



(12) 发明专利

(10) 授权公告号 CN 101620725 B

(45) 授权公告日 2012.07.18

(21) 申请号 200910148769.7

至第 18 栏.

(22) 申请日 2009.07.03

US 6768491 B2, 2004.07.27, 说明书第 2 栏
第 24-41.

(30) 优先权数据

12/167,997 2008.07.03 US

US 7173631 B2, 2007.02.06, 说明书第 1 栏
第 58-67 行.

12/167,998 2008.07.03 US

US 6967663 B1, 2005.11.22, 说明书第 1 栏
至第 18 栏.

(73) 专利权人 辉达公司

审查员 张维克

地址 美国加利福尼亚州

(72) 发明人 卡斯·W.·埃弗里特
史蒂文·E··莫尔纳

(74) 专利代理机构 北京市磐华律师事务所

11336

代理人 董巍 顾珊

(51) Int. Cl.

G06T 1/20 (2006.01)

G06T 5/00 (2006.01)

G06T 11/40 (2006.01)

(56) 对比文件

CN 1795468 A, 2006.06.28, 全文.

US 6967663 B1, 2005.11.22, 说明书第 1 栏

权利要求书 1 页 说明书 11 页 附图 7 页

(54) 发明名称

混合多重采样 / 超采样抗锯齿

(57) 摘要

本发明公开了一种配置成用混合抗锯齿对图形图元进行着色的计算装置。根据本发明的在图元着色过程中用于动态调整像素采样率的系统和方法能改进图像质量或增加着色性能。混合抗锯齿处理通过在每个像素段中选择许多着色样本来实施。当一簇子像素样本（多个样本）每次穿过段着色器流水线都被处理时，超采样和多重采样混合抗锯齿就被使用。基于渲染状态，每个图元的着色器流水线和每个簇的多个样本能被动态地确定。



1. 一种配置成用混合抗锯齿对图形图元进行着色的计算装置，所述计算装置包括：光栅化器，所述光栅化器包括混合抗锯齿控制单元，所述混合抗锯齿控制单元配置用来：

接收图形图元；

确定用于对与图形图元交叉的每个像素进行抗锯齿处理的超采样簇的数量；

确定用于为每个超采样簇处理图形图元的多重样本的数量；以及

片段着色单元，所述片段着色单元与所述光栅化器相连接，并设置为使用通过片段着色单元的多个通道对图形图元着色，其中用于产生和图形图元交叉的每个混合抗锯齿像素的多个通道的数量少于或等于超采样簇的数量。

2. 根据权利要求 1 所述的计算装置，其中所述超采样簇的数量基于所述计算装置的渲染状态来确定。

3. 根据权利要求 2 所述的计算装置，其中所述渲染状态包括高品质模式设置、高性能设置、alpha 测试设置和使用具有高频内容的纹理贴图中的一种或多种。

4. 根据权利要求 1 所述的计算装置，其中所述片段着色单元进一步被设置为产生后着色器覆盖，所述后着色器覆盖表明对于每个超采样簇来说，哪个多重样本被图形图元所覆盖。

5. 根据权利要求 4 所述的计算装置，进一步包括光栅运算单元，所述光栅运算单元连接于所述片段着色单元，并被设置为根据所述后着色器覆盖范围对由图形图元所覆盖的每个多重样本进行图形图元的 z 测试，以产生 z 测试值。

6. 根据权利要求 5 所述的计算装置，其中所述光栅运算单元进一步被设置成为和每个所述图形图元相交叉的一部分 z 缓冲器来压缩 z 测试值。

7. 根据权利要求 1 所述的计算装置，其中被用来对和第一个图形图元相交叉的每个像素进行抗锯齿处理的所述超采样簇的数量不同于被用来对和第二个图形图元相交叉的每个像素进行抗锯齿处理的所述超采样簇的数量。

8. 根据权利要求 1 所述的计算装置，其中被用来产生和图形图元相交叉的每个混合抗锯齿化像素的多个通道的数量不包括不具有被所述图形图元覆盖的至少一个多重样本的任何超采样簇的通道。

9. 根据权利要求 1 所述的计算装置，其中所述片段着色单元进一步被设置为对所述图形图元进行着色，所述着色是通过对每个所述超采样簇中的仅仅其中一个所述多重样本计算着色值，并为处于同一超采样簇中的其他多重样本复制所述着色值来进行的。

10. 根据权利要求 1 所述的计算装置，其中所述片段着色单元进一步被设置用于通过下列方式为所述超采样簇中的第一个超采样簇计算着色值：

使用在所述第一个超采样簇中的第一个多重样本的位置；

使用被图形图元覆盖的所述第一个超采样簇中的多重样本的几何质心作为质心；或

使用在第一个超采样簇中被图形图元覆盖并最接近第一个超采样簇的几何质心的多重样本作为近似质心。

混合多重采样 / 超采样抗锯齿

技术领域

[0001] 本发明的实施例大体上涉及图像处理中的抗锯齿技术,更具体地涉及动态调整每个像素段中被着色的样本数目。

背景技术

[0002] 图像处理器通常被配置为通过多重采样或超采样来实行抗锯齿。在多重采样中,每个像素段被着色一次,所产生的颜色值被复制以用于所有覆盖的子像素样本。在超采样中,每个像素段被着色 N 次,每个被覆盖的子像素样本被着色一次。

[0003] 多重采样适用于图元边缘的抗锯齿处理,因为这里重要的是哪些样本被到来的图元所覆盖。图像纹理通常经过预过滤处理,这样被着色的颜色值具有足够低的空间频率,以至于每个像素着色一次就足够了。但是,例如纹理 alpha 透明和高频镜面突出的这样一些效果可能具有高于像素频率的频率,需要以高于像素频率的频率来进行着色以避免锯齿假象产生。通常需要进行超采样以避免这些类型的锯齿现象。但是,由于着色通常是渲染中代价最高的操作,因此对像素中的每个样本着色是代价极其高的。同时,一些超采样实现方式需要对输入图元进行多次处理,即每个子像素样本处理一次,这就更降低了效率。超过每像素一次但低于每个样本一次的着色率可足以缓解上述锯齿的产生。

[0004] 相应地,现有技术所需要的是一种使用适合于当前被渲染的几何图形的像素着色率的系统和方法。着色率可以被降低以改善图像质量或者改进着色性能。

发明内容

[0005] 一种用于动态调整图元着色过程中像素采样率的系统和方法,所述系统和方法能改进图像质量或增加着色性能。着色率可以从每像素一次(多重采样)到每样本一次(超采样)之间进行任意变化,或者是任意某个中间值,用以改进图像质量或提高着色性能。假设对一个渲染目标(图像缓冲器),每个像素有指定数目的样本,则可以动态选择多个着色器通道。在对于片段着色器的每个通道处理一族子像素样本(多重采样)的情况下,使用了超采样和多重采样抗锯齿的组合。对于每个像素组合该超采样簇以产生抗锯齿的像素。

