

[19] 中华人民共和国国家知识产权局

[51] Int. Cl.  
G06K 9/00 (2006.01)



# [12] 发明专利说明书

专利号 ZL 200510004183.5

[45] 授权公告日 2009年7月22日

[11] 授权公告号 CN 100517368C

[22] 申请日 2005.1.6

[21] 申请号 200510004183.5

[30] 优先权

[32] 2004.1.6 [33] US [31] 10/753,176

[73] 专利权人 微软公司

地址 美国华盛顿州

[72] 发明人 王 坚 陈黎勇 党映农

[56] 参考文献

US2002076080A1 2002.6.20

US6577748B2 2003.6.10

US6608942B1 2003.8.19

CN1359587A 2002.7.17

US6327395B1 2001.12.4

审查员 颜世莹

[74] 专利代理机构 上海专利商标事务所有限公司  
代理人 谢喜堂

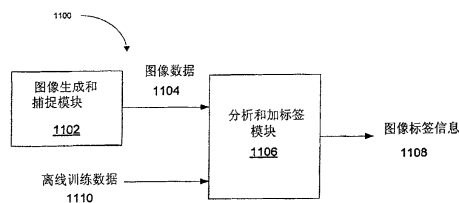
权利要求书 3 页 说明书 18 页 附图 18 页

[54] 发明名称

位置编码的文档图像分析和标签

[57] 摘要

本发明所揭示的实施例涉及分析包含诸如迷宫图案水印等经位置编码信息的文档图像，并基于诸如文本等文档内容遮蔽该经位置编码信息的程度来对该图像加标签。取决于这一遮蔽程度，可能无法从由照相机捕捉的文档图像中提取足够的位置编码比特来确定该由照相机捕捉的图像在文档内的位置。一分析和加标签模块接收由图像生成和捕捉模块输出的图像数据以及离线训练数据作为输入；执行分析和加标签处理；并输出图像标签信息。文档分析和加标签处理的结果可用于有效地确定由照相机捕捉的图像在经位置编码的文档内的位置。



1. 一种对包含迷宫图案的文档图像加标签以指示可从该图像中提取的迷宫图案量的方法，其特征在于，所述方法包括：

获取所述文档图像；以及

将所述图像加标签为至少选自以下的一种类型：包含迷宫图案并且不包含文档内容的第一类型、包含遮蔽所述迷宫图案的至少一部分的文档内容的第二类型，所述第二类型被划分成表示所述迷宫图案被所述文档内容遮蔽的不同程度的多个子类型，所述加标签包括：

对每一图像，对以该图像为中心的邻域窗中的完全可见的块的数量进行计数；以及

基于该数量对图像加标签。

2. 如权利要求1所述的方法，其特征在于，获取文档图像还包括：将一电子文档变为对应于印刷文档的位图表示。

3. 如权利要求1所述的方法，其特征在于，获取文档图像还包括：处理经扫描的纸件文档。

4. 一种对包含迷宫图案的由照相机捕捉的图像加标签以指示可从该图像中提取的迷宫图案量的方法，其特征在于，所述方法包括：

获取所述由照相机捕捉的图像；以及

将所述图像加标签为至少选自以下的一种类型：包含迷宫图案并且不包含文档内容的第一类型、以及包含遮蔽所述迷宫图案的至少一部分的文档内容的第二类型，所述第二类型被划分成表示所述迷宫图案被所述文档内容遮蔽的不同程度的多个子类型，所述加标签包括：

对每一图像，对以该图像为中心的邻域窗中的完全可见的块的数量进行计数；以及

基于该数量对图像加标签。

5. 如权利要求4所述的方法，其特征在于，使用一梯度图像直方图的支持间隔来确定所述图像是所述第一类型还是所述第二类型，所述支持间隔是沿所述直方图的x轴的具有非零值的最大数。

6. 如权利要求 5 所述的方法, 其特征在于, 它还包括: 向所述图像应用一梯度算子以获取一梯度图像。

7. 如权利要求 6 所述的方法, 其特征在于, 所述梯度算子是 Sobel 边算子。

8. 如权利要求 7 所述的方法, 其特征在于, 它还包括生成所述梯度图像的直方图。

9. 如权利要求 5 所述的方法, 其特征在于, 使用一离线训练时期和一在线加标签时期来确定所述图像是所述第一类型还是所述第二类型。

10. 如权利要求 9 所述的方法, 其特征在于, 用于在所述第一类型图像和所述第二类型图像之间进行区别的一阈值是基于在训练数据图像上执行的所述离线训练时期的结果来选择的。

11. 如权利要求 10 所述的方法, 其特征在于, 在所述在线加标签时期, 将所述阈值与所述图像的梯度图像直方图支持间隔相比较, 以确定所述图像是所述第一类型还是所述第二类型。

12. 一种对包含迷宫图案的文档图像加标签以指示可从该图像提取的迷宫图案量的系统, 其特征在于, 所述系统包括:

一图像生成和捕捉模块, 它获取所述文档图像; 以及

一分析和加标签模块, 它将所述图像加标签为至少选自以下的一种类型: 包含迷宫图案并不包含文档内容的第一类型、以及包含遮蔽所述迷宫图案的至少一部分的文档内容的第二类型, 所述第二类型被划分成表示所述迷宫图案被所述文档内容遮蔽的不同程度的多个子类型, 所述加标签包括:

对每一图像, 对以该图像为中心的邻域窗中的完全可见的块的数量进行计数; 以及

基于该数量对图像加标签。

13. 如权利要求 12 所述的系统, 其特征在于, 所述图像生成和捕捉模块将一电子文档变为对应于印刷文档的位图表示。

14. 如权利要求 12 所述的系统, 其特征在于, 所述图像生成和捕捉模块处理扫描的纸件文档。

15. 一种对包含迷宫图案的由照相机捕捉的图像加标签以指示可从所述图像提取的迷宫图案量的系统, 其特征在于, 所述系统包括:

一图像生成和捕捉模块，它获取所述由照相机捕捉的图像；以及

一分析和加标签模块，它将所述图像加标签为至少选自以下的一种类型：包含迷宫图案并不包含文档内容的第一类型、以及包含遮蔽所述迷宫图案的至少一部分的文档内容的第二类型，所述第二类型被划分成表示所述迷宫图案被所述文档内容遮蔽的不同程度的多个子类型，所述加标签包括：

对每一图像，对以该图像为中心的邻域窗中的完全可见的块的数量进行计数；以及

基于该数量对图像加标签。

16. 如权利要求 15 所述的系统，其特征在于，使用一梯度图像直方图的支持间隔来确定所述图像是所述第一类型还是所述第二类型，所述支持间隔是沿所述直方图的 x 轴的具有非零值的最大数。

17. 如权利要求 16 所述的系统，其特征在于，所述分析和加标签模块向所述图像应用一梯度算子来获取一梯度图像。

18. 如权利要求 17 所述的系统，其特征在于，所述梯度算子是 Sobel 边算子。

19. 如权利要求 16 所述的系统，其特征在于，所述分析和加标签模块生成所述梯度图像的直方图。

20. 如权利要求 15 所述的系统，其特征在于，所述分析和加标签模块执行一离线训练时期和一在线加标签时期，以确定所述图像是所述第一类型还是所述第二类型。

21. 如权利要求 20 所述的系统，其特征在于，用于在所述第一类型图像和所述第二类型图像之间进行区别的一阈值是基于在训练数据图像上执行的所述离线训练时期的结果来选择的。

22. 如权利要求 21 所述的系统，其特征在于，在所述在线时期，所述分析和加标签模块将所述阈值与所述图像的梯度图像直方图的支持间隔相比较，以确定所述图像是所述第一类型还是所述第二类型。

## 位置编码的文档图像分析和标签

### 技术领域

本发明的实施例涉及位置编码的媒质和数字笔之间的交互，尤其涉及基于文档内容遮蔽位置编码的信息的程度对包含位置编码的信息的文档图像加标签。

### 背景技术

计算机用户习惯于使用鼠标和键盘作为与个人计算机交互的一种方法。尽管个人计算机提供了优于书面文档的众多优点，然而大多数用户继续使用印刷的纸张来执行某些功能。这些功能的部分包括读和注释书面文档。在注释的情况下，由于由用户放置在其上的注释，印刷的文档呈现更大的意义。然而，令印刷的文档具有注释的困难之一是随后需要将注释输入回该文档的电子形式中。这需要原始用户或另一用户很辛苦地读完注释，然后将它们输入到个人计算机中。在某些情况下，用户将在注释和原始文本中扫描，由此来创建一个新文档。这多个步骤令印刷文档和文档的电子版本之间的交互在重复的基础上变得难以处理。此外，扫描的图像通常是不可修改的。没有方法将注释从原始文本中分离。这令使用注释变得困难。因此，需要一种处理注释的改进的方法。

捕捉手写信息的一种技术是通过使用其位置可在书写过程中确定的笔。提供这一功能的一种笔是 Anoto Inc 的 Anoto 笔。这一笔通过使用照相机捕捉用预定图案编码的纸件图像来起作用。该图像笔的一个示例在图 21 中示出。该图案由 Anoto 笔（Anoto Inc 生产）用于确定笔在一张纸上的位置。然而，用 Anoto 笔使用的系统来确定位置的效率究竟如何是不清楚的。为提供对已捕捉图像的位置的有效确定，需要一种提供捕捉图像的有效解码的系统。

当注释文档时，用户可通过相对于文档移动笔尖来对该文档加标签。笔尖的路径可包括多个笔画，其中，每一笔画对应于一系列被捕捉的图像。因此，希望有效地标识笔的路径以处理文档上的注释。

位置编码信息的一部分，如加水印的迷宫图案（maze pattern），可被诸如文本和/或图形等文档的内容遮蔽。当文档的内容遮蔽文档区域中相对少量（或没有）

位置编码的信息时,则该区域在文档内的位置可被有效地确定,而无需执行大量的计算来确定位置。然而,当文档的内容遮蔽文档区域中的大量位置编码的信息时,需要使用计算量更大的技术来确定该区域在文档内的位置。因此,需要将相对少量的位置编码信息被遮蔽的情况与相对大量的信息被遮蔽的情况区别的技术来提高用户与包含位置编码信息的文档进行交互的效率。

### 发明内容

本发明的实施例涉及分析包含像迷宫图案这样的位置编码信息,并对捕捉的图像加标签。这类文档分析的结果可用于有效地确定由照相机捕捉的图像在位置编码的文档内的位置。

一种系统包括一图像捕捉模块和一分析和加标签模块。该分析和加标签模块接收由图像捕捉模块输出的已捕捉图像的数据和离线训练数据作为输入;执行分析和加标签处理;并输出图像标签信息。

印刷的文档用迷宫图案来加水印。这类迷宫图案可以被诸如文本等文档内容所遮蔽。根据这类遮蔽的程度,可能无法从照相机捕捉的图像中提取足够的位置编码比特来确定照相机捕捉的图像在文档中的位置。

加水印的文档图像被分析并加标签。分析和加标签涉及基于迷宫图案在文档中的特定位置上的可见程度来确定加水印的文档内的 x-y 位置的能力。

阅读以下详细描述,可以清楚本发明的其它特征和优点。

### 附图说明

当接合附图阅读时,可以更好地理解以上本发明的概述以及以下较佳实施例的详细描述,附图包括示例而不是对本发明的局限。

图 1 示出了可用于本发明的实施例的计算机的通用描述。

图 2A 和 2B 示出了依照本发明的实施例的图像捕捉系统和对应的已捕捉图像。

图 3A 到 3F 示出了依照本发明的实施例的各种序列和折叠技术。

图 4A 到 4E 示出了依照本发明的实施例的各种编码系统。

图 5A 到 5D 示出了与依照图 4A 和 4B 的编码系统相关联的四个可能的结果角。

图 6 示出了依照本发明的实施例的已捕捉图像部分的旋转。

图 7 示出了结合图 4A 到 4E 的编码系统使用的各个旋转角度。

图 8 示出了依照本发明的实施例用于确定已捕捉数组的位置的过程。

图 9 示出了依照本发明的实施例用于确定已捕捉图像的位置的方法。

图 10 示出了依照本发明的实施例用于确定已捕捉图像的位置的另一方法。

图 11 描述了依照本发明的实施例的图像捕捉模块和分析加标签模块。

图 12 描述了用于生成加水印的灰度级文档图像的步骤，该图像可依照本发明的各种实施例来分析和加标签。

图 13 描述了文档内容的一部分的图像、位置编码的迷宫图案的一部分、组合的内容和迷宫图案、以及内容和迷宫图案的一部分的放大视图。

图 14 示出了图 13 所示的组合文档和迷宫图案的子窗以及子窗的中心像素。

图 15 示出了依照本发明的各种实施例的示例文档的文档分析结果。

图 16 描述了具有非零旋转角和 45 度旋转角的照相机捕捉的图像。

图 17 描述了形成 Sobel 边算子的两个卷积核。

图 18 示出了仅包含迷宫图案的已捕捉图像的梯度图像的直方图。

图 19 示出了包含文档内容的已捕捉图像的梯度图像的直方图。

图 20 示出了离线训练的一个示例结果，它包括一用于在纯迷宫图案图像和包含文档内容的图像之间进行区别的所选择的阈值。

图 21 示出了依照现有技术的文档中的编码空间的表示。

## 具体实施方式

本发明的各方面涉及确定已捕捉的图像相对于较大图像的位置。本发明描述的位置确定方法和系统可结合多功能笔使用。

为方便读者，以下内容由小标题分割。小标题包括：术语、通用计算机、图像捕捉笔、数组的编码、解码、纠错、位置确定、以及照相机捕捉的图像的分析加标签。

### I. 术语

笔—可包括或不包括储存墨水的能力的任何书写工具。在某些示例中，依照本发明的实施例，不储存墨水能力的触针可用做笔。

照相机—可从纸张或任何其它媒质捕捉图像的图像捕捉系统。

### II. 通用计算机

图 1 是可用于实现本发明的各方面的常规通用数字计算环境的一个示例的功能框图。在图 1 中，计算机 100 包括处理单元 110、系统存储器 120 和将包括系统存储器的各种系统组件耦合到处理单元 110 的系统总线 130。系统总线 130 可以是若干类型总线结构的任一种，包括存储器总线和存储器控制器、外围总线以及使用各种总线体系结构的任一种的局部总线。系统存储器 120 包括只读存储器（ROM）140 和随机存取存储器（RAM）150。

