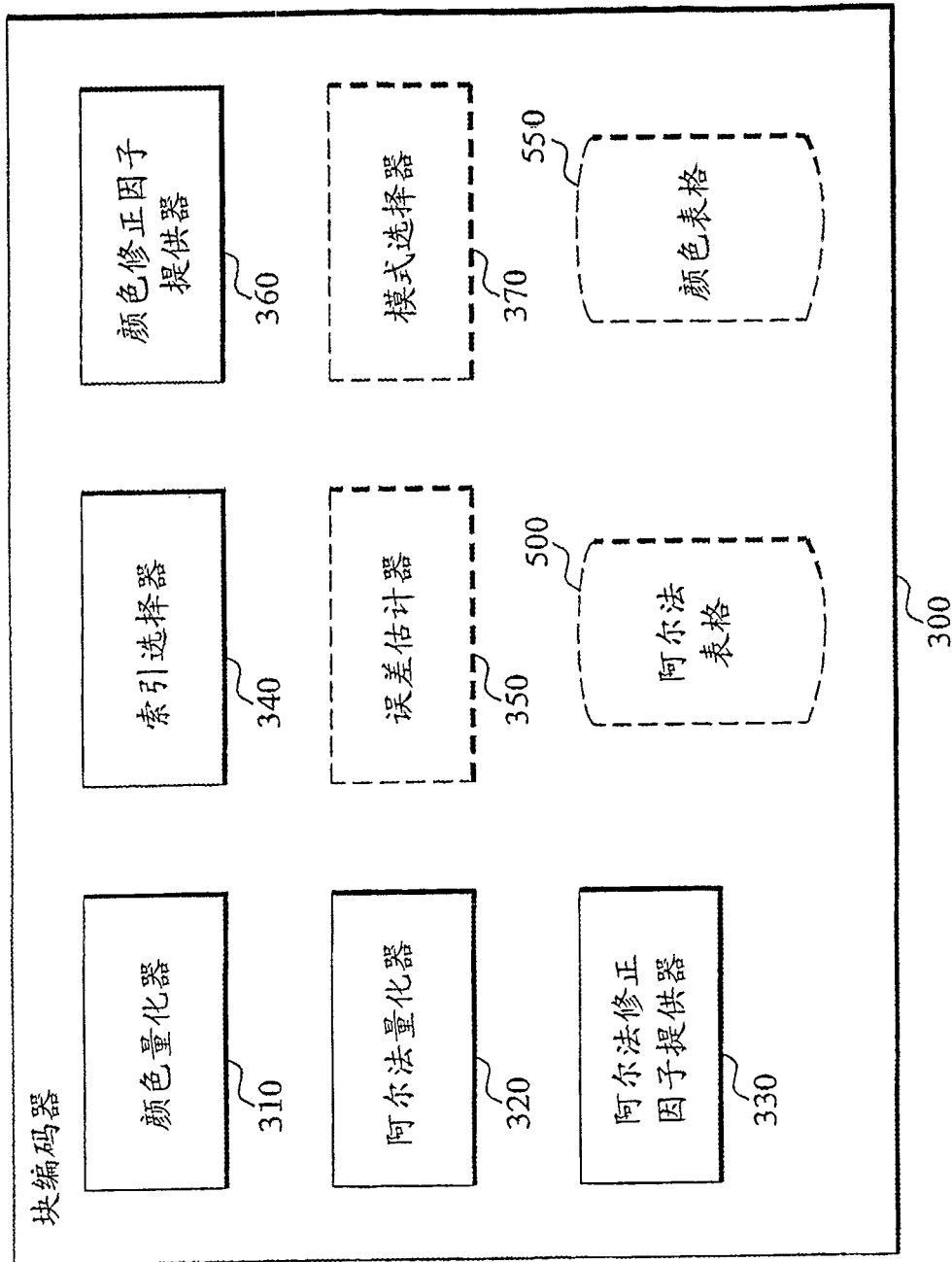


图 14

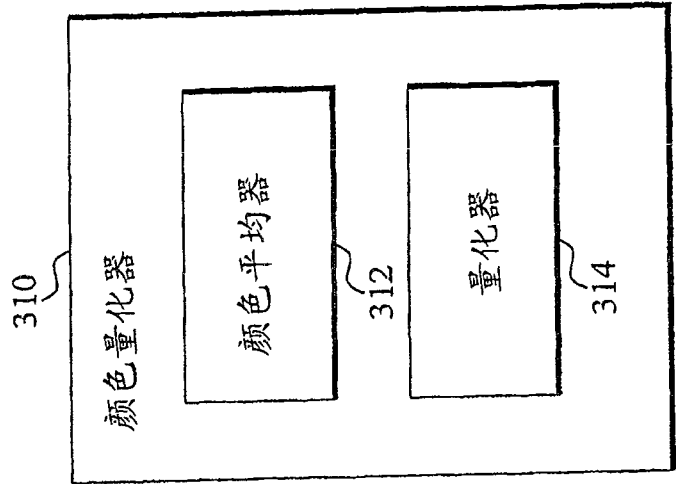