[0006] 本发明的用于着色图元方法的多个实施例在计算装置中使用混合抗锯齿,该计算装置被配置为能每像素产生多个样本,这些实施例包括接受图形图元并确定用来对插入到图形图元的每个像素进行抗锯齿的超采样簇的数目。在计算装置中图形图元使用多个流水线通过段着色单元来着色,其中用于产生每个插入图形图元中混合抗锯齿像素的多个流水线的数目,少于或等于超采样簇的数目。

[0007] 本发明的不同实施例包括被配置为使用混合抗锯齿对图形图元进行着色的计算装置。该计算装置包括连接到段着色单元的光栅。该光栅包括混合抗锯齿控制单元。该混合抗锯齿控制单元被配置成接收图形图元并确定用来对每个插入到图形图元中的像素进行抗锯齿处理的超采样簇数目。该段着色单元被配置成使用多流水线对图形图元进行着色,其中用于产生每个插入图形图元中混合抗锯齿像素的多个流水线的数目,少于或等于

超采样簇的数目。

附图说明

[0008] 为了详细地理解本发明的上述特征,对于以上简要说明的发明,将参照实施例进行更为具体的描述,其中对部分实施例结合附图进行了说明。然而,需要注意的是,附图中示出的只是本发明代表性的实施例,因此不能认为附图限制了本发明的范围,本发明可以适用于其他同样有效的实施例。

[0009] 图 1 是为实施本发明一个或多个方面所配置的计算机系统的框图;

[0010] 图 2 是根据本发明一个或多个方面的图 1 所示计算机系统的并行处理子系统的框图;

[0011] 图 3 是根据本发明一个或多个方面的图 2 所示并行处理子系统核心的框图;

[0012] 图 4 是根据本发明一个或多个方面的图像处理流水线的概念图;

[0013] 图 5A 示出了根据本发明一个或多个方面的在一个像素中的超采样簇和多重采样位置;

[0014] 图 5B 示出了根据本发明一个或多个方面的在多重采样簇中的片段和质心位置;

[0015] 图 5C 是根据本发明一个或多个方面的图像处理流水线一部分的框图;知

[0016] 图 6 是根据本发明一个或多个方面的实施混合抗锯齿处理方法步骤的流程图。

具体实施方式

[0017] 在下文的描述中,给出了大量具体的细节以便提供对本发明更为彻底的理解。然而,对于本领域技术人员来说显而易见的是,本发明可以无需一个或多个这些细节而得以实施。在其他的例子中,为了避免与本发明发生混淆,对于本领域公知的一些技术特征未进行描述。

[0018] 系统概述

[0019] 图 1 是为实施本发明一个或多个方面而配置的计算机系统 100 的框图。计算机系统 100 包括中央处理单元 (CPU) 102 以及系统存储器 104,二者通过包括存储器桥 105 的总线路径进行通讯。存储器桥 105 例如可以是北桥芯片,通过总线或其他通讯路径 106(例如超传输链接连接)与 I/O(输入 / 输出)桥 107 相连接。I/O 桥 107 例如可以是南桥芯片,从一个或多个用户输入装置 108(例如键盘、鼠标)接收用户输入,并将该输入通过路径 106 和存储器桥 105 转发给 CPU 102。并行处理子系统 112 通过总线或其他通讯路径 113(例如 PCI Express、加速图像端口或超传输链接)与存储器桥 105 相连接;在一个实施例中,并行处理子系统 112 是将像素传送给显示装置 110(例如传统的 CRT 或基于 LCD 的显示器)的图像子系统。存储在系统存储器 104 中的装置驱动器 103 在由 CPU 102 执行的进程(例如应用程序)和并行处理子系统 112 之间实现接口,翻译所需的程序指令以便由并行处理子系统 112 执行。

[0020] 系统盘 114 同样连接于 I/O 桥 107。开关 116 为 I/O 桥 107 和诸如网络适配器 118 以及各种外插卡 120 和 121 的其他部件之间提供了连接。其他部件(图中没有示出)包括 USB 或其他端口连接、CD 驱动器、DVD 驱动器、电影刻录装置及类似的部件,也可以与 I/O 桥 107 相连接。将图 1 中的各种部件相互连接的通信路径可以用任何适用的协议来实现,比如

PCI(外设部件互联)、PCI Express(PCI-E)、AGP(加速图像端口)、超传输或其他任何一种总线或点对点通信协议，并且不同装置之间的连接可以使用本领域已知的不同协议。

[0021] 图2示出了并行处理子系统112的一个实施例。并行处理子系统112包括一个或多个并行处理单元(PPU)202，每个并行处理单元都和本地并行处理(PP)存储器204相连接。通常，并行处理子系统包括U个PPU，其中U≥1。(在这里，具有类似目标的多个例子用表明该目标的参考数字表示，如果需附加说明的数字表明该例子)。PPU202和并行处理存储器204可以例如使用一个或多个集成电路装置来实现。所述集成电路装置例如是可编程处理器、专用集成电路(ASIC)和存储器装置。

[0022] 如所示出的PPU202(0)细节，每个PPU202包括通过通信路径113和系统100的其他部分通信的主机接口206，该主机接口206与存储器桥105相连接(或是在另一个实施例中直接与CPU102相连接)。在一个实施例中，通信路径113是PCI-E链接，其中如本领域所已知的，每个PPU202都被分配了专用通道。也可以使用其他通信路径。主机接口206产生信息包(或其他信号)以在通信路径113上传送，接收所有来自通信路径113的输入信息包(或其他信号)，并将这些信息包引导至PPU202的适当的部件。例如，和处理任务有关的命令可以被引导至前端单元212，而和存储器操作有关的命令(例如从并行处理存储器204中读取或向其写入)可以被引导至存储器接口214。主机接口206、前端单元212和存储器接口214可以是常规设计，由于其对于本发明并非至关重要的，因此在此省略对其详细的描述。