基本输入/输出系统 160（BIOS）包含如在启动时帮助在计算机 100 中的元件之间传输信息的基本例程，储存在 ROM 140 中。计算机 100 也包括用于对硬盘（未示出）进行读写的硬盘驱动器 170、用于对可移动磁盘 190 进行读写的磁盘驱动器 180、以及用于对诸如 CD ROM 或其它光媒质等可移动光盘 192 进行读写的光盘驱动器 191。硬盘驱动器 170、磁盘驱动器 180 和光盘驱动器 191 分别由硬盘驱动器接口 192、磁盘驱动器接口 193 和光盘驱动器接口 194 连接到系统总线 130。驱动器及其关联的计算机可读媒质为个人计算机 100 提供了计算机可读指令、数据结构、程序模块和其它数据的非易失存储。本领域的技术人员可以理解，在示例操作环境中也可使用可储存可由计算机访问的数据的其它类型的计算机可读媒质，如磁带盒、闪存卡、数字视频盘、贝努利盒式磁带、随机存取存储器（RAM）、只读存储器（ROM）等等。

众多程序模块可储存在硬盘 170、磁盘 190、光盘 192、ROM 140 或 RAM 150 中，包括操作系统 195、一个或多个应用程序 196、其它程序模块 197 和程序数据 198。用户可通过输入设备，如键盘 101 和定位设备 102 向计算机 100 输入命令和信息。其它输入设备（未示出）可包括麦克风、操纵杆、游戏垫、圆盘式卫星天线、扫描仪等等。这些和其它输入设备通常通过耦合到系统总线的串行端口接口连接到处理单元 110，但也可通过其它接口连接，如并行端口、游戏端口和通用串行总线（USB）。此外，这些设备可通过适当的接口（未示出）直接耦合至系统总线 130。监视器 107 或其它类型的显示设备也通过接口，如视频适配器 108 连接到系统总线 130。除监视器之外，个人计算机通常包括其它外围输出设备（未示出），如扬声器或打印机。在一个较佳的实施例中，提供了笔数字化仪 165 和所附的笔或触针 166，以便数字地捕捉手画线输入。尽管示出了笔数字化仪 165 和串行端口之间的直接连接，然而实际上，笔数字化仪 165 可以通过并行端口或其它接口和系统总线 130 直接耦合到处理单元 110，这在本领域中是已知的。此外，尽管示出数字化仪 165 与监视器 107 分离，然而较佳的是数字化仪 165 的可用输入区域可与监视器 107

的显示区域有相同的范围。此外，数字化仪 165 可被集成到监视器 107 中，或可作为单独的设备叠加或附加在监视器 107 上。

计算机 100 可在使用到一个或多个远程计算机，如远程计算机 109 的逻辑连接的网络化环境中操作。远程计算机 109 可以是服务器、路由器、网络 PC、对等设备或其它公用网络节点，并通常包括上述涉及计算机 100 的许多或所有元件，尽管图 1 中仅示出了存储器存储设备 111。图 1 所描述的逻辑连接包括局域网 (LAN) 112 和广域网 (WAN) 113。这类网络环境常见于办公室、企业范围计算机网络、内联网和因特网。

当在 LAN 网络环境中使用时，计算机 100 通过网络接口或适配器 114 连接到局域网 112。当在 WAN 网络环境中使用时，个人计算机 100 通常包括调制解调器 115 或其它装置，用于通过广域网 113，如因特网建立通信。调制解调器 115 可以是内置或外置的，通过串行端口接口 106 连接到系统总线 130。在网络化环境中，相对个人计算机 100 所描述的程序模块或其部分可储存在远程存储器存储设备中。

可以理解，所示的网络连接是说明性的，可以使用在计算机之间建立通信链路的其它技术。假定存在各种公知协议的任一种，如 TCP/IP、以太网、FTP、HTTP、蓝牙、IEEE 802.11x 等等，系统可在客户机—服务器配置中操作，以准许用户从基于 web 的服务器检索网页。可使用各种常规 web 浏览器的任一种来显示并操作网页上的数据。

### III. 图像捕捉笔

本发明的各方面包括将已编码的数据流置于表示该已编码的数据流的显示形式（例如，如后文参考图 4B 所讨论的，已编码数据流用于创建图形图案）。显示形式可以是印刷的纸张（或其它物理介质），或可以是结合另一图像或图像集来投影已编码数据流的显示屏。例如，已编码数据流可被表示为纸张上的具体图形图像或叠加在显示的图像（如表示文档的文本）上的图形图像，或可以是显示屏上的具体（不可修改）图形图像（因此，由笔捕捉的任何图像部分都位于该显示屏上）。

已捕捉图像的位置的确定可用于确定用户与纸张、介质或显示屏交互的位置。在本发明的某些方面，笔可以是在纸张上书写的墨水笔。在其它方面，笔可以是用户在计算机显示屏的表面上书写的触针。交互活动可返回给已知文档上的已编码图像或支持计算机屏幕上显示的文档的系统。通过当笔或触针遍历文档时用笔或触针中的照相机重复地捕捉图像，系统可跟踪由用户操纵的触针的移动。显示的或印刷

的图像可以是与空白或内容丰富纸张相关联的水印,或可以是与叠加在屏幕上或内建到屏幕中的显示图像或固定编码相关联的水印。

图 2A 和 2B 示出了具有照相机 203 的笔 201 的一个说明性示例。笔 201 包括笔尖 202, 它可包括或不包括墨水贮液器。照相机 203 从表面 207 捕捉图像 204。笔 201 还可包括另外的传感器和/或处理器, 如由虚线框 206 所表示的。这些传感器和/或处理器 206 也可包括向另一笔 201 和/或个人计算机发送信息的能力(例如, 通过蓝牙或其它无线协议)。

图 2B 表示由照相机 203 察看的图像。在一个说明性示例中, 照相机 203 的可见区(即, 照相机的图像传感器的分辨率)是  $32 \times 32$  像素(其中,  $N=32$ )。在一个实施例中, 已捕捉图像( $32$  像素  $\times$   $32$  像素)对应于由照相机 203 捕捉的表面平面的约  $5\text{mm} \times 5\text{mm}$  的区域。因此, 图 2B 示出了  $32$  像素长  $\times$   $32$  像素宽的可见区。 $N$  的大小是可调节的, 使得较大的  $N$  对应于更高的图像分辨率。而且, 尽管此处为说明目的, 示出照相机 203 的可见区是正方形, 然而, 本领域中已知, 可见区可包括其它形状。

由照相机 203 捕捉的图像可被定义为图像帧序列  $\{I_i\}$ , 其中,  $I_i$  由笔 201 在采样时刻  $t_i$  捕捉。采样率可大可小, 取决于系统配置和性能要求。已捕捉图像帧的大小可大可小, 取决于系统配置和性能要求。

由照相机 203 捕捉的图像可由处理系统直接使用, 或可经历预过滤。这一预过滤可在笔 201 中发生, 或可在笔 201 之外发生(例如, 在个人计算机中)。

图 2B 的图像大小是  $32 \times 32$  像素。如果每一编码单元大小是  $3 \times 3$  像素, 则已捕捉的已编码单元的数量约为 100 单元。如果编码单元大小是  $5 \times 5$  像素, 则已捕捉的已编码单元的数量约为 36 单元。

图 2a 也示出了图像平面 209, 在其上形成来自位置 204 的图案的图像 210。从对象平面 207 的图案接收的光可由透镜 208 聚焦。透镜 208 可以是单个透镜或多部分透镜系统, 但是为简明起见, 在此处表示为单个透镜。图像捕捉传感器 211 捕捉图像 210。

图像传感器 211 可以足够大来捕捉图像 210。可选地, 图像传感器 211 可以足够大来捕捉位置 212 处的笔尖 202 的图像。仅供参考, 位置 212 处的图像被称为虚拟笔尖。注意, 由于笔尖、透镜 208 和图像传感器 211 之间的恒定关系, 相对于图像传感器 211 的虚拟笔尖位置是固定的。

以下变换  $F_{S \rightarrow p}$  将由照相机捕捉的图像中的位置坐标转换成纸张上的真实图像

的位置坐标：

$$L_{\text{纸张}} = F_{S \rightarrow P}(L_{\text{传感器}})$$

在书写的过程中，笔尖和纸张在同一平面上。因此，从虚拟笔尖到真实笔尖的变换也是  $F_{S \rightarrow P}$ ：

$$L_{\text{笔尖}} = F_{S \rightarrow P}(L_{\text{虚拟笔尖}})$$

变换  $F_{S \rightarrow P}$  可以被估算为一仿射变换。它被简化为：

$$F'_{S \rightarrow P} = \left\{ \begin{array}{ccc} \frac{s_x \sin \theta_y}{\cos \theta_x \sin \theta_y - \cos \theta_y \sin \theta_x}, & -\frac{s_x \cos \theta_y}{\cos \theta_x \sin \theta_y - \cos \theta_y \sin \theta_x}, & 0 \\ \frac{s_y \sin \theta_x}{\cos \theta_x \sin \theta_y - \cos \theta_y \sin \theta_x}, & \frac{s_y \cos \theta_x}{\cos \theta_x \sin \theta_y - \cos \theta_y \sin \theta_x}, & 0 \\ 0, & 0, & 1 \end{array} \right\}$$

作为  $F_{S \rightarrow P}$  的估算，其中， $\theta_x$ ， $\theta_y$ ， $s_x$  和  $s_y$  是在位置 204 处捕捉的图案的两个方向的旋转和比例。此外，可通过将已捕捉图像与纸张上对应的真实图像进行匹配来细化  $F'_{S \rightarrow P}$ 。“细化”指通过被称为递归方法的一种类型的优化算法来获取变换  $F_{S \rightarrow P}$  的更精确的估算。该递归方法将矩阵  $F'_{S \rightarrow P}$  作为初始值来处理。经细化的估算更精确地描述了  $S$  和  $P$  之间的转换。

下一步，可通过校准来确定虚拟笔尖的位置。

将笔尖 202 置于纸张上的固定位置  $L_{\text{笔尖}}$  上。接着倾斜该笔，从而允许照相机 203 用不同的笔姿势来捕捉一系列图像。对于捕捉到的每一图像，可获得变换  $F_{S \rightarrow P}$ 。从该变换，可获得虚拟笔尖的位置  $L_{\text{虚拟笔尖}}$ ：

$$L_{\text{虚拟笔尖}} = F_{P \rightarrow S}(L_{\text{笔尖}})$$

其中， $L_{\text{笔尖}}$  被初始化为  $(0, 0)$ ，并且

$$F_{P \rightarrow S} = (F_{S \rightarrow P})^{-1}$$

通过对从每一图像获得的  $L_{\text{虚拟笔尖}}$  求平均值，可确定虚拟笔尖的位置  $L_{\text{虚拟笔尖}}$ 。采用  $L_{\text{虚拟笔尖}}$ ，可获得  $L_{\text{笔尖}}$  的更准确的估算。在若干次迭代之后，可确定虚拟笔尖的准确位置  $L_{\text{虚拟笔尖}}$ 。

虚拟笔尖的位置  $L_{\text{虚拟笔尖}}$  现在已知。也可从捕捉的图像获得变换  $F_{S \rightarrow P}$ 。最后，可使用该信息来确定真实笔尖的位置  $L_{\text{笔尖}}$ ：

$$L_{\text{笔尖}} = F_{S \rightarrow P}(L_{\text{虚拟笔尖}})$$

#### IV. 数组的编码

可通过折叠一维序列来构造二维数组。包含足够多数量比特的二维数组的任何部分可用于确定其在整个二维数组中的位置。然而，有必要从一个已捕捉的图像或几个已捕捉的图像确定位置。为将已捕捉的图像部分与二维数组中的两个或多个位置相关联的概率最小化，可使用一非重复序列来创建该数组。所创建的序列的一个属性是该序列不在一段长度（或窗） $n$ 上重复。下文描述了一维序列的创建，随后将序列折叠成数组。