图 15

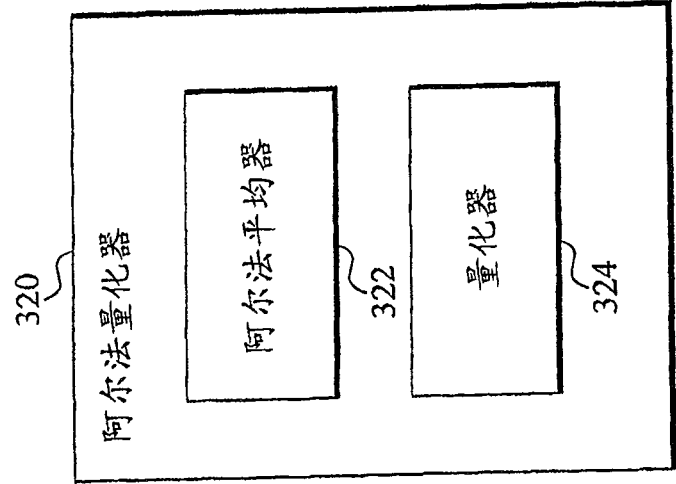


图 16

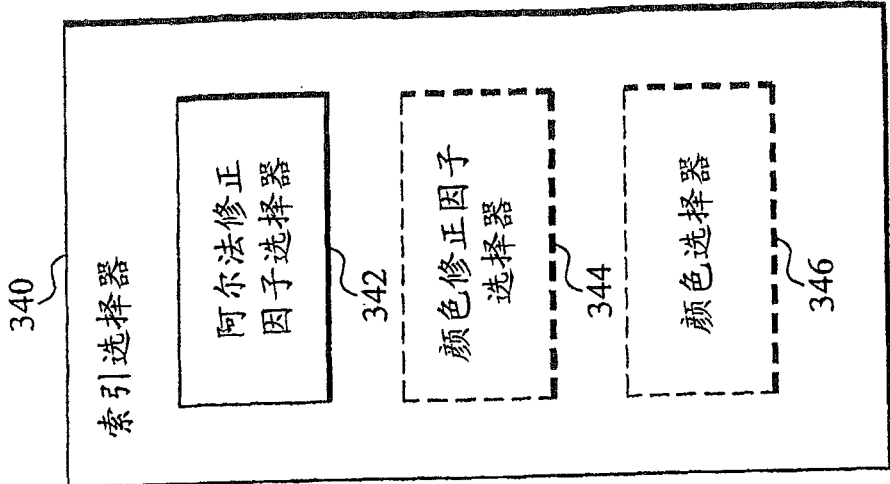


图 17

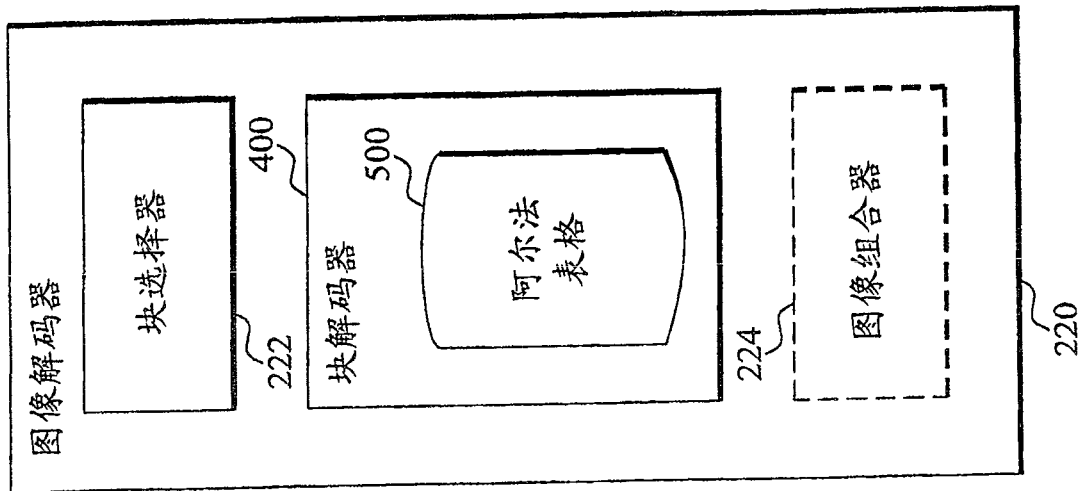