[0023] 每个PPU202有利地实现高度并行处理器。如示出的PPU202(0)细节，PPU202包括数量为C的内核208，此处C≥1。每个处理内核208能同时执行大量(例如数十个或数百个)线程，其中每个线程是一例程序；以下描述了多线程处理内核208的一个实施例。内核208通过工作分配单元210接收将要执行的处理任务，该工作分配单元从前端单元212接收定义了处理任务的命令。工作分配单元210能执行各种算法来分配工作。例如，在一个实施例中，工作分配单元210从每个内核208接收“就绪”的信号，该信号表明该内核是否有足够的资源来接受新的处理任务。当一个新的处理任务到达时，工作分配单元210将该任务分配给发出该就绪信号的内核208；如果没有内核208发出就绪信号，则工作分配单元210保留该新处理任务直到某个内核208发出就绪信号。本领域普通技术人员会认识到也可以使用其他算法，且工作分配单元210是以何种特定方式分配输入的处理任务对本发明来说并非是至关重要的。

[0024] 内核208和存储器接口214进行通信，以从各种外部存储器装置中读取或向各种外部存储器装置中写入数据。在一个实施例中，存储器接口214不仅包括适配于和本地并行处理存储器204进行通信的接口，还包括到主机接口206的连接，因此使内核208能和系统存储器104或其他不属于PPU202本地的存储器进行通信。存储器接口214可以是常规设计，在此省略对其的详细描述。

[0025] 内核208能被编程以执行与各种广泛的应用相关的处理任务，所述应用包括但不限于线性和非线性数据变换、视频和/或音频数据的过滤、建模运算(例如，应用物理法则来确定物体的位置、速度和其他属性)、图像渲染操作(例如，顶点着色器、几何着色器和/或像素着色器编程)等等。并行处理单元202可以将数据从系统存储器104和/或本地并行处理存储器204传输到内部(片上)存储器，对数据进行处理，并将结果数据写回到系统

存储器 104 和 / 或本地并行处理存储器 204 中, 在这些存储器中这些数据能够由包括例如 CPU 102 或另一并行处理子系统 112 的其他系统部件来访问。

[0026] 再次参考图 1, 在一些实施例中, 在并行处理子系统 112 中的一些或者全部的并行处理单元 202 是具有渲染流水线的图形处理器, 其能被配置成执行与下列情形有关的各种任务: 由 CPU 102 和 / 或系统存储器 104 通过存储器桥 105 和总线 113 提供的图形数据产生像素数据; 与本地并行处理存储器 204(其能被用作图形存储器, 例如包括常规的帧缓冲器) 交互以储存并更新像素数据; 传送像素数据给显示装置 110; 及其他类似情形。在一些实施例中, 并行处理子系统 112 可包括一个或多个作为图形处理器工作的并行处理单元 202 和一个或多个用于通用目的计算的其他并行处理单元 202。这些并行处理单元 202 可以是相同或不同的, 每个并行处理单元 202 可以有其自己专用的并行处理存储装置 204 或非专用的并行处理装置。

[0027] 在操作中, CPU 102 是系统 100 的主处理器, 控制和协调其他系统部件的工作。特别是, CPU 102 发出控制并行处理单元 202 工作的命令。在一些实施例中, CPU 102 将对于每个并行处理单元 202 的命令流写入到推入缓冲器(图 1 中未示出), 该推入缓冲器可以位于系统存储器 104 中、并行处理存储器 204 中、或是 CPU 102 和并行处理单元 202 都可以访问的其他存储位置中。并行处理单元 202 从推入缓冲器中读出命令流, 并且相对于 CPU 102 的工作异步地执行这些命令。因此, 并行处理单元 202 可被配置成使 CPU 102 卸载处理以增加系统 100 的处理吞吐量和 / 或性能。

[0028] 可以理解的是, 这里示出的系统只是示意性的, 可以对其进行变化和修改。连接拓扑结构包括桥的数目和排列, 可以根据需要改变。例如, 在一些实施例中, 系统存储器 104 直接与 CPU 102 相连接而不是通过一个桥相连接, 其他装置通过存储器桥 105 和 CPU 102 与系统存储器 104 相通信。在其他可选择的拓扑结构中, 并行处理系统 112 与 I/O 桥 107 相连接或直接与 CPU 102 相连接, 而不是和存储器桥 105 相连接。在其他实施例中, I/O 桥 107 和存储器桥 105 可以被集成到单一芯片上。这里示出的特定部件是任选的; 例如, 可以支持任何数目的外插卡或外设。在一些实施例中, 省去了开关 116, 网络适配器 118 和外插卡 120、121 直接和 I/O 桥 107 相连接。

[0029] 并行处理单元 202 和系统 100 剩余部分的连接也同样可以变化。在一些实施例中, 并行处理系统 112 实现为一个外插卡, 其能插入到系统 100 的扩展槽中。在其他实施例中, 并行处理单元 202 能和总线桥一起集成到单一芯片上, 总线桥例如可以是存储器桥 105 或 I/O 桥 107。在另外其他一些实施例中, 并行处理单元 202 的一些或全部组件可以连同 CPU 102 一起集成到单一芯片上。

[0030] 并行处理单元可以设置有任何数量的本地并行处理存储器, 而不包括本地存储器, 也可以使用本地存储器和系统存储器的任意组合。例如, 在统一存储器架构(UMA)的实施例中, 并行处理单元 202 可以是图形处理器, 在这些实施例中, 将会设置极少或者不设置任何专用图形(并行处理)存储器, 并行处理单元 202 将只使用或几乎只使用系统存储器。在 UMA 实施例中, 并行处理单元 202 可被集成到桥芯片或处理器芯片中, 或者是设置为具有高速链接(例如 PCI-E)的分离的芯片, 该高速链接将并行处理单元通过例如桥芯片与系统存储器相连接。

[0031] 如上所述, 并行处理子系统可以包括任意数量的并行处理单元 202。例如, 多个并

行处理单元 202 可以设置在单个的外插卡上,或者多个外插卡可以与通信路径 113 相连,或者一个或多个并行处理单元 202 可以集成到桥芯片中。多并行处理单元系统中的并行处理单元可以是相同的,也可以是彼此不同的;例如,不同的并行处理单元可以具有不同数目的内核、不同数目的本地并行处理存储器及其他类似部件。当存在有多个并行处理单元 202 时,这些并行处理单元 202 可以以高于单个并行处理单元 202 可能达到的数据吞吐量来并行工作以处理数据。包含有一个或多个并行处理单元 202 的系统可以实现为各种配置和形式,包括桌上型电脑、笔记本电脑、或是手持个人电脑、服务器、工作站、游戏控制台、嵌入式系统等等。

[0032] 内核概述

[0033] 图 3 是根据本发明一个或多个方面的图 2 所示并行处理子系统 112 的内核 208 的框图。并行处理单元 202 包括配置成并行执行多个线程的一个内核 208(或多个内核 208),在此术语“线程”是指一例环境,即对一组特定的输入数据执行的特定程序。在一些实施例中,单指令多数据 (SIMD) 指令发出被用来支持并行执行大量线程,而不用提供多个独立的指令单元的技术。