#### IV.A. 序列构造

数字序列可用做编码系统的起始点。例如，序列（也称为  $m$  序列）可被表示为域  $F_q$  中的  $q$  元素集。此处， $q = p^n$ ，其中， $n \geq 1$ ，并且  $p$  是质数。序列或  $m$  序列可通过各种不同的技术来生成，包括但不限于，多项式除法。使用多项式除法，序列可被定义如下：

$$\frac{R_l(x)}{p_n(x)}$$

其中， $P_n(x)$  是在域  $F_q[x]$ （具有  $q^n$  个元素）中次数为  $n$  的本原多项式。 $R_l(x)$  是域  $F_q[x]$  中次数为  $l$ （其中， $l < n$ ）的非零多项式。序列可使用一迭代过程来创建，它具有两个步骤：首先，将两个多项式相除（得到域  $F_q$  中的一个元素），其次，将余项乘以  $x$ 。当输出开始重复时，停止计算。该过程可使用一线性反馈移位寄存器来实现，如 Douglas W. Clark 和 Lih-Jyh Weng 的文章“最大和近最大移位寄存器序列：有效的事件寄存器和简单的离散对数（Maximal and Near-Maximal Shift Register Sequences: Efficient Event Counters and Easy Discrete Logarithms）”中所描述的，该文发表在 IEEE 计算机学报 43.5（IEEE Transactions on Computers 43.5）（1994 年 5 月，560-568 页）。在这一实施例中，在序列的循环移位和多项式  $R_l(x)$  之间建立一种关系：改变  $R_l(x)$  仅对该序列循环移位，并且每次循环移位对应于一个多项式  $R_l(x)$ 。所得的序列的一个性质是，该序列具有  $q^n - 1$  的周期，并且在一个月周期内，在宽度（或长度） $n$  上，任何部分在该序列中存在一次且仅一次。这被称为“窗特性”。周期  $q^n - 1$  也被称为序列的长度， $n$  被称为序列的阶。

上述过程仅是可用于创建具有窗特性的序列的各种过程的其中一个。

#### IV.B. 数组构造

可用于创建图像（其一部分可由照相机捕捉）的数组（或  $m$  数组）是一维序

列或  $m$  序列的扩展。设  $A$  为周期  $(m_1, m_2)$  的数组，即  $A(k+m_1, l)=A(k, l+m_2)=A(k, l)$ 。当  $n_1 \times n_2$  的窗通过  $A$  的周期移位时， $F_q$  上所有的非零  $n_1 \times n_2$  矩阵出现一次且仅一次。这一特性也被称为“窗特性”，因为每一窗是唯一的。窗然后可被表达为周期  $(m_1, m_2)$  和阶  $(n_1, n_2)$  的数组（ $m_1$  和  $m_2$  是数组中存在的水平和垂直比特数）

可通过折叠序列来构造二进制数组（或  $m$  数组）。一种方法是获得序列，然后将其折叠成大小为  $m_1 \times m_2$ ，其中，该数组的长度为  $L=m_1 \times m_2=2^n-1$ 。可选地，可以想要覆盖的空间的预定大小开始（例如，一张纸、30 张纸，或计算机监视器的大小），确定区域  $(m_1 \times m_2)$ ，然后使用该尺寸以使  $L \geq m_1 \times m_2$ ，其中， $L=2^n-1$ 。

可使用各种不同的折叠技术。例如，图 3A 到图 3C 示出了三个不同的序列。其每一个可被折叠成如图 3D 所示的数组。这三个不同的折叠方法被示出为图 3D 中的覆盖图，以及图 3E 和 3F 中的光栅路径。采用图 3D 所示的折叠方法。

为创建如图 3D 所示的折叠方法，创建长度为  $L$  且阶为  $n$  的序列  $\{a_i\}$ 。下一步，通过使数组的每一比特按以下公式 1 计算，从序列  $\{a_i\}$  创建大小为  $m_1 \times m_2$  的数组  $\{b_{kl}\}$ ，其中， $\gcd(m_1, m_2)=1$ ，并且  $L=m_1 \times m_2$ ：

$$b_{kl}=a_i, \text{ 其中, } k=i \bmod(m_1), l=i \bmod(m_2), i=0, \dots, L-1. \quad (1)$$

该折叠方法可被可选地表达为将序列放置在数组的对角线上，然后当达到边时从对边开始继续。

图 4A 示出了可用于对图 3D 的数组进行编码的示例编码技术。可以理解，也可使用其它编码技术。例如，图 11 示出了一个替换编码技术。

参考图 4A，第一比特 401（例如，“1”）由深色墨水的列来表示。第二比特 402（例如，“0”）由深色墨水的行来表示。可以理解，可使用任何颜色的墨水来表示各种比特。对所选择的墨水颜色的唯一要求是它提供了与介质的背景的显著对比，以便可由图像捕捉系统来区分。图 4A 中的比特由单元的  $3 \times 3$  的矩阵来表示。该矩阵的大小可被修改成任何大小，如基于图像捕捉系统的大小和分辨率。比特 0 和 1 的替换表示在图 4C-4E 中示出。可以理解，图 4A-4E 的示例编码的 1 或 0 的表示可以交换，而没有任何影响。图 4C 以隔行排列示出了占据两行或两列的比特表示。图 4D 以虚线形式示出了行和列中的像素的替换排列。最后，图 4E 以不规则间隔格式（例如，两个深色点之后为一个空白点）示出了列和行中的像素表示。

再次参考图 4A，如果比特由  $3 \times 3$  的矩阵表示，并且成像系统检测到  $3 \times 3$  区域中的一个深色行和两个空白行，则检测到 0（或 1）。如果检测到图像具有一个深色列和两个空白列，则检测到 1（或 0）。

此处，使用了一个以上像素或点来表示一个比特。使用单个像素（或点）来表示一个比特是脆弱的。灰尘、纸张中的折痕、不平坦表面等等会造成读取数据单元的单个比特表示的困难。然而，可以理解，可使用不同的方法来图形地表示表面上的数组。某些方法在图 4C 到 4E 中示出。可以理解，也可使用其它方法。一种方法在图 11 中仅使用间隔移位的点来描述。

使用比特流来创建图 4B 的图形图案 403。图形图案 403 包括 12 行和 18 列。行和列由比特流形成，该比特流使用比特表示 401 和 402 被转化成图形表示。图 4B 可被视为具有以下比特表示：

$$\begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

## V. 解码

当一个人用图 2A 的笔书写时，或将笔移动到靠近已编码的图案时，照相机捕捉图像。例如，当笔 201 被压向纸张时，且笔 201 遍历纸张上的文档时，笔 201 可使用压力传感器。然后处理图像来确定已捕捉的图像相对于已编码图像的完整表示的方向，并提取构成已捕捉图像的比特。

为了确定已捕捉图像相对于整个已编码区域的方向，可注意到，并非图 5A-5D 所示的所有四个可想象的角在图形图案 403 中都存在。实际上，采用了正确的方向，图 5A 所示的角的类型无法在图形图案 403 中存在。因此，缺少图 5A 所示的角类型的方向是正确的方向。

继续到图 6，可分析由照相机 601 捕捉的图像，并确定其方向，以便可对由图像 601 实际表示的位置进行解释。首先，审阅图像 601 以确定旋转该图像使得像素在水平和垂直上对齐所需的角度  $\theta$ 。注意，替换的网格对齐也是可能的，包括底层网格到非水平和垂直排列（如，45 度）的旋转。使用非水平和垂直排列可提供消除来自用户的视觉分心的可能益处，因为用户往往在其它图案之前注意到水平和垂直图案。为简明目的，网格的方向（水平和垂直，以及底层网格的任何其它旋转）被总称为预定义网格方向。

下一步，分析图像 601 以确定缺少了那个角。将图像 601 旋转到可用于解码的图像 603 所需的旋转量  $\alpha$  被示出为  $\alpha = (\theta \text{ 加上旋转量 } \{ \text{由缺少哪个角来定义} \})$ 。该

旋转量由图 7 中的公式示出。再次参考图 6，首先通过像素的布局来确定角度  $\theta$ ，以到达像素的水平和垂直（或其它预定义网格方向）排列，并且如 602 所示地旋转图像。然后进行分析以确定缺少的角，并且将图像 602 旋转到图像 603，以建立用于解码的图像。此处，图像被逆时针旋转 90 度，使得图像 603 具有正确的方向，并可用于解码。

可以理解，旋转角度  $\theta$  可在图像 601 的旋转之前或之后应用，以考虑丢失的角。也可理解，通过考虑已捕捉图像中的噪声，所有四个类型的角都可存在。可计算每一类型的角的数量，并选择具有最少数量的类型作为所缺少的角类型。

最后，读出图像 603 中的代码，并将其与用于创建图像 403 的原始比特流相关。相关可以众多方式执行。例如，它可通过递归方式执行，其中，将恢复的比特流与原始比特流内的所有其它比特流片段进行比较。其次，可在恢复的比特流和原始比特流之间执行统计分析，例如使用两个比特流之间的汉明距离来执行。可以理解，可使用各种方法来确定恢复的比特流在原始比特流中的位置。

一旦得到了恢复的比特，需要在原始数组（例如，图 4B 所示的）中查找已捕捉图像。确定比特分段在整个数组中的位置的这一过程可被若干项目变得更复杂。首先，可遮蔽要被捕捉的实际比特（例如，照相机可捕捉具有遮蔽原始代码的手写体的图像）。其次，灰尘、折痕、反射等等也可在已捕捉图像中引入误差。这些误差可令查找过程变得更困难。在这一点上，图像捕捉系统可能需要用从该图像提取的非顺序比特来运作。下文示出了用来自图像的非顺序比特来操作的方法。

设序列（或  $m$  序列） $\mathbf{I}$  对应于幂级数  $I(x)=1/P_n(x)$ ，其中， $n$  是  $m$  序列的阶，并且已捕捉的图像包含  $\mathbf{Ib}=(b_0 \ b_1 \ b_2 \ \cdots \ b_{K-1})^t$  的  $K$  个比特，其中  $K \geq n$ ，并且上标  $t$  表示矩阵或向量的转置。 $K$  个比特的位置  $s$  正好是  $\mathbf{I}$  的循环移位的次数，使得  $b_0$  被移位到序列的起始。然后，该移位的序列  $\mathbf{R}$  对应于幂级数  $x^s/P_n(x)$ ，即  $\mathbf{R}=T^s(\mathbf{I})$ ，其中  $T$  是循环移位算子。间接地找出这一  $s$ 。多项式的模数  $P_n(x)$  形成了一个域。确保  $x^s \equiv r_0 + r_1x + \cdots + r_{n-1}x^{n-1} \pmod{P_n(x)}$ 。因此，可找出  $(r_0, r_1, \cdots, r_{n-1})$ ，并解  $s$ 。

关系式  $x^s \equiv r_0 + r_1x + \cdots + r_{n-1}x^{n-1} \pmod{P_n(x)}$  隐含  $\mathbf{R}=r_0+r_1T(\mathbf{I})+\cdots+r_{n-1}T^{n-1}(\mathbf{I})$ 。以二元线性方程写出，变为：

$$\mathbf{R}=\mathbf{r}'\mathbf{A} \quad (2)$$

其中， $\mathbf{r}=(r_0 \ r_1 \ r_2 \ \cdots \ r_{n-1})^t$ ，并且  $\mathbf{A}=(\mathbf{I} \ T(\mathbf{I}) \ \cdots \ T^{n-1}(\mathbf{I}))^t$ ，它包括  $\mathbf{I}$  从 0 移位到  $(n-1)$  移位的循环移位。现在在  $\mathbf{R}$  中仅稀少的  $K$  个比特可用于解  $\mathbf{r}$ 。设  $\mathbf{R}$  中  $b_i$  和  $b_0$  之间的下标差为  $k_i$ ， $i=1, 2, \cdots, k-1$ ，则  $\mathbf{R}$  的第 1 个和  $(k_i+1)$  个元素 ( $i=1, 2, \cdots, k-1$ ) 正

好是  $b_0, b_1, \dots, b_{k-1}$ 。通过选择  $\mathbf{A}$  的第 1 列和第  $(k_i+1)$  列,  $i=1, 2, \dots, k-1$ , 形成以下二元线性方程:

$$\mathbf{b}' = \mathbf{r}'\mathbf{M} \quad (3)$$

其中,  $\mathbf{M}$  是  $\mathbf{A}$  的  $n \times K$  子矩阵。

如果  $\mathbf{b}$  是无误差的, 则  $\mathbf{r}$  的解可被表示为:

$$\mathbf{r}' = \tilde{\mathbf{b}}'\tilde{\mathbf{M}}^{-1} \quad (4)$$

其中,  $\tilde{\mathbf{M}}$  是  $\mathbf{M}$  的任何非退化  $n \times n$  子矩阵,  $\tilde{\mathbf{b}}$  是  $\mathbf{b}$  的对应的子向量。

已知  $\mathbf{r}$ , 可使用 Douglas W.Clark 和 Lih-Jyh Weng 的“最大和近最大移位寄存器序列: 有效的事件计数器和简单的离散对数 (Maximal and Near-Maximal Shift Register Sequences: Efficient Event Counters and Easy Discrete Logarithms)”, IEEE 计算机学报 43.5 (IEEE Transactions on Computers 43.5) (1994 年 5 月, 560-568 页) 中描述的 Pohlig-Hellman-Silver 算法来找出  $s$ , 使得  $x^s \equiv r_0 + r_1x + \dots + r_{n-1}x^{n-1} \pmod{P_n(x)}$ 。