图 18

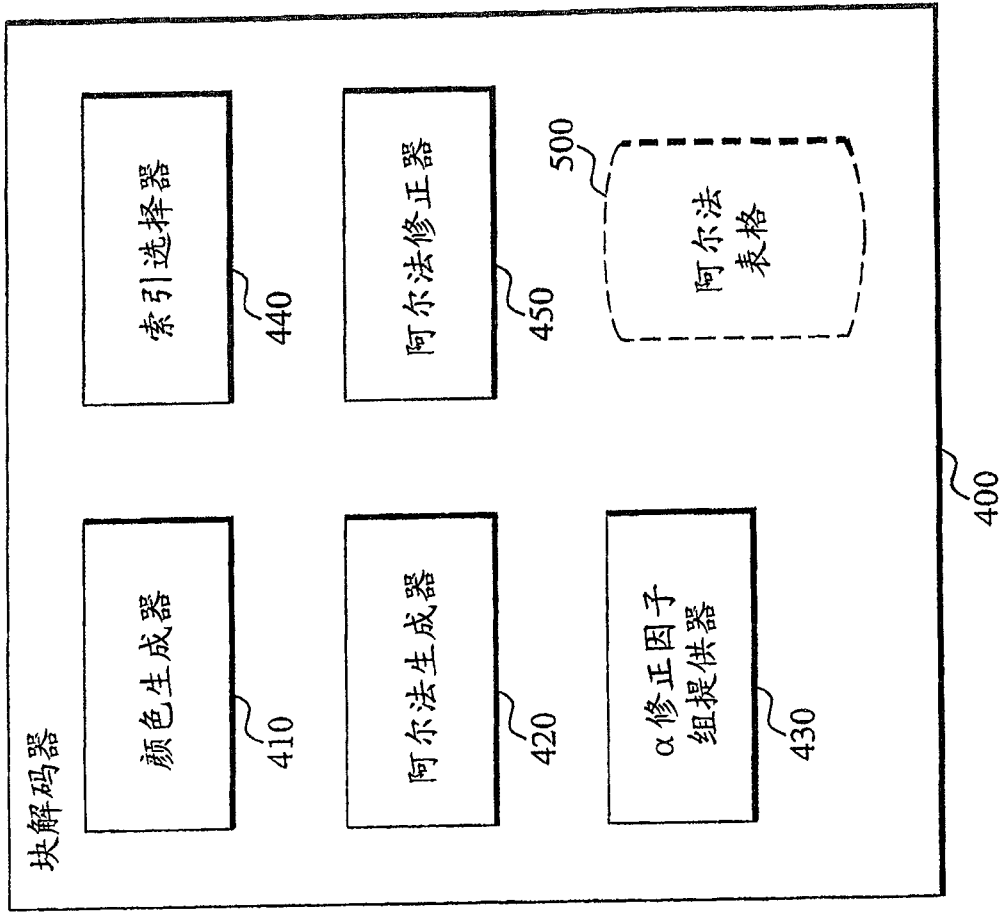


图 19