[0034] 在一个实施例中,每个内核 208 包括 P(例如 8、16 等) 个并行处理引擎 302 的阵列,该并行处理引擎 302 配置成从单一指令单元 312 中接收 SIMD 指令。每个处理引擎 302 有利地包括一组相同的功能性单元(例如算术逻辑单元等)。该功能性单元可排列成流水线,允许新指令在完成前个指令之前发出,如同现有技术中公知的那样。可以使用任何功能单元的组合。在一个实施例中,功能性单元支持各种运算,包括整数和浮点运算(例如加法和乘法)、比较运算、布尔运算(AND、OR、XOR)、移位运算和各种代数函数的运算(例如平面插值、三角、指数和对数函数等);并且同样的功能性单元的硬件可进行改变以用于执行不同的运算。

[0035] 每个处理引擎 302 使用本地寄存器文件(LRF)304 的空间来存储它的本地输入数据、中间结果和类似数据。在一个实施例中,本地寄存器文件 304 物理上或逻辑上被分成 P 个通道,每个通道具有一些项目(每个项目可以储存例如 32 位字)。一个通道被分配给每个处理引擎 302,不同通道的相应项目可以用于为执行相同程序的不同线程来容纳数据以促进 SIMD 的执行。在一些实施例中,每个处理引擎 302 只能访问分配给它的通道中的 LRF 项目。本地寄存器文件 304 中项目的总数目优选足够大以使每个处理引擎 302 同时支持多个线程。

[0036] 每个处理引擎 302 也能访问片上共享存储器 306,该片上存储器 306 被内核 208 上所有的处理引擎 302 所共享。共享存储器 306 可以依需要尽可能的大,在一些实施例中,任何处理引擎 302 能以(例如可与访问本地寄存器文件 304 相比)同样低的延迟对共享存储器 306 的任何位置进行读取或写入操作。在一些实施例中,共享存储器 306 实现为共享寄存器文件;在其他实施例中,共享存储器 306 可以使用共享缓存器得以实施。

[0037] 除共享存储器 306 之外,一些实施例还提供另外的片上参数存储器和 / 或缓存器 308,它们可以实现为例如常规 RAM 或缓存器的形式。参数存储器 / 缓存器 308 能被用来例如保持多个线程可能需要的状态参数和 / 或其他数据(例如各种常数)。处理引擎 302 也可通过存储器接口 214 访问片外“全局”存储器,并且系统存储器 104 通过主机接口 206 来访问,该“全局”存储器可以包括例如并行处理存储器 204 和 / 或系统存储器 104。可以理

解的是任何在并行处理单元 202 之外的存储器都可以用作全局存储器。

[0038] 在一个实施例中，每个处理引擎 302 都是多线程的，并能同时执行多达 G(例如 24) 个线程，这可通过例如在本地寄存器文件 304 中其被分配的通道的不同部分中维持与每个线程有关的当前状态信息来实现。处理引擎 302 有利地被设计成从一个线程快速切换到另一个线程，以使不同线程的指令可以以任意顺序发出而不会降低效率。由于每个线程可以对应不同的环境，因此随着为每个周期发出不同的线程，可以在多个周期上处理多重环境。

[0039] 指令单元 312 设置成对于任何给定的处理周期，一条指令 (INSTR) 会被发送给 P 个处理引擎 302 的每一个。当多重环境被同时处理时，每个处理引擎 302 可在任何给定的处理周期中接收不同指令。当所有 P 个处理引擎 302 处理单一环境时，内核 208 实现 P 路 SIMD 微架构。因为每个处理引擎 302 同样被进行同时支持高达 G 个线程的多线程化，因此在这个实施例中的内核 208 能同时执行 $P \times G$ 个线程。例如，如果 $P = 16$ 并且 $G = 24$ ，则内核 208 对单一环境能同时支持高达 384 个线程，或是对每个环境同时支持 $N \times 24$ 个线程，这里 N 为分配给该环境的处理引擎 302 的数目。

[0040] 内核 208 的运行通过工作分配单元 200 进行有利的控制。在一些实施例中，工作分配单元 200 接收将要被处理的数据（例如图元数据、顶点数据和 / 或像素数据）的指针以及包含数据或定义数据将要怎样处理（例如要执行什么程序）的指令的推入缓冲器的位置。工作分配单元 210 能将要被处理的数据载入到共享存储器 306 中，并将参数载入到参数存储器 308 中。工作分配单元 210 还初始化指令单元 312 中的每个新环境，然后发出信号以指示指令单元 312 开始执行该环境。指令单元 312 读取指令推入缓冲器并执行指令以产生处理后的数据。当环境执行完成后，内核 208 将有利地通知工作分配单元 210。工作分配单元 210 然后可以初始化其他进程，以便例如从共享存储器 306 中检索输出数据和 / 或准备内核 208 以执行另外的环境。

[0041] 可以理解的是，此处描述的并行处理单元和内核架构仅仅是示意性的，可对其进行变化和修改。可以包括任意数目的处理引擎 302。在一些实施例中，每个处理引擎 302 有其自己的本地寄存器文件，并且每个线程中本地寄存器文件项目的分配可以是固定的，也可以随需要来设置。特别地，本地寄存器文件 304 的项目可被分配用于处理每个环境。进一步地，虽然只示出了一个内核 208，但并行处理单元 202 可以包括任意数目的内核 208，这些内核 208 有利地具有彼此相同的设计，因此执行方式不依赖于哪个内核 208 接收到特定的处理任务。每个内核 208 有利地独立于其他内核 208 而运行，并具有其自己的处理引擎、共享存储器等等。

[0042] 图形流水线架构

[0043] 图 4 是根据本发明一个或多个方面的图形处理流水线 400 的概念图。并行处理单元 202 可被设置为形成图形处理流水线 400。例如，内核 208 可被设置成执行顶点处理单元 444、几何处理单元 448 和片段处理单元 460 中的一个或多个单元的功能。数据汇编器 442、图元汇编器 446、光栅化器 455 和光栅运算单元 465 的功能也可以由内核 208 来执行。作为选择，图形处理流水线 400 可通过在顶点处理单元 444、几何处理单元 448、片段处理单元 460、数据汇编器 442、图元汇编器 446、光栅化器 455 和光栅运算单元 465 中的一个或多个单元上使用专用处理单元来实现。