由于矩阵  $\mathbf{A}$  (其大小为  $n \times L$ , 其中,  $L=2^n-1$ ) 可能是非常大的, 应当避免储存整个矩阵  $\mathbf{A}$ 。实际上, 如可从上述过程中所见到的, 给定其下标差为  $k_i$  的提取的比特, 仅  $\mathbf{A}$  的第一和第  $(k_i+1)$  列与计算相关。给定已捕捉图像的大小,  $k_i$  的这种选择是相当有限的。由此, 仅计算中涉及的那些列需要被保存。这些列的总数远小于  $L$  (其中,  $L=2^n-1$  是  $m$  序列的长度)。

## VI. 纠错

如果在  $\mathbf{b}$  中有误差, 则  $\mathbf{r}$  的解法变得更复杂。带有纠错的解码的传统方法并不能容易地应用, 因为与已捕捉比特相关联的矩阵  $\mathbf{M}$  可在已捕捉的图像之间变化。

采用一种随机方法。假定  $\mathbf{b}$  中的误差比特数  $n_e$  与  $K$  相比相对较小, 则从  $\mathbf{b}$  的  $K$  个比特中选择正确的  $n$  个比特并且  $\mathbf{M}$  的对应子矩阵  $\tilde{\mathbf{M}}$  为非退化的概率较高。

当所选择的  $n$  个比特都正确时,  $\mathbf{b}'$  和  $\mathbf{r}'\mathbf{M}$  之间的汉明距离, 或与  $\mathbf{r}$  相关联的误差比特数应当最小, 其中,  $\mathbf{r}$  通过公式 (4) 来计算。重复该过程若干次, 标识出导致最小误差比特的正确的  $\mathbf{r}$  是可能的。

如果仅有一个  $\mathbf{r}$  与最小误差比特数相关联, 则它可被看作是正确的解。否则, 如果有一个以上  $\mathbf{r}$  与最小误差比特数相关联, 则  $n_e$  超出由  $\mathbf{M}$  生成的代码的纠错能力的概率是相当高的, 并且解码过程失败。系统然后可继续处理下一已捕捉的图像。在另一实现中, 可考虑关于笔的先前位置的信息。即, 对于每一已捕捉的图像, 可

标识其中接下来期望笔的目标区域。例如，如果用户在由照相机捕捉的两个图像之间为提起笔，则由第二次图像捕捉所确定的该笔的位置应当不会太远离第一个位置。然后可检查与最小误差比特数相关联的每一  $\mathbf{r}$ ，来看从  $\mathbf{r}$  计算的位置  $s$  是否满足局部约束，即，该位置是否在指定的目标区域内。

如果位置  $s$  满足局部约束，则返回数组中的所提取的比特的  $X, Y$  位置。如果不是，则解码过程失败。

图 8 描述了可用于确定序列（或  $m$  序列）中已捕捉图像的位置的过程。首先，在步骤 801，接收涉及已捕捉图像的数据流。在步骤 802，从  $\mathbf{A}$  中提取对应的列，并构造矩阵  $\mathbf{M}$ 。

在步骤 803，从矩阵  $\mathbf{M}$  中随机地选择  $n$  个不相关的列向量，并通过解方程式 (4) 来确定向量  $\mathbf{r}$ 。在步骤 804，该过程被执行  $Q$  次（例如，100 次）。循环次数的确定在“循环次数计算”一节中讨论。

在步骤 805，依照其相关联的误差比特数来对  $\mathbf{r}$  进行排序。排序可使用本领域中已知的各种排序算法来完成。例如，可使用选择排序算法。选择排序算法在数字  $Q$  不大时是有益的。然而，如果  $Q$  变得非常大，则可使用更有效地处理大量项的其它排序算法（例如，合并排序）。

系统然后在步骤 806 通过检查是否有多个  $\mathbf{r}$  与最小误差比特数相关联，来确定是否成功地执行了纠错。如果有多个  $\mathbf{r}$ ，则在步骤 809 返回误差，指示解码过程失败。如果没有多个  $\mathbf{r}$ ，则在步骤 807 计算提取的比特在序列（或  $m$  序列）中的位置，例如，通过使用 Pohig-hellman-Silver 算法来计算。

下一步，计算数组中的  $(X, Y)$  位置如下： $x = s \bmod m_1$ ，并且  $y = s \bmod m_2$ ，并在步骤 808 返回结果。

## VII. 位置确定

图 9 示出了用于确定笔尖的位置的过程。输入是由照相机捕捉的图像，输出可以是笔尖的位置坐标。而且，输出可包括（或不包括）其它信息，如已捕捉图像的旋转角度。

在步骤 901，从照相机接收图像。下一步，在步骤 902 可任选地预处理所接收的图像（如由步骤 902 的虚线框所示），以调整亮和暗像素之间的对比度等等。

下一步，在步骤 903，分析该图像以确定其中的比特流。

下一步，在步骤 904，多次从比特流中随机地选择  $n$  个比特，并确定接收的比

特流在原始序列（或  $m$  序列）中的位置。

最后，一旦在步骤 904 确定了已捕捉图像的位置，可在步骤 905 确定笔尖的位置。

图 10 给出了关于 903 和 904 的更多细节，并示出了提取已捕捉图像内的比特流的方法。首先，在步骤 1001 从照相机接收图像。然后在步骤 1002，该图像可任选地经历图像预处理（如由步骤 1002 的虚线所示的）。在步骤 1003 提取图案。此处，可提取各线上的像素以找出图案的方向和角度  $\theta$ 。

下一步，在步骤 1004 分析接收的图像，以确定底层的网格线。如果在步骤 1005 找到网格线，则在步骤 1006 从图案中提取代码。然后在步骤 1007 对该代码进行解码，并在步骤 1008 确定笔尖位置。如果在步骤 1005 没有找到网格线，则在步骤 1009 返回错误。

#### VIII. 加水印的文档图像：生成、分析和加标签

本发明的实施例涉及分析包含诸如迷宫图案等位置编码的信息的文档图像，并对图像加标签。这类文档分析的结果可用于有效地确定由照相机捕捉的图像在位置编码的文档内的位置。

参考图 11，依照本发明的各种实施例，系统 1100 包括图像生成和捕捉模块 1102 以及分析和加标签模块 1106。捕捉图像用于后续分析和加标签的合适的技术在上文名为“图像捕捉笔”的 III 节中，以及在本节的以下内容和名为“生成加水印的灰度级文档图像”的 VIII.A 节中有揭示。该分析和加标签模块 1106 接收由图像生成和捕捉模块 1102 输出的图像数据 1104 以及离线训练数据 1110 作为输入；执行分析和加标签过程，如下文更详细讨论的；并输出图像标签信息 1108。

依照本发明的各个实施例，照相机 203 捕捉的图像的大小为  $32 \times 32$  像素。嵌入的  $m$  数组的阶为 36，即  $m$  数组的大小为  $(2^{18}+1) \times (2^{18}-1)$ ，它用于对表面 207 上的位置进行位置编码。因此，为从子块  $m$  数组中解码唯一的  $x-y$  位置，用于确定表面 207 上的位置的子块的比特数应当至少为 36。

依照本发明的各个实施例，用迷宫图案对印刷文档加水印。这类迷宫图案可被诸如文本等文档的内容遮蔽。根据这类遮蔽的程度，可能无法从由照相机捕捉的图像中提取足够的  $m$  数组比特来确定该由照相机捕捉的图像在文档内的位置。

依照本发明的各个实施例，加水印的文档图像被分析并加标签。分析和加标签涉及基于文档内特定位置上的迷宫图案单元的可见程度来确定加水印的文档中

的 x-y 位置的能力。

#### VIII.A. 生成加水印的灰度级文档图像

图 12 描述了依照本发明的各个实施例用于生成加水印的灰度级文档图像的步骤，该文档可被分析和加标签。通过如由 1200 所示的将电子文档描绘成对应于印刷文档的位图，或通过如 1202 和 1206 所示的扫描纸件文档，来获取文档图像。然后对位图进行重采样到适当的分辨率，如 1208 所示。分辨率的确定基于以下考虑：

(1) 文档图像的分辨率不应当小于已捕捉图像的分辨率，因为当已捕捉图像的位置无法通过  $m$  数组解码来确定时，已捕捉图像将与加水印的灰度级文档图像进行匹配，以确定位置；(2) 一个印刷的迷宫图案单元应当被映射到整数个文档图像像素，使得匹配算法可以有效地起作用。例如，如果照相机的分辨率是  $0.15\text{mm}/\text{像素}$ ，即，物理世界的  $0.15\text{mm}$  被映射到一个照相机像素，则印刷迷宫图案单元大小为  $0.45\text{mm} \times 0.45\text{mm}$ ，即，一个印刷迷宫图案单元被映射到照相机传感器上的  $3 \times 3$  像素，文档图像的分辨率应当也被设置为  $0.15\text{mm}/\text{像素}$ ，使得印刷迷宫图案单元将被映射到文档图像中的  $3 \times 3$  像素区域。

然后对获取的图像加水印。图 13 描述了文档内容 1300 的一部分的图像 1300、经位置编码的迷宫图案的一部分 1302、组合的内容和迷宫图案 1304、以及内容和迷宫图案的一部分的经放大的视图 1306，它示出了：(1) 迷宫图案单元由于被叠加的文本遮蔽而不可见的位置 1308、以及 (2) 迷宫图案单元未被文档内容遮蔽并因此可见的位置 1310。

#### VIII.B. 文档图像的分析 and 加标签

依照本发明的各种实施例，文档图像通过以下步骤来加以分析：将图像划分成具有实际上与迷宫图案单元相同大小的相对较小的块；确定这些小块是否被诸如文档文本等文档内容遮蔽；对每一像素，对以该像素为中心的邻域窗中的完全可见的块的数量进行计数；以及基于该数量对像素加标签。邻域窗可具有实际上与照相机 203 捕捉的图像相同的大小，即， $32 \times 32$  像素。

注意，上述过程不要求对灰度级文档图像加水印，即，内容与迷宫图案相组合。只要文档图像被划分成具有实际上与迷宫图案单元相同大小的相对较小的块，则分析被文档内容遮蔽的迷宫图案单元就等效于分析被文档内容遮蔽的块。

图 14 示出了这一  $32 \times 32$  像素邻域窗 1400 以及其中心像素 1402。依照本发明

的各个实施例，文档图像的像素可被加标签为以下四种类型之一：

类型 I：32×32 像素邻域窗（以所考虑的像素为中心）基本上仅包含经位置编码的迷宫图案单元。

类型 II：32×32 像素邻域窗包含 60 个或更多可见迷宫图案单元。

类型 III：32×32 像素邻域窗包含 36 到 60 个可见迷宫图案单元。

类型 IV：32×32 像素邻域窗包含 35 个或更少的可见迷宫图案单元。

依照本发明的各个实施例，当已捕捉图像的中心位于类型 I 或类型 II 区域中时，图像在文档内的位置可唯一地通过 m 数组解码来确定。

图 15 示出了示例文档的文档分析结果，其中，文档的区域被不同地加上颜色（或阴影），以表示基于每一像素的 32×32 邻域窗内的可见迷宫图案单元的数量，已对文档的该部分加上了哪一类型（即，类型 I、II、III 或 IV）的标签。

注意，在对可见迷宫图案单元的数量进行计数时，32×32 像素邻域窗应当被旋转 360 度，以考虑照相机可能在任一角度捕捉文档图像的事实。然而，为易于实现，仅分析非倾斜视图，并且由倾斜角引起的误差在确定像素类型的阈值中考虑。例如，图 16 描述了已捕捉的图像被旋转 45 度的情况。这是最坏的情况，其中，丢失了可见迷宫图案单元的最大值的 17.2%。即使如此，类型 II 像素仍具有至少 49（=60×(1-17.2%））个可见迷宫图案单元，这允许通过 m 数组解码来确定的唯一的 x-y 位置。换言之，如果像素被加标签为类型 I 或 II，则无论照相机已什么角度来捕捉图像，仍具有足够数量的可见迷宫图案单元来用 m 数组解码确定 x-y 位置。因此，如果已捕捉图像无法被解码，则很可能它位于类型 III 或 IV 区域。

如果由于没有足够的迷宫图案单元可见而无法从笔画图像的序列中解码 x-y 位置，则可使用一搜索算法来确定文档中诸如在类型 III 和 IV 像素区域中的图像的位置。仅在类型 III 和 IV 区域上使用这一搜索算法相对于对实际上文档的所有部分使用该搜索算法，可降低用于确定图像位置的计算成本。例如，在图 15 的示例文档中，仅 13.7% 的区域被加标签为类型 III（11.1%）或类型 IV（2.6%）。与搜索整个文档相关联的计算成本远高于仅搜索类型 III 和 IV 区域的计算成本。在一个实现中，可花费 10 秒以上来搜索整个文档，而只需不到 1 秒来仅搜索类型 III 和 IV 区域。文档图像分析很大程度上降低了计算成本。

#### IX. 由照相机捕捉的图像的加标签

由照相机捕捉的图像也可用上文在名为“文档图像分析和加标签”的 VIII.B

节中讨论的四种类型来加标签。可确定由照相机捕捉的图像基本上是否仅包含经位置编码的迷宫图案。换言之，可确定该图像是否为类型 I 图像。这一初始确定将在下文更详细讨论。