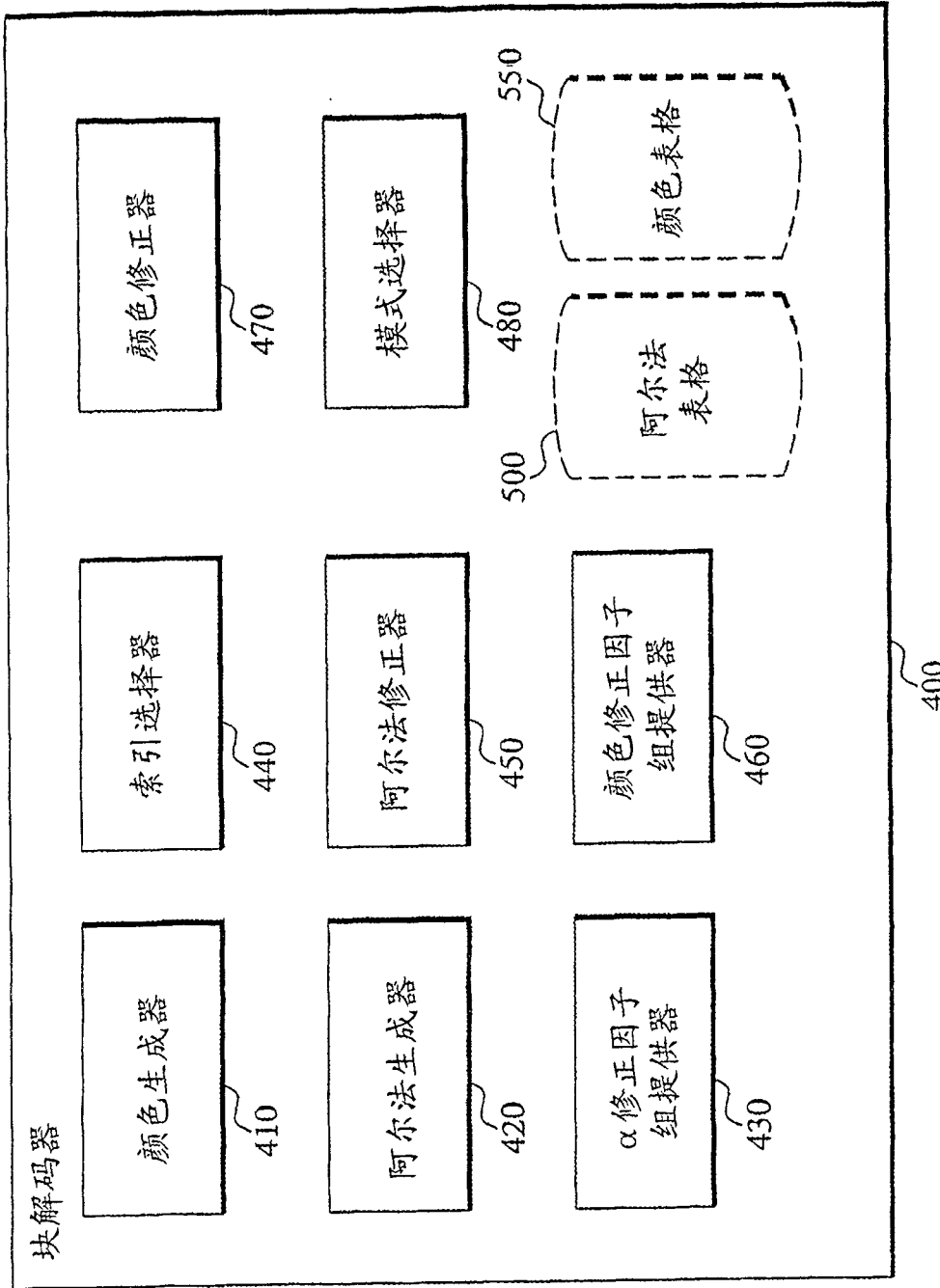


图 20

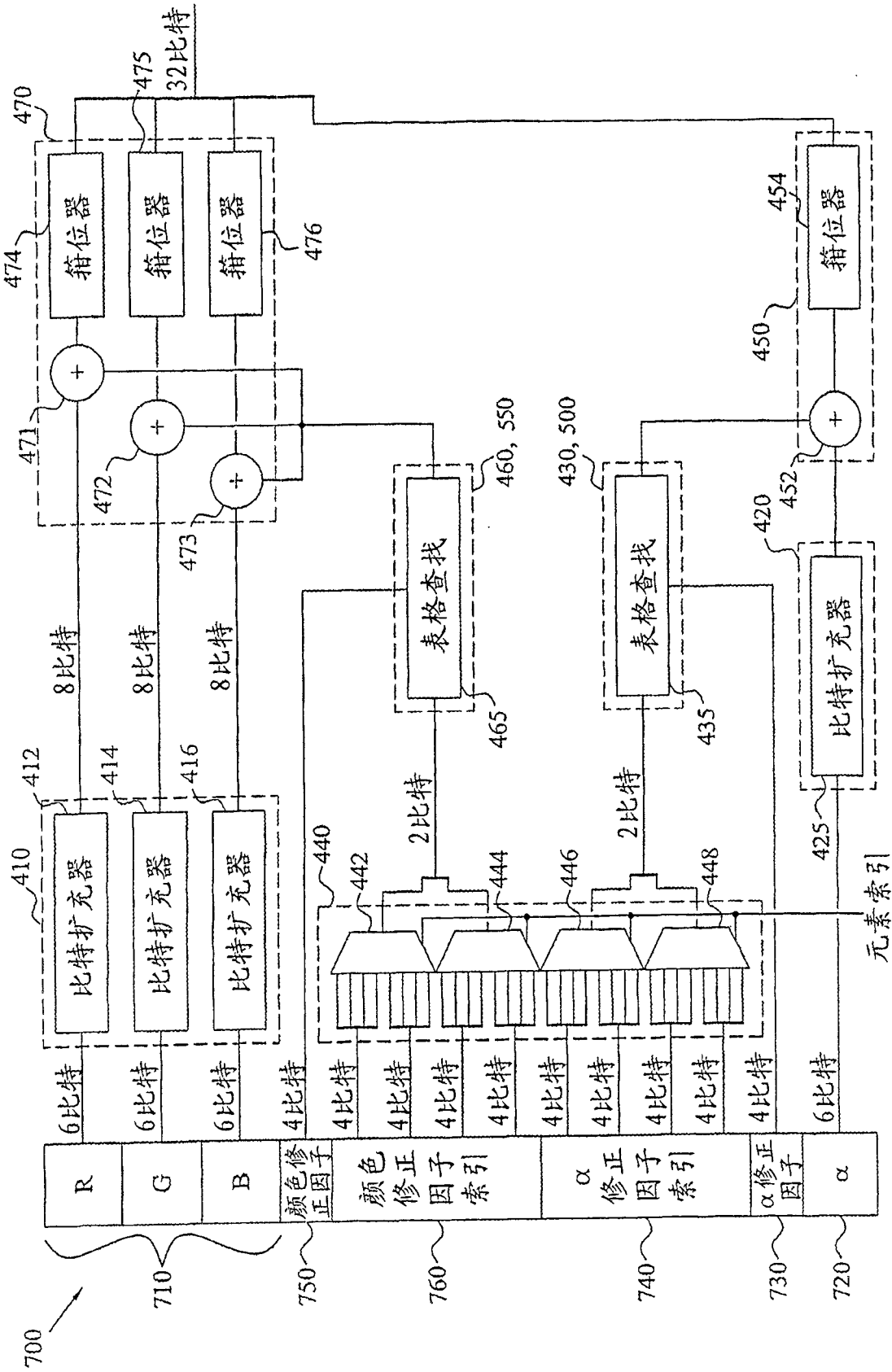