[0044] 数据汇编器 442 是为高阶表面、图元和其他类似物收集顶点数据并向顶点处理单

元 444 输出顶点数据的处理单元。顶点处理单元 444 是设置成执行顶点着色器程序的可编程执行单元,按照该顶点着色器程序的规定对顶点数据进行变换。例如,顶点处理单元 444 可被编程为将用基于物体的坐标表示(物体空间)转换成基于可选择的其他坐标系统,例如世界空间或规范化设备坐标(NDC)空间。顶点处理单元 444 可以读取存储在并行处理存储器 204 或系统处理器 104 中的数据,以在处理顶点数据时使用。

[0045] 图元汇编器 446 从顶点处理单元 444 中接收处理过的顶点数据并组建图形图元,例如点、线、三角或其他,以用于由几何处理单元 448 来处理。几何处理单元 448 是配置用于执行几何着色器程序的可编程执行单元,其将从图元汇编器 446 接收的由几何着色器程序指定的图形图元进行变换。例如,几何处理单元 448 可被编程为将图形图元再细分为一个或多个新图形图元,并计算用于光栅化该新图图元的参数,例如平面平衡系数。在本发明的一些实施例中,几何处理单元 448 可以在几何数据流中增加或删除元件。几何处理单元 448 输出为光栅化器 455 或为存储器接口 214 指定新图形图元的参数和顶点数据。几何处理单元 448 可读取存储在并行处理存储器 204 或系统存储器 104 中的数据以用来处理几何数据。

[0046] 光栅化器 455 扫描用来转换新的图形图元并输出片段和覆盖数据给片段处理单元 260。当抗锯齿被用于产生图像数据时,光栅化器 455 被设置成产生子像素样本覆盖数据。当使用混合抗锯齿处理时,可以驻留在光栅化器 455 中的混合抗锯齿控制单元 500 设置成确定通过用于处理每个图元的片段处理单元 460 的通道数目,如结合图 5C 和图 6 所说明。

[0047] 片段处理单元 460 是设置成执行片段着色器程序的可编程执行单元,并将从光栅化器 455 接收的由片段着色程序指定的片段进行变换。例如,片段处理单元 460 可被编程以执行例如透视修正、纹理、阴影化、混合等等的操作来产生输出到光栅运算单元 465 的着色的片段。片段处理单元 460 能读取存储在并行处理存储器 204 或系统存储器 104 中的数据以便在处理片段数据时使用。根据混合抗锯齿控制单元选择的采样率,可以在像素、样本或者超采样簇颗粒度的程度上对片段进行着色。

[0048] 存储器接口 214 对存储在图形存储器中的数据产生读取请求,并执行纹理过滤操作,例如双线性、三线性、各向异性等。在本发明的一些实施例中,存储器接口 214 可以设置成能解压缩数据。特别地,存储器接口 214 可被设置成对固定长度的块编码数据进行解压缩,例如以 DXT 格式表示的压缩数据。光栅运算单元 465 是执行诸如模板、Z 测试等光栅操作的处理单元,并输出像素数据作为经处理的图形数据以存储在图形存储器中。经处理的图形数据可以存储在图形存储器(例如并行处理存储器 204)和/或系统存储器 104 中,以显示在显示装置 110 上,或由 CPU 102 或并行处理子系统 112 进一步处理。在本发明的一些实施例中,光栅运算单元 465 设置成能压缩写到存储器中的 Z 或颜色数据,并能解压缩从存储器读取的 Z 或颜色数据。

[0049] 混合抗锯齿处理

[0050] 如前所述,并行处理单元 202 可被设置成以各种采样率执行着色,以改善图像质量或改善着色性能。混合抗锯齿控制单元决定用于对图元中每个像素进行着色的着色器通道数目。每个像素中一个或多个多重样本(子像素样本)的超采样簇通过内核 208 处理,内核 208 为每个通道设置为片段处理单元 460 以产生单一着色颜色值,该颜色值被复制用

于超采样簇中所有的多重样本。在场景被渲染后，超采样簇的样本被结合用于产生抗锯齿图像。

[0051] 每个图元的子像素样本和着色器通道的数目被增加以改善图像质量。当应用程序被启动且对于渲染目标（图像缓冲器）的每个像素都保持一致时，子像素样本的数目被确定。混合抗锯齿控制单元可基于渲染状态动态确定着色通道的数目，该渲染状态例如可以为 alpha 测试启用 / 关闭、纹理贴图内容、用户提供的质量 / 性能控制等等。

[0052] 图 5A 示出了根据本发明一个或多个方面的像素 501 中的超采样簇 503 和 511 以及多重样本 502、504 和 513。当使用 8 个子像素样本抗锯齿化时，可以使用多重样本和超采样簇的各种不同组合来产生 8 个子像素样本。在图 5A 所示的例子中，对于像素 501 中总共 8 个子像素样本的位置，三个超采样簇 503 和超采样簇 511 每个都包括两个多重样本，例如在超采样簇 511 中的多重样本 502 和 504。其他 8 个子像素样本的配置包括多达 8 个超采样簇，每个有一个多重样本，或 8 个多重样本仅有 1 个超采样簇。每个超采样簇实施一次着色，着色后的值（例如颜色）被存储以便供该超采样簇中的所有多重样本使用。

[0053] 着色器属性可在超采样簇中特定的多重样本位置处被采样，或者在超采样簇中或接近超采样簇的其他位置处采样。例如，图 5A 中片段属性（颜色、纹理坐标或其他）可在实体多重样本位置处采样，例如超采样簇 511 中的多重样本 502。另外，当片段仅仅部分覆盖超采样簇时，调整属性被采样的位置以位于超采样簇中被覆盖的多重样本区域内是有利的。这通常被称为质心采样，但是该术语在这里应用于超采样簇而不是整个像素片段。

[0054] 图 5B 示出了根据本发明一个或多个方面的超采样簇 511 中的片段 509 和质心位置 517。在本发明的一些实施例中，质心采样用来修改属性被评估的位置，以更好地对应于被片段实际覆盖的显示屏区域。在本发明的一些实施例中，样本插值单元 510 可以设置成在特定多重样本位置处或在近似质心位置处对每个超采样簇进行采样。

[0055] 质心可以是被覆盖的多重样本的几何质心，或者例如通过选择超采样簇中最接近被完全覆盖的超采样簇的质心的被覆盖多重样本来近似确定该质心。例如，质心位置 517 是计算出的在超采样簇 511 的几何中心的多重样本位置，由于多重样本 502 的位置接近边缘而不是接近片段 509 的中心，所以该质心位置用来代表超采样簇 511 的采样颜色。着色值在质心位置 517 处被计算出来，以相对于多重样本 502 更加精确地代表片段颜色。