如果由照相机捕捉的图像不是类型 I 图像，则可分析由照相机捕捉的图像的经位置编码的迷宫图案，来确定可从该图像中提取的若干位置编码比特数。首先计算迷宫图案网格线的参数（沿每一维度的比例和旋转，即仿射变换），然后确定其中嵌入了迷宫图案的原始方向（或象限）。因此，m 数组比特信息是基于网格线和条形方向（bar direction）来确定的。如果提取的比特数大于约 60，则将该图像加标签为类型 II；如果在提取的比特数在约 36 和 60 之间，则该图像被加标签为类型 III；如果少于约 36 个比特被提取，则该图像被加标签为类型 IV。为以这一方式对由照相机捕捉的图像进行分类，可使用下文名为“阈值算法”的 IX.B 节中讨论的阈值算法，以将迷宫图案从文本区域中分离。然后，可基于所提取的比特数对图像加标签。从类型 I、II 和 III 中提取的比特可用于确定由照相机捕捉的图像在较大的文档中的 x-y 位置。这一类型的位置确定过程对类型 IV 图像可省略，因为它们不包含用于 x-y 位置解码的足够的比特。

与同时包含文档内容和迷宫图案相反，为确定图像基本上是否仅包含经位置编码的迷宫图案，依照本发明的各个实施例，可使用一种被称为梯度图像直方图的支持间隔（SIGIH）的特征。SIGIH 是基于以下知识来使用的：纯迷宫图案图像通常不包含边，而具有诸如文本等文档内容的图像通常包含边，因为文档内容通常比迷宫图案单元或空白区域更深。

#### IX.A. 特征提取

依照本发明的各个实施例，特征提取的第一个步骤是梯度算子，如 Sobel 边算子或另一算子，可用于获取梯度图像。参考图 17，两个卷积核 1700 和 1702 形成了 Sobel 边算子。为使用 Sobel 边算子，图像中的每一像素与这两个卷积核 1700 和 1702 进行卷积。一个核最大地对应于一般性的垂直边，而另一个核最大地对应于水平边。两个卷积的“向量和”值可被用做特定像素的输出值。该结果是梯度图像。

可计算由梯度算子产生的梯度图像的直方图。然后可从梯度图像的直方图中获取梯度图像直方图的支持间隔（SIGIH）。例如，图 18 示出了仅包含迷宫图案的已捕捉图像的梯度图像的直方图。对图 18 的直方图，SIGIH 特征为 17，它是具

有非零值的沿 x 轴的最大数。

图 19 示出了包含诸如文本和/或一个或多个图等文档内容的已捕捉图像的梯度图像的直方图。对图 19 的直方图，SIGIH 特征为 44。

### IX.B. 阈值算法

依照本发明的各个实施例，用于确定图像是否为类型 I 的阈值算法可被划分成两个时期：（1）离线训练、以及（2）在线加标签。

#### IX.B.1. 离线训练

在离线训练时期，可对具有已知标签（即，类型 I、II、III 或 IV）的相对大量的图像计算 SIGIH 特征。纯迷宫图案图像的 SIGIH 通常低于其它类型的图像的 SIGIH。可基于处理训练数据的结果选择最优阈值  $\eta$ 。对于训练组中的基本上所有的纯迷宫图案图像，其 SIGIH 小于  $\eta$ ，而对于基本上所有的非纯迷宫图案图像，其 SIGIH 大于  $\eta$ 。图 20 示出了离线训练的一个示例结果，其中，最优阈值  $2004 - \eta$  被选择为 32。还示出对应于纯迷宫图案图像的直方图条 2002，以及对应于非纯迷宫图案图像（即，包括文档内容的图像）的直方图条 2006。

#### IX.B.2. 在线加标签

在在线加标签期间，计算每一已捕捉图像的 SIGIH。如果 SIGIH 小于  $\eta$ ，则将该图像加标签为类型 I 图像。否则，如果 SIGIH 大于  $\eta$ ，则将该图像加标签为除类型 I 之外的类型（即，类型 II、III 和 IV）。

上述涉及文档分析和加标签的讨论假定已捕捉图像的对比度保持相对一致。换言之，假定图像传感器的放大器和/或照相机的光照条件保持相对一致。如果这些条件明显地改变，则可执行新的训练来更新特征参数。

## X. 总结性评论

上文描述的仅是本发明的原理的说明性应用。本领域的计数人员可在不脱离本发明的精神和范围的情况下实现其它方案和方法。本发明的任何方法可以可储存在计算机的盘或其它计算机可读媒质上的软件来实现。

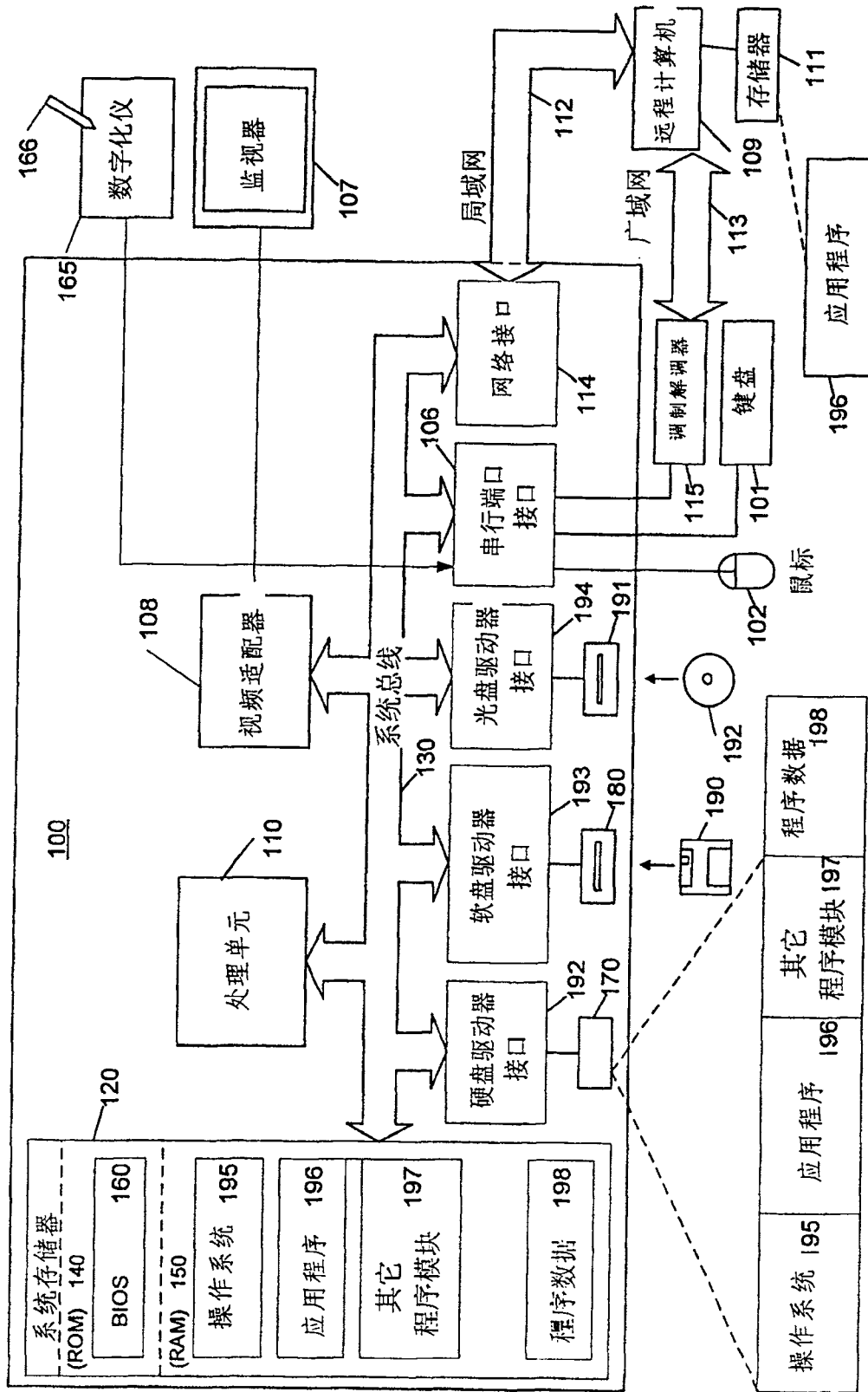


图 1

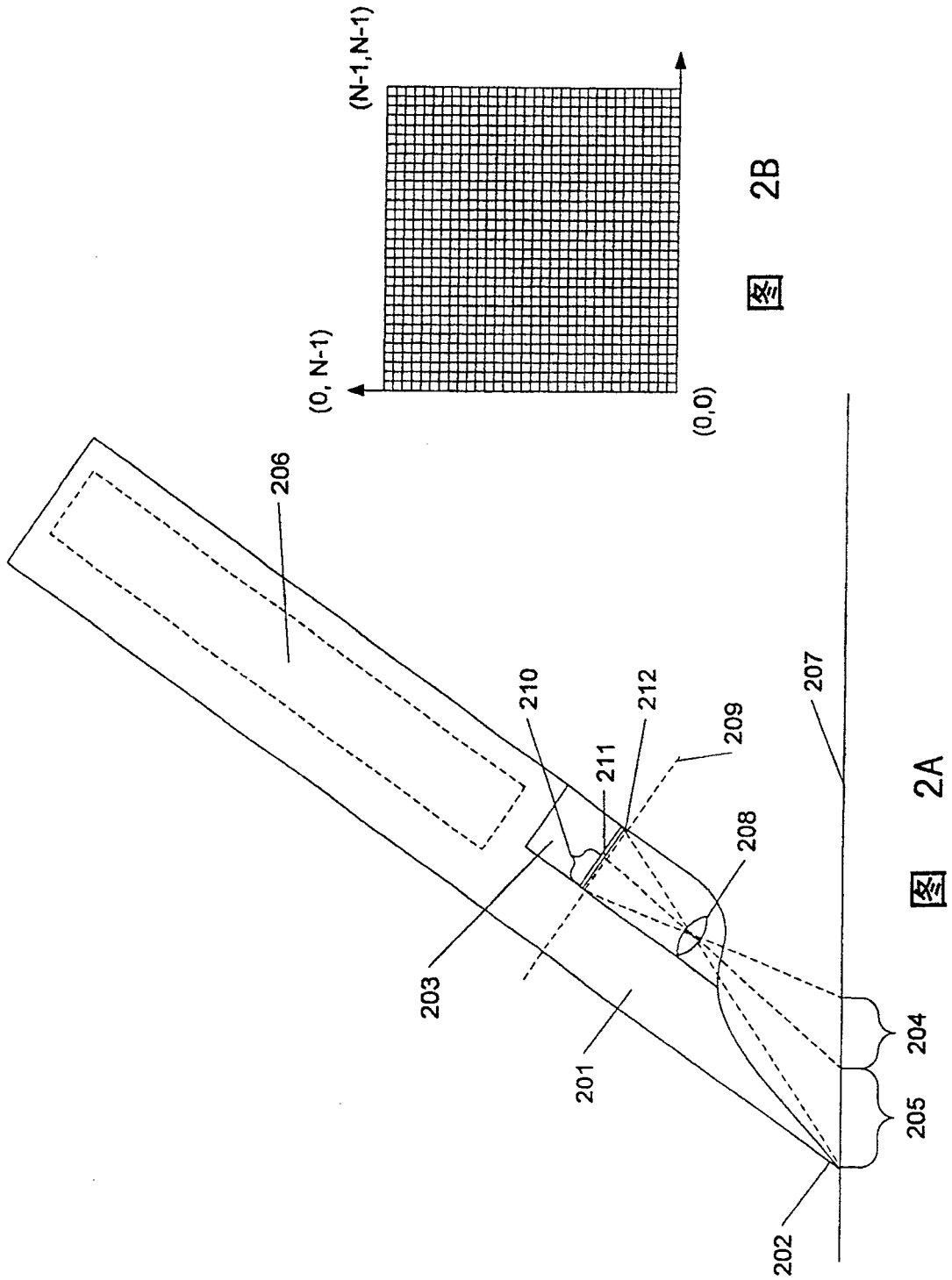


图 3A 0000010 00 01100 0101 001 111010 0 0111001 0 010110 111 0110 01101 01 0111111