图 21

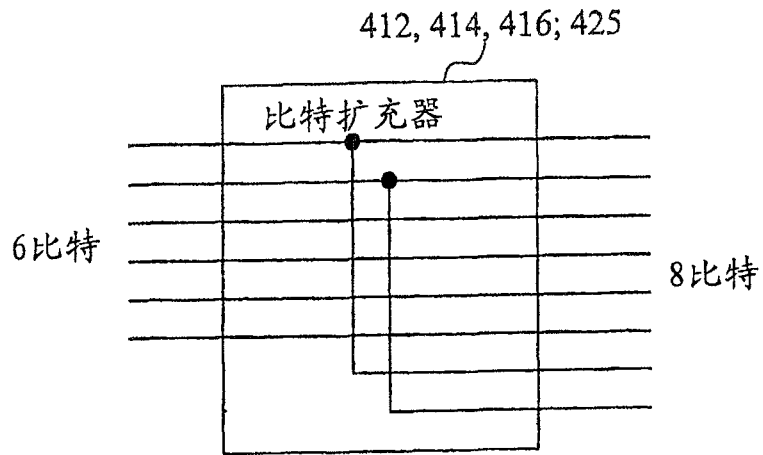


图 22

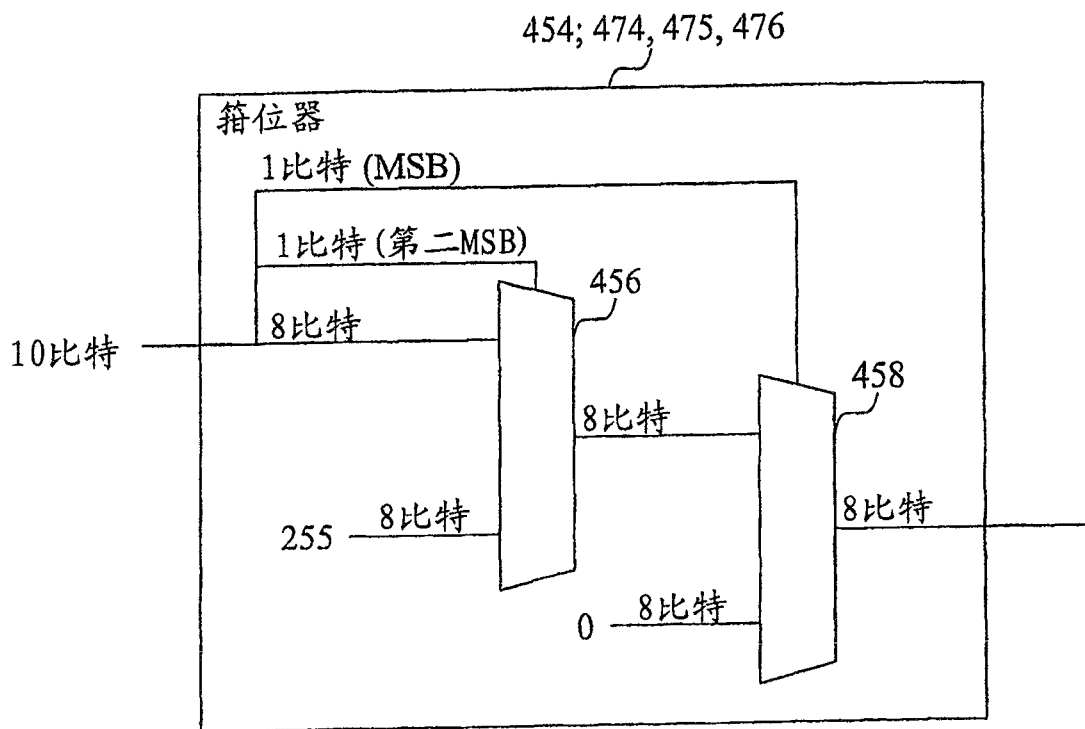


图 23

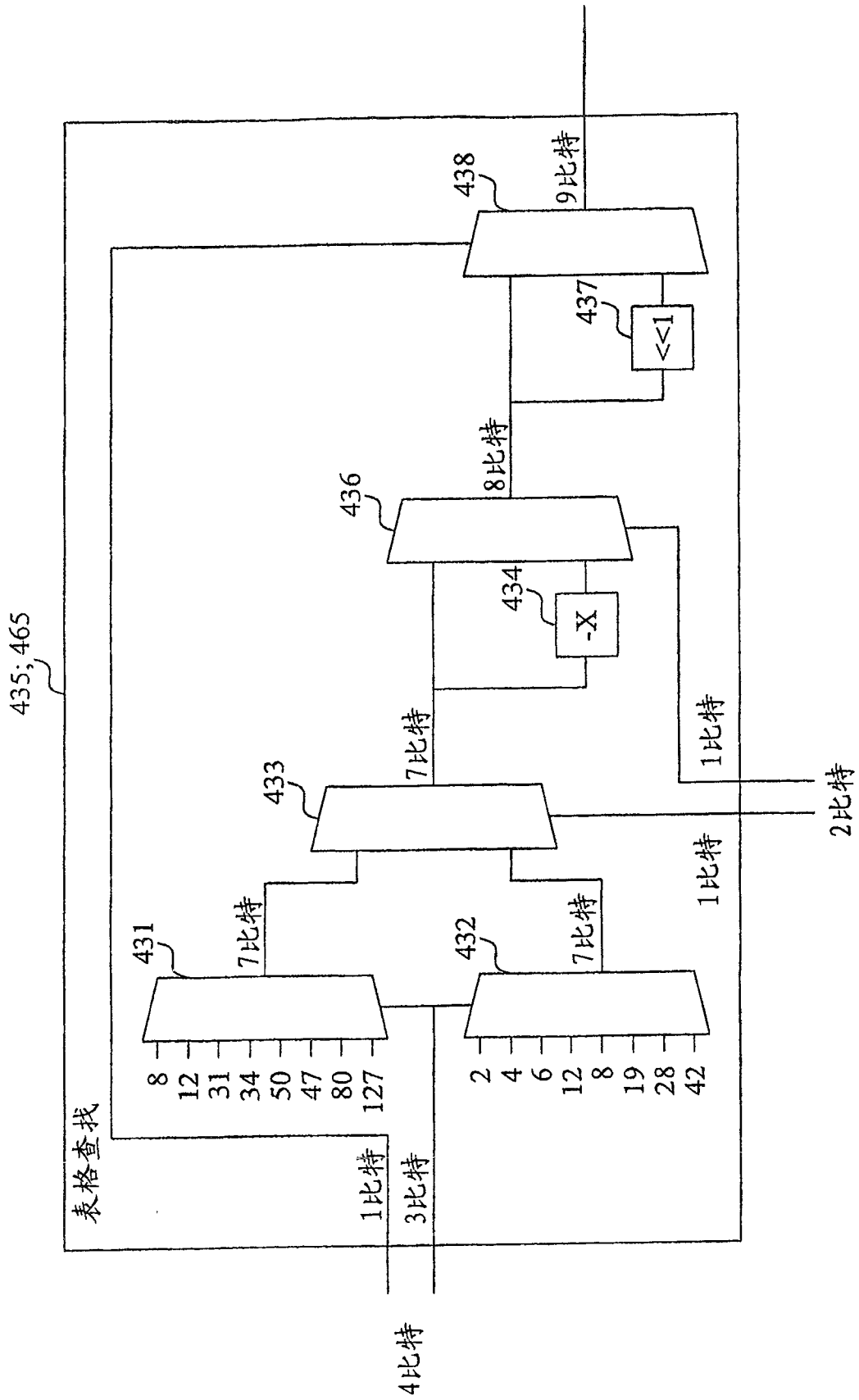


图 24