[0056] 图 5C 是根据本发明一个或多个方面的图形处理流水线 400 一部分的框图，图形处理流水线 400 包括光栅化器 455、片段处理单元 460 和光栅运算单元 465。其他处理单元也可包括在光栅化器 455、片段处理单元 460 和光栅运算单元 465 中。由于这些其他处理单元可以是普通的常规设计，因此没有在图 5C 中示出。由于对本发明并非至关重要，省略对它们的详细描述。

[0057] 光栅化器 455 从几何处理单元 448 中接收图元并为与图元相交叉的每个像素产生片段。混合抗锯齿控制单元 500（可选地置于光栅化器 455 中）可设置成根据渲染状态，例如 Alpha 测试开启 / 关闭、纹理贴图内容、用户提供的质量 / 性能控制等等，来动态确定用于处理每个图元片段的着色器通道的数目。

[0058] 混合抗锯齿控制单元 500 通过对图元实施更多的得益于更高着色率的着色通道并降低其他图元的着色率，提高了抗锯齿处理的效率。混合抗锯齿控制单元 500 可通过用户、应用程序或装置驱动器 103 来设置，以便运行在各种品质设定中。这些品质设定可以从

“一直多重采样”的最低品质设定到“一直超采样”的最高品质设定。中间品质设定可在确定着色通道数目时考虑渲染流水线状态。例如,如果 Alpha 检测或者着色器像素消除启动时,就需要有更多的着色通道。相反地,当指定高性能时,Alpha 检测和着色器像素消除关闭,混合抗锯齿控制单元 500 可以降低采样率。混合抗锯齿控制单元 500 在确定着色通道数目时也可以考虑像素着色器的特征或纹理采样器的设定。本领域的普通技术人员可以认识到,混合抗锯齿控制单元 500 可使用各种广泛的标准来确定着色通道的数目。在常规图形系统中,对场景中的所有图元都基于用户提供的或者固定的设置来确定采样率。另外,常规系统的采样限定于多重采样或超采样,而不是中间的可选择方式。

[0059] 在一个实施例中,光栅化器 455 产生 2×2 的四方像素片段,其通过混合抗锯齿迭代单元 515 接收。当混合抗锯齿控制单元 500 设定通道数等于 1 时(即当多重采样时),混合抗锯齿迭代单元 515 原封不动地将这些四方像素片段传递给片段处理单元 460。但是,当混合抗锯齿控制单元 500 设定通道数 N 大于 1 时,混合抗锯齿迭代单元 515 为片段处理单元 460 多次输出每个四方像素片段,包括相应于着色器通道的通道数目。混合抗锯齿迭代单元 515 可以屏蔽传送到片段处理单元 460 的覆盖部分,这样只有超采样簇中和当前通道对应的多重样本被开启。在其他的实施例中,片段处理单元 460 可以基于由混合抗锯齿迭代单元 515 提供给它的通道数目来屏蔽覆盖部分。值得注意的是,其他实施例可在 2×2 段的四方像素片段以外的区域上迭代,例如可以是单一像素、 4×4 片段拼贴或其他。在像素(四方)区域上而不是图元上进行迭代可以是有利的,这是因为纹理贴图数据很可能再次用于随后的特定四方片段的着色器通道,而在图元上进行迭代(可能会很大)会造成再次从诸如并行处理存储器 204 或系统存储器 104 的存储器中获取纹理数据。

[0060] 重要的是,并没有对每个着色器通道重复进行需要用来产生片段的几何计算。相反,惯常使用样本蒙板向多重样本缓冲器中进行超采样的系统通常为每个着色通道重复进行几何计算。注意到在片段处理单元 460 中采样的图元属性只需要计算一次,而不管混合抗锯齿通道的数目如何,这是由于它们将被随后迭代的四方像素片段所引用然后被丢弃。

[0061] 片段处理单元 460 中的样本查找表使用混合抗锯齿参数和通道数目来确定插值片段参数被采样的位置。样本查找表 505 可为每个超采样簇选择质心位置或多重样本位置。多重样本位置被输出到样本插值单元 510,该样本插值单元 510 用来为每个超采样簇计算一个或多个例如颜色通道(红、绿、蓝、Alpha)的插值参数、纹理坐标等,即对于四方像素片中的每个像素计算一组插值参数。着色器 520 为四方像素片中的每个像素处理该组插值参数,使用本领域普通技术人员公知的技术来执行片段着色器程序或类似程序,并为每个超采样簇产生着色像素值,例如颜色。

[0062] 在着色过程中,由于 alpha 检测或者着色器像素消除,每个超采样簇的子像素样本可能会被除去(剔除或消除),以至于基于像素消除或 alpha 检测结果来修改光栅产生的覆盖范围,以产生后着色器覆盖范围。由于超采样簇是通过着色器 520 在单独的通道中被处理的,因此超采样簇可以在 alpha 检测期间被个别地消除。相反地,当常规的多重采样是用来处理单一着色通道中的所有子像素样本时,所有子像素样本或者被保留或者被消除,这就导致了产生较低图像品质的粗糙的 alpha 检测颗粒度。

[0063] 着色器 520 将着色的像素值和子像素覆盖范围(相对于光栅化器 455 提供的覆盖范围可能被修改)分别输出给颜色缓冲器 535 和覆盖集合器 530。覆盖集合器 530 为每个

着色器通道集合后着色器覆盖范围,以为每个像素产生集合的覆盖范围信息。颜色缓冲器 535 为每个像素集合着色值。当接收到最后着色器通道的着色值时,集合的覆盖信息被输出到光栅运算单元 465。四方像素片段的着色值可连同集合的覆盖范围信息一起输出或随后输出,例如在光栅运算单元 465 完成 z 测试之后。在本发明的其他实施例中,可以省略覆盖集合器 530 和颜色缓冲器 535。

[0064] 在将每个像素的样本在存储器中打包的系统中,覆盖集合和合并颜色值到颜色缓冲器中是有利的,因此多个样本能使用单一存储器处理来写入或读取。其他实施例可以省略覆盖集合器 530。在其中不在存储器中连续地存储像素的样本值的系统中,覆盖集合器 530 可能不那么有利。

[0065] 光栅运算单元 465 中作为可选的 z/ 颜色压缩单元 550 接收集合覆盖信息和 z 值或其他代表片段(跟随 z 测试)的 z 或深度值,并为一个区域的像素产生压缩 z 值。z/ 颜色压缩单元 550 也可以接收片段的集合颜色值,并为一区域的像素产生压缩颜色值。当应用到更大群的像素时,可以改善压缩。因此,若干个四方像素片段可被集合在一起并在结果被压缩之前进行 z 测试。重要的是,混合抗锯齿不会妨碍或降低 z 压缩的效率。z 压缩在降低访问 z 缓冲器和一些实施例中的存储器足迹所需的存储器带宽方面是有利的。