图 3B 0000000 0011101 0100111 1110100 0111010 1000101 1110100 0100111 0011101

图 3C 000100100 001111110 01011101 010011001 011100111 001011010 011000011

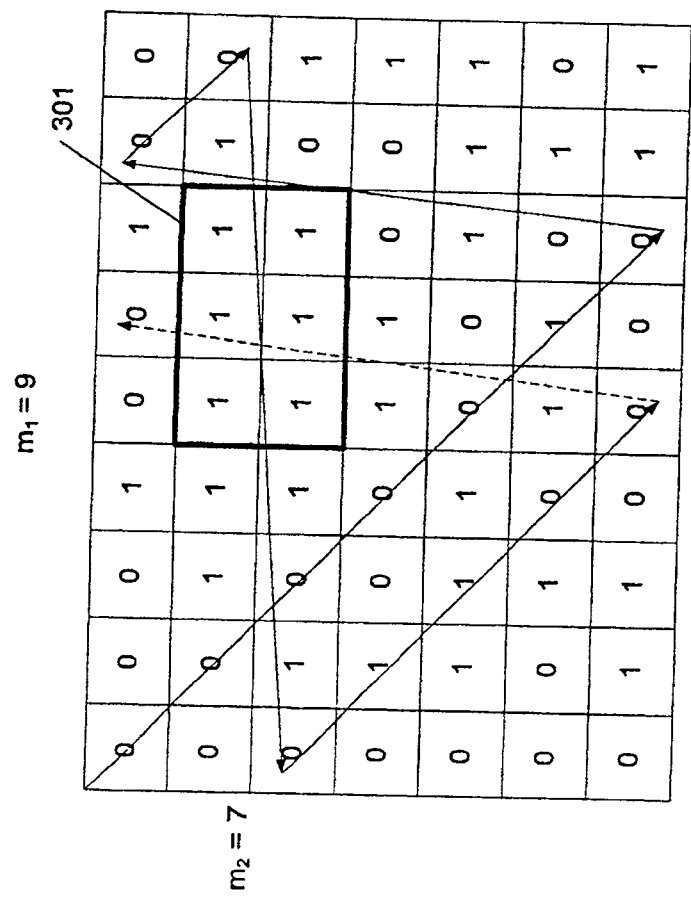


图 3D

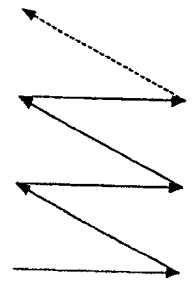


图 3E

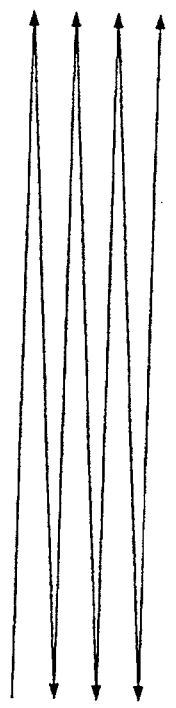


图 3F

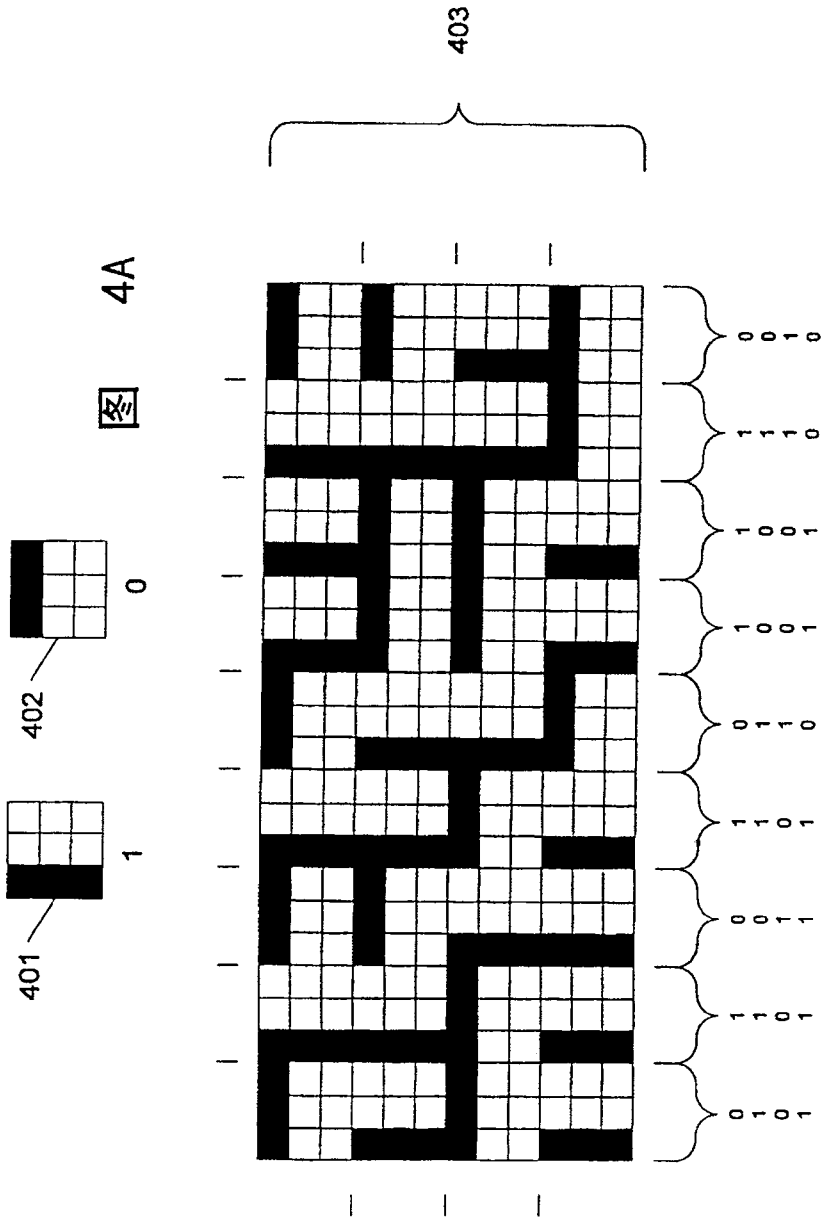
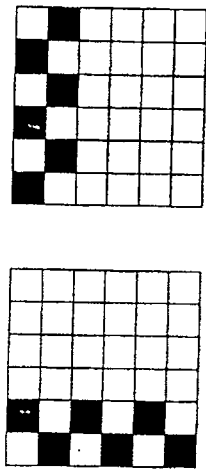



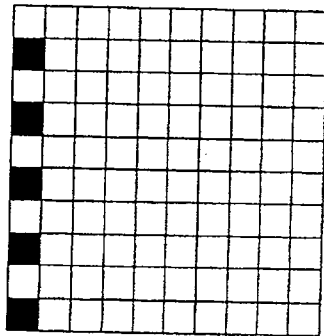
图 4B



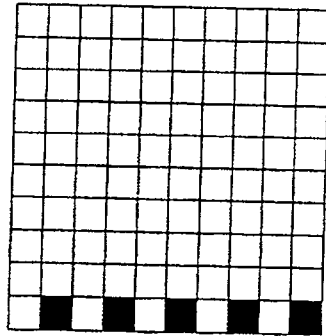
1

0

 4C

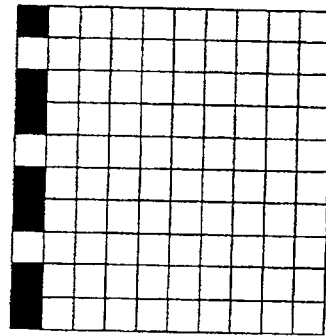


1

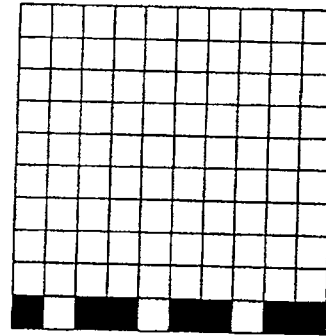


0

 4D



1



0

 4E



5A



5B



5C



5D



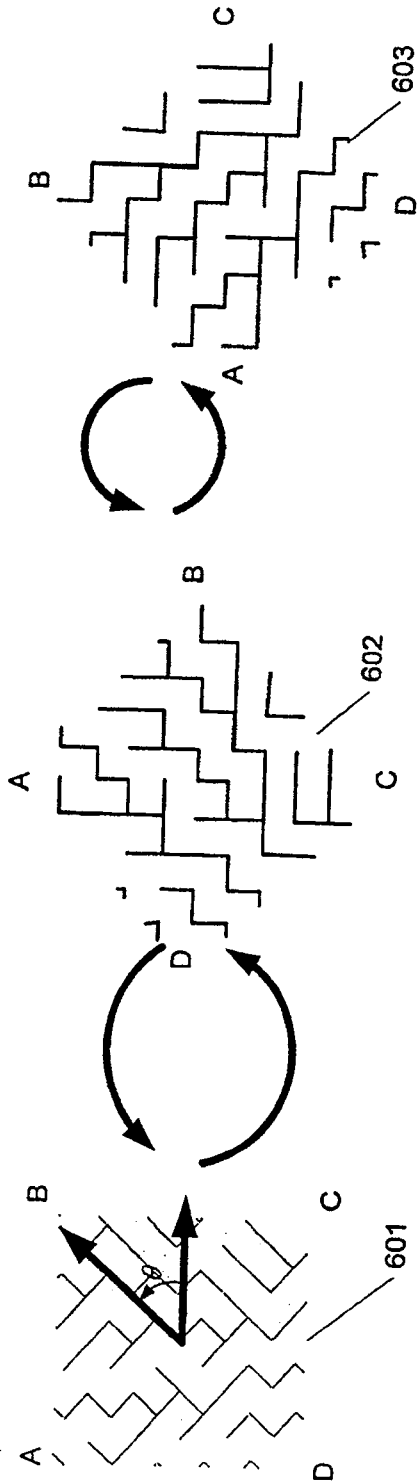
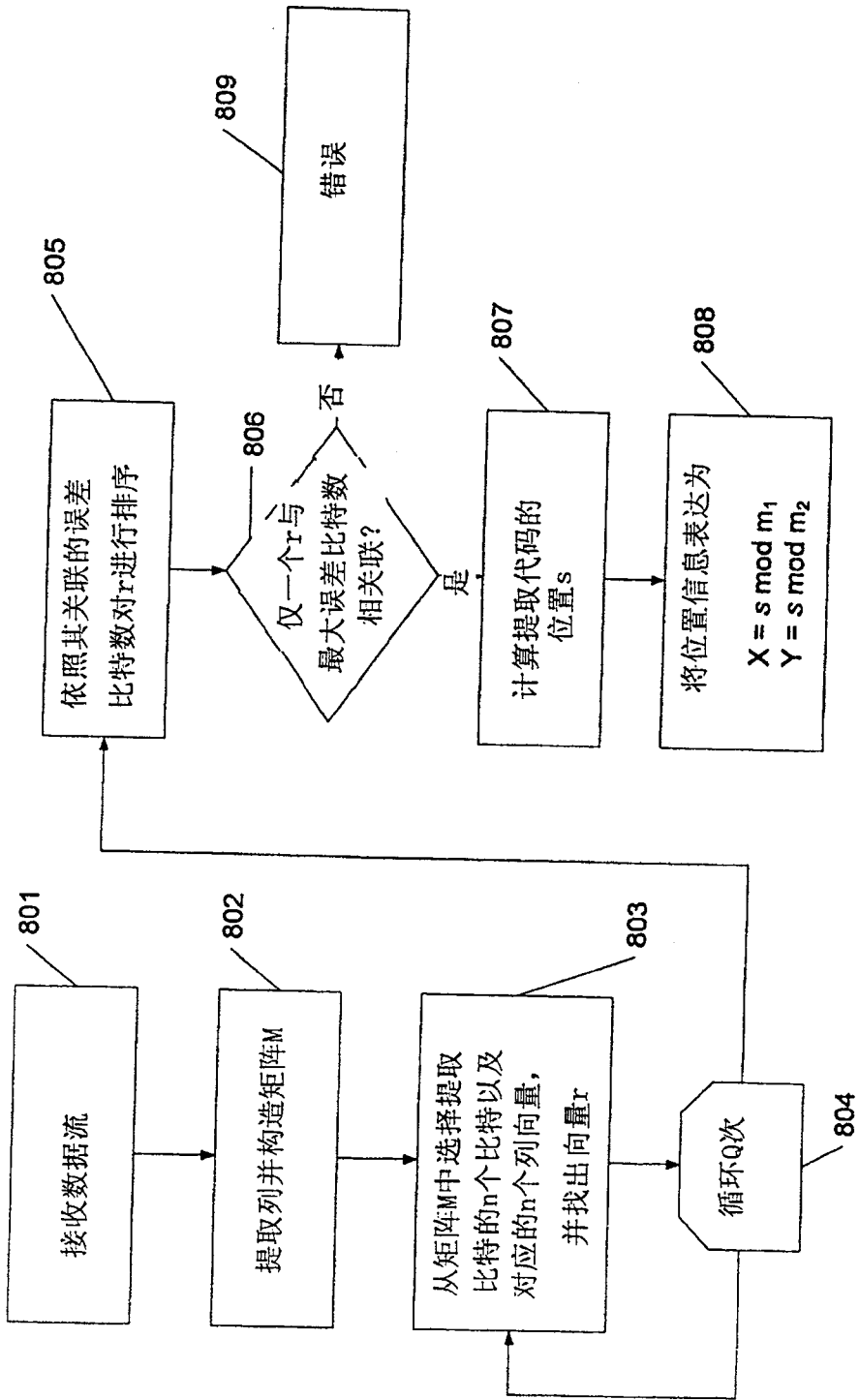


图 6

$$o = \theta + \begin{cases} 0 - no a \\ \frac{\pi}{2} - no b \\ \pi - no c \\ \frac{3\pi}{2} - no d \end{cases}$$

图 7



8

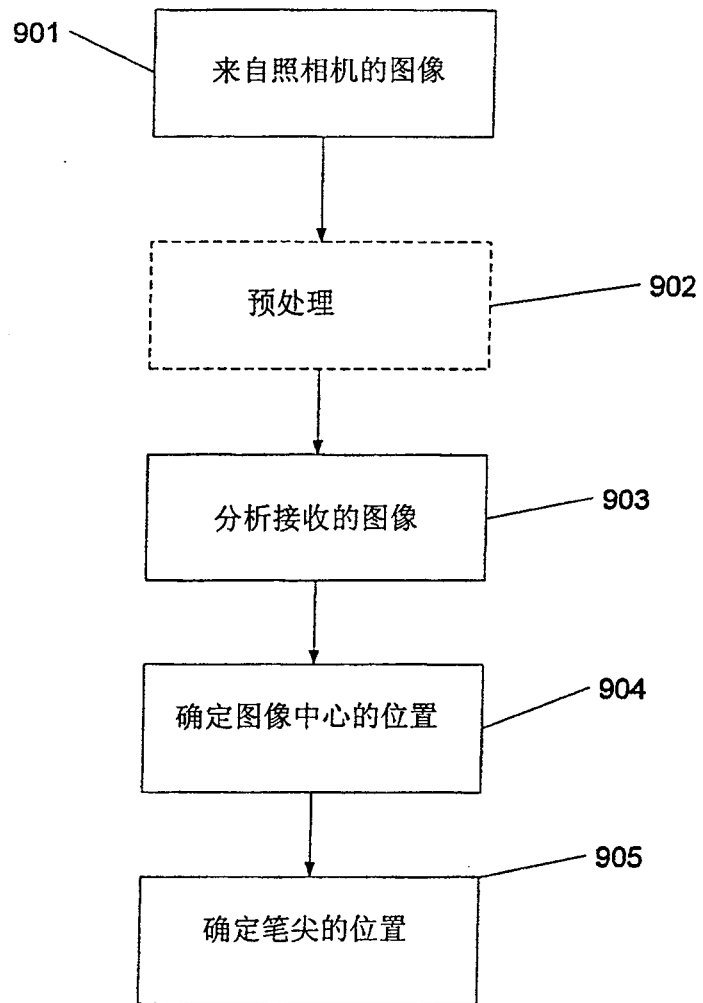


图 9

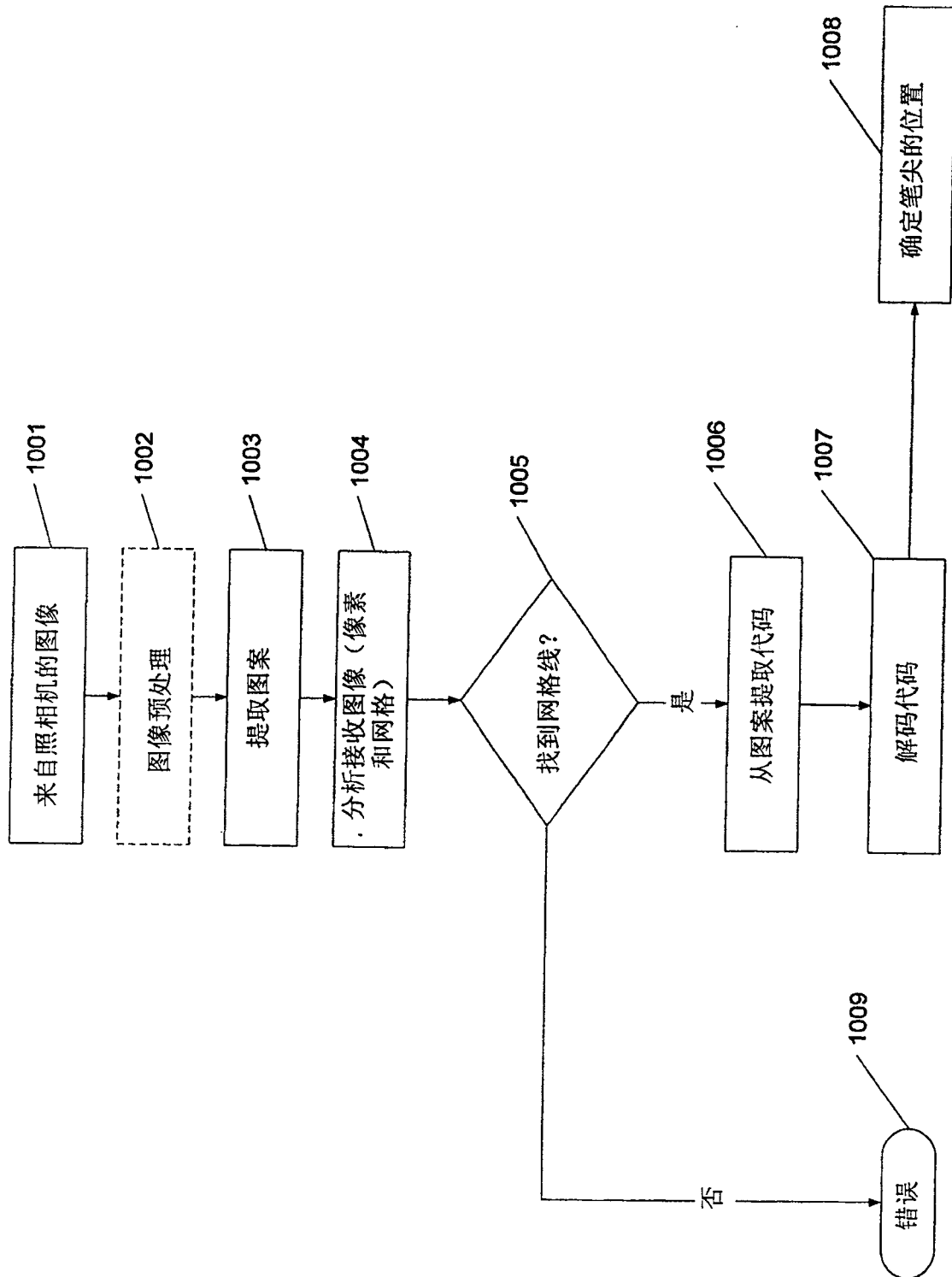


图 10

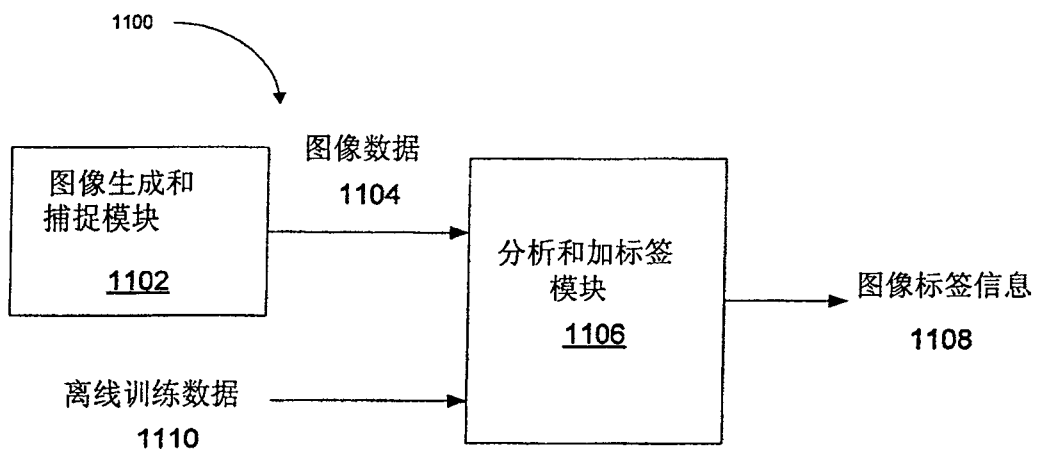


图 11

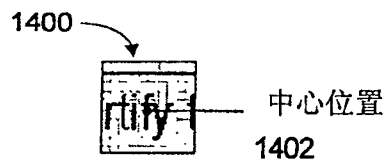


图 14

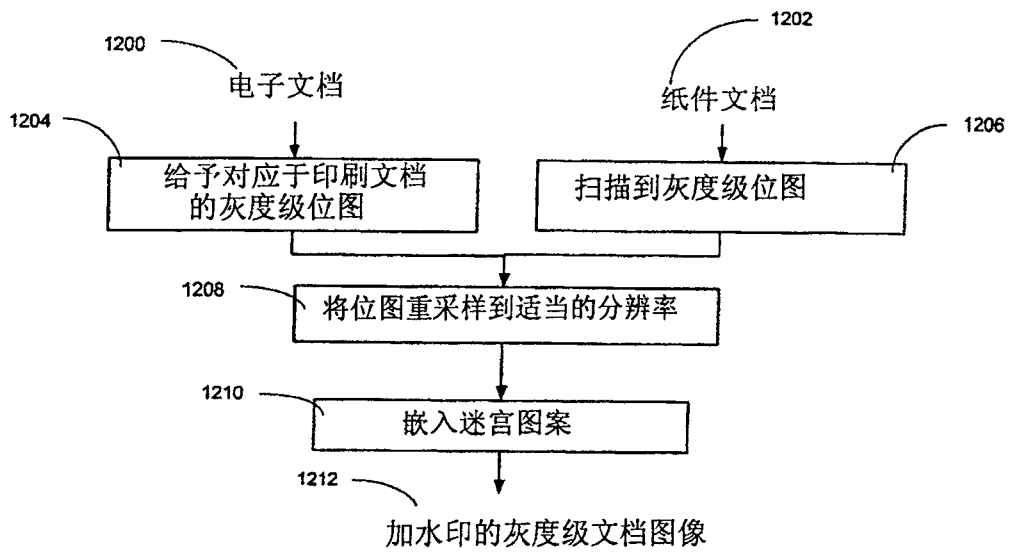


图 12

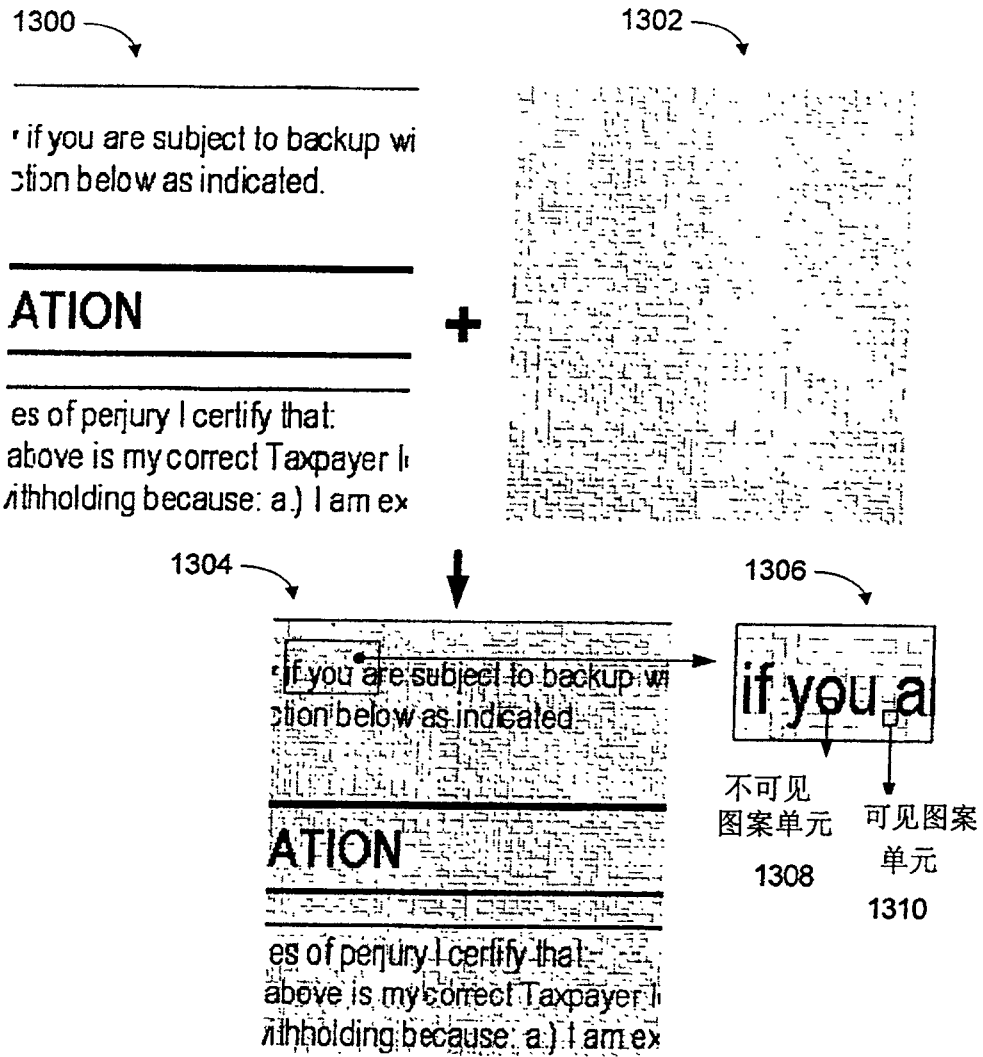


图 13