[0066] 图 6 是根据本发明一个或多个方面的实施混合抗锯齿处理的方法流程图。在步骤 610,混合抗锯齿控制单元 500 接收图元。步骤 615,混合抗锯齿控制单元 500 确定混合抗锯齿是否启用,如果未启用则使用惯常的抗锯齿方法来处理片段。如果在步骤 615 中混合抗锯齿被启用,在步骤 635 中混合抗锯齿控制单元 500 为图元确定混合抗锯齿参数。更具体地,当对与图元交叉的每个像素着色时,混合抗锯齿控制单元 500 确定要使用的超采样簇(着色器通道)的数目。

[0067] 在步骤 640,光栅化器 455 为图元的覆盖部分产生样本水平覆盖范围。这个覆盖范围的颗粒度可以是粗糙或细致的,但是至少是四方像素片段的大小。光栅化器 455 将与图元交叉的四方像素的覆盖信息输出到混合抗锯齿迭代单元 515。混合抗锯齿迭代单元 515 基于混合抗锯齿参数来扩大每个四方像素,以在多个通道中对四方像素进行着色。根据覆盖信息,当超采样簇中所有的多重样本都没有被覆盖时,混合抗锯齿迭代单元 515 可设置成跳过着色器通道。在步骤 643 中,混合抗锯齿迭代单元 515 确定通道数目(第一、第二等),并将四方像素和通道数目输出到片段处理单元 460。如前所述,当通道数目大于 1 时,混合抗锯齿迭代单元 515 可以屏蔽覆盖信息。样本查找表 505 使用通道数目和多重样本的数目来编制索引,以为多重样本位置读取编程后的值,包括在用来插值片段参数的超采样簇中的位置指示。插值参数通过样本插值单元 510 来为超采样簇进行计算。

[0068] 步骤 645 中,片段处理单元 460 对四方像素进行着色,为每个超采样簇产生着色值,即在四方像素中每个像素有一个着色值。在超采样簇中,着色值用于每个被图元覆盖的多重样本。片段处理单元 460 也为四方像素输出后着色器覆盖范围。后着色器覆盖范围可以和光栅化的像素覆盖信息不同,这是因为多重样本可能如前所述在着色期间被消除。

[0069] 在步骤 650 中,混合抗锯齿迭代单元 515 确定是否有另外的着色器通道被用来处理四方像素,如果说有的话,则步骤 643 和步骤 645 将会对其他的着色器通道(第二、第三等)重复进行。如果在步骤 650 中,混合抗锯齿迭代单元 515 确定在处理四方像素中不需要其他的着色器通道,那么步骤 660 中覆盖集合器 530 为每个着色器通道组合这些后着色器覆

盖范围,以为四方像素产生集合的覆盖范围信息。步骤 660 中,覆盖集合器 530 也可以为每个着色器通道组合后着色器颜色值,以为每个四方像素产生集合的颜色值。覆盖集合器 530 可设置成以多个四方像素像素级别来集合后着色器颜色和覆盖范围信息。在步骤 665 中,光栅运算单元 465 执行光栅操作以确定哪个着色值被写入到帧缓冲器中。光栅操作可以以四方像素或多个四方像素的程度来执行。光栅运算单元 465 中的 z / 颜色压缩单元 550 可在 z 和 / 或颜色数据存储到 z 缓冲器和 / 或颜色缓冲器之前为四方像素压缩 z 和 / 或颜色数据。

[0070] 在步骤 670 中,光栅化器 455 确定是否有其他四方像素与图元交叉,如果有,那么在步骤 640 中,光栅化器 455 处理被图元覆盖的不同四方像素。如果在步骤 670 中,光栅化器 455 确定所有和与图元交叉的四方像素都已着色,那么在步骤 675 中,图元处理就完成了。在流水线系统中,图 6 中示出的一个或多个步骤可为不同的四方像素并行实施。

[0071] 基于诸如 alpha 测试开启 / 关闭、纹理贴图内容、用户提供的品质 / 性能控制等渲染状态,混合抗锯齿控制单元 500 可以为每个图元动态确定混合抗锯齿参数,例如每个像素超采样簇的数目。由于得益于高品质抗锯齿处理的图元能用更多的样本来着色,而用较少的样本为其他的图元着色,因此基于渲染状态采用抗锯齿处理改善了效率,优化了图像品质和性能。

[0072] 本发明已参考特定实施例描述如上。但是,本领域技术人员可以理解的是,根据本发明的教导还可以做出更多种的变型和修改,这些变型和修改对仍然没有脱离本发明所附的权利要求界定的本发明的精神和范围。本发明的一个实施例可被实施为软件产品以和电脑系统使用。软件产品的软件定义了实施例的功能(包括本文描述的方法),并可被包括在不同的计算机可读存储媒介上。计算机可读存储媒介例如可包括但不限于:(i) 其上信息被永久存储的不可重写存储媒介(例如,计算机中诸如 CD-ROM 驱动可读的 CD-ROM 盘、闪存、ROM 芯片或其他固态非挥发半导体存取器等只读存储器装置);(ii) 存储可改变信息的可读存储媒介(例如,软盘驱动器或硬盘驱动器或其他任何固态随即存取半导体存储器等软盘)。相应地,前面的描述和附图应被当做示意性的而不是限制性的。