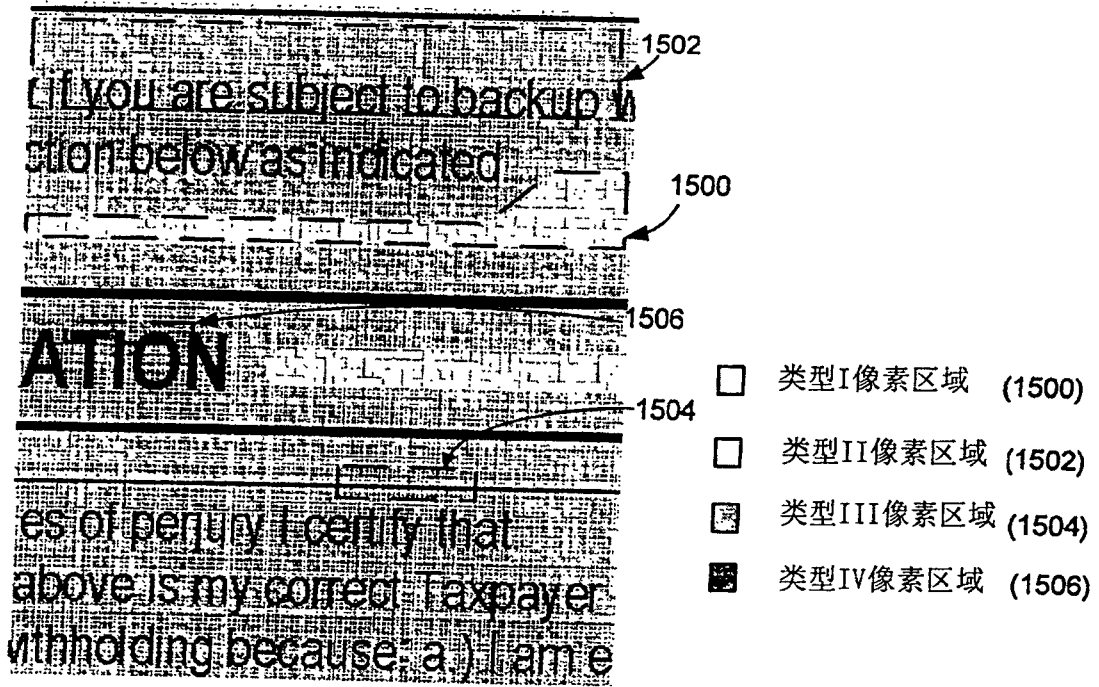


图 15

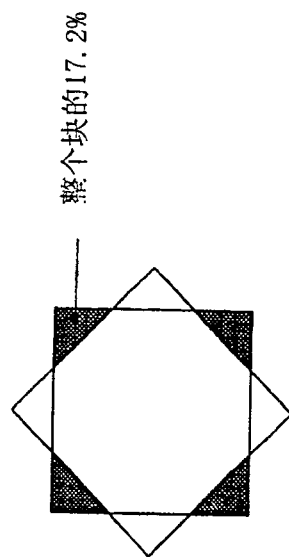


图 16

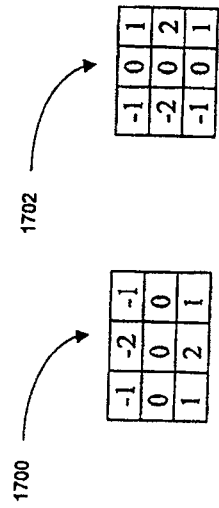


图 17

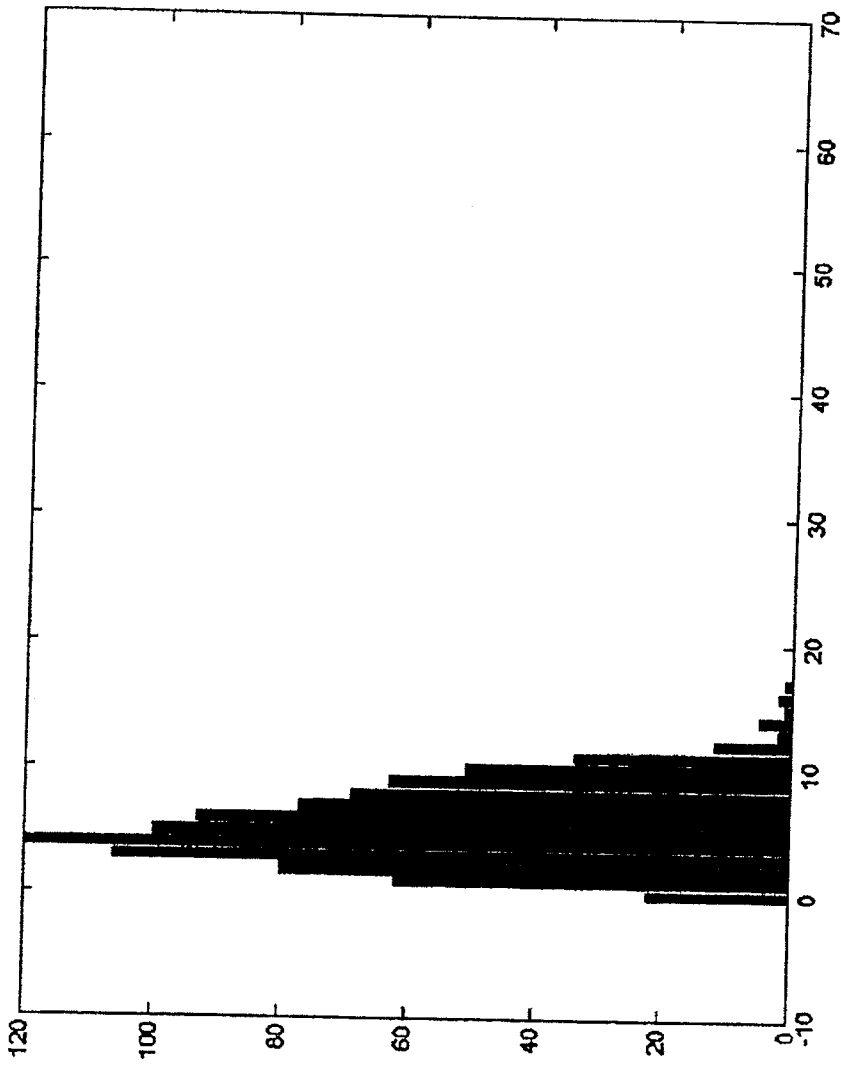


图 18

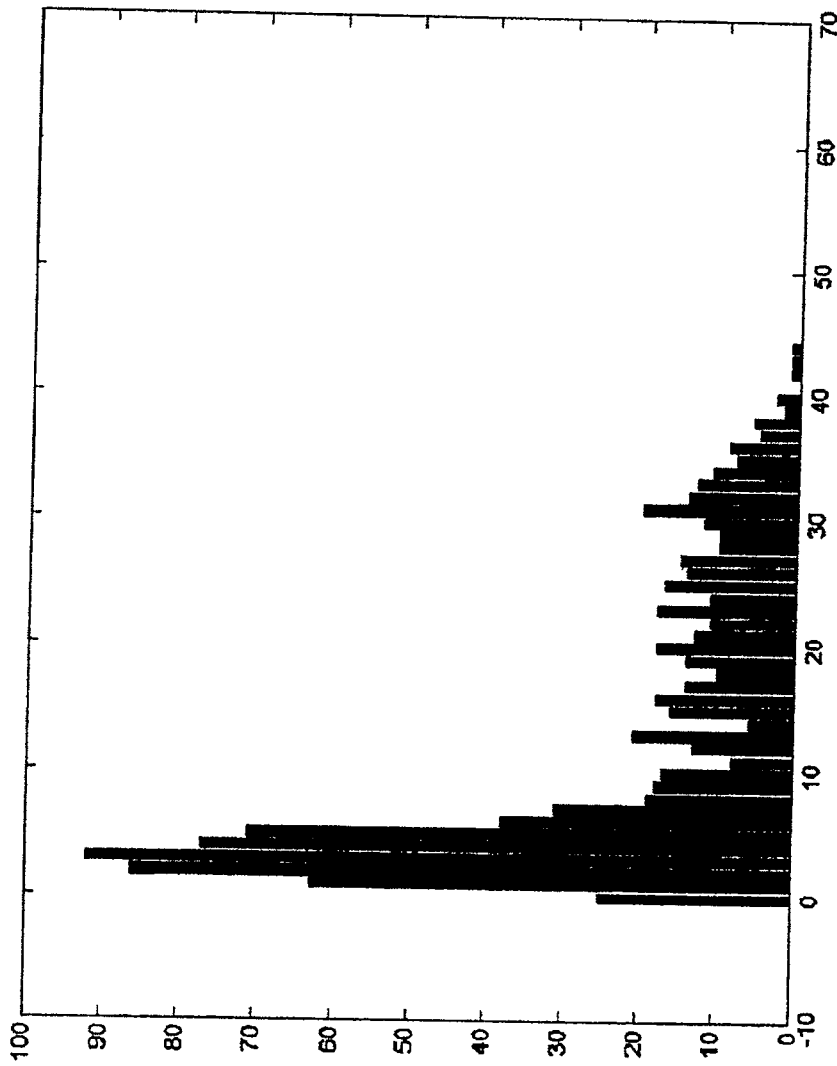


图 19

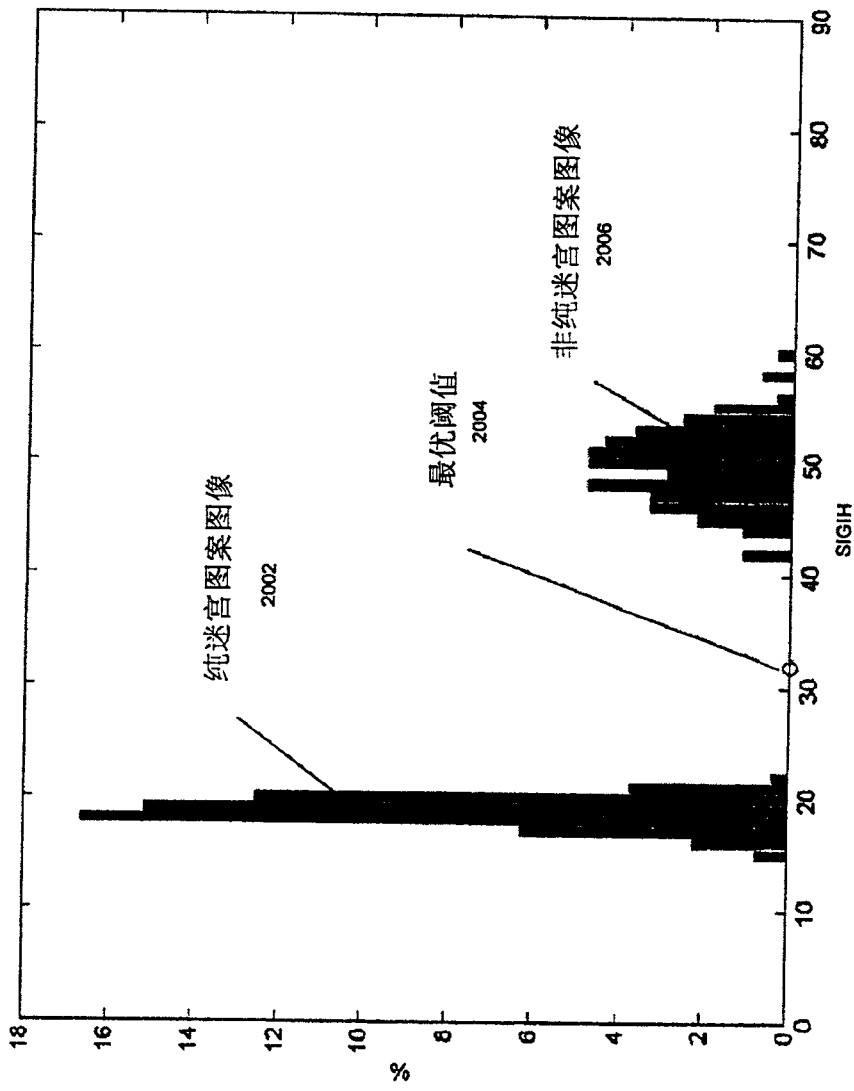


图 20

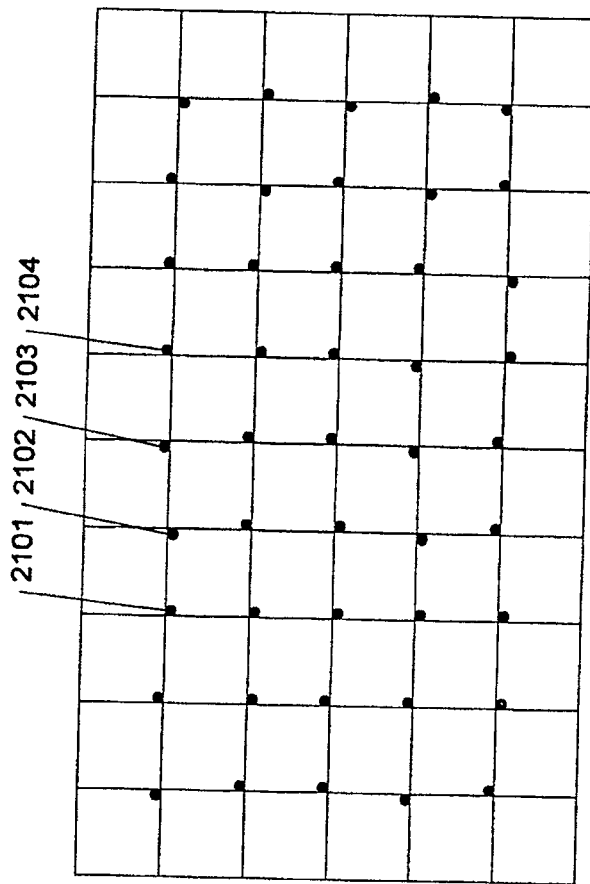


图 21