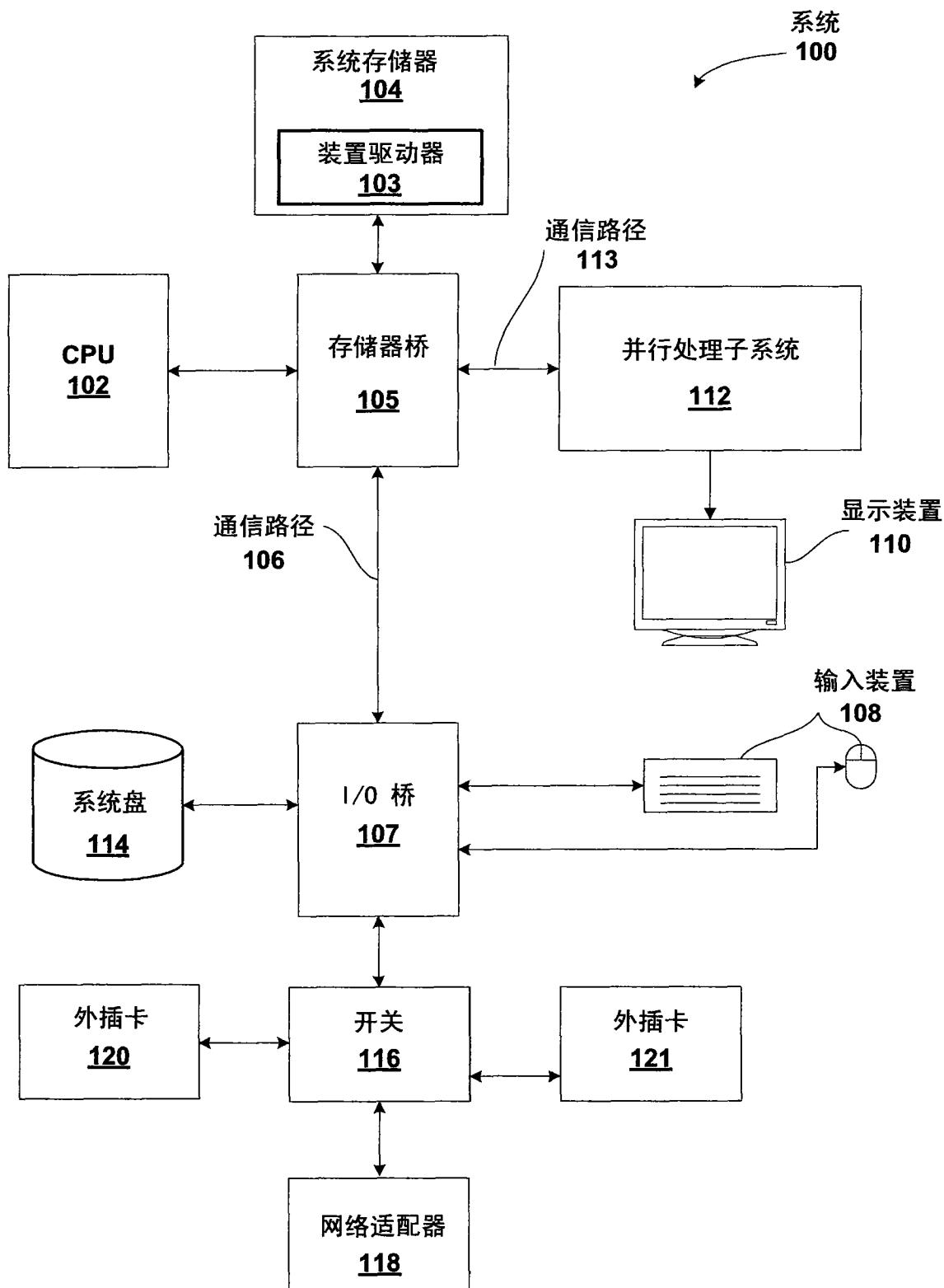


图 1

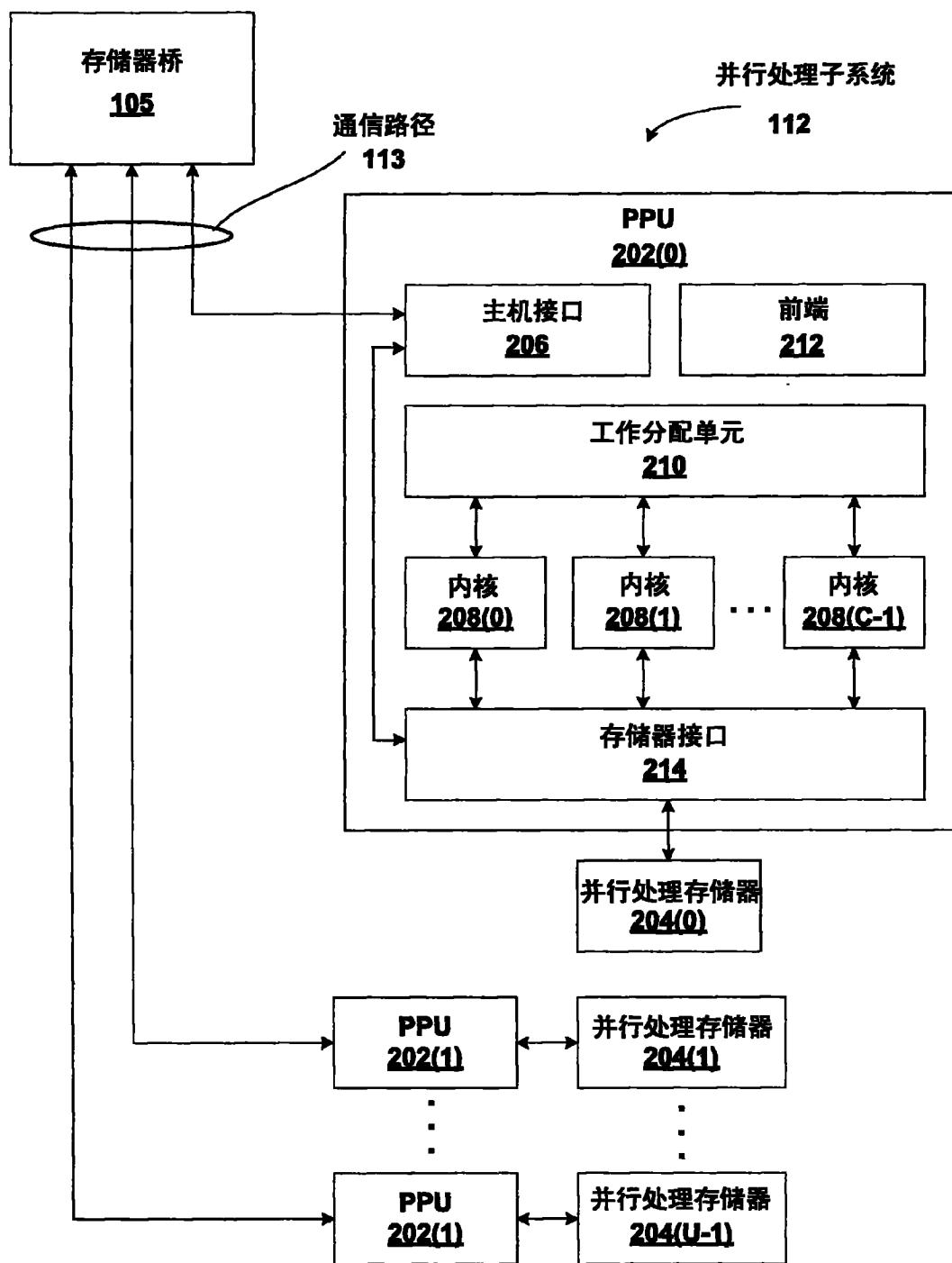


图 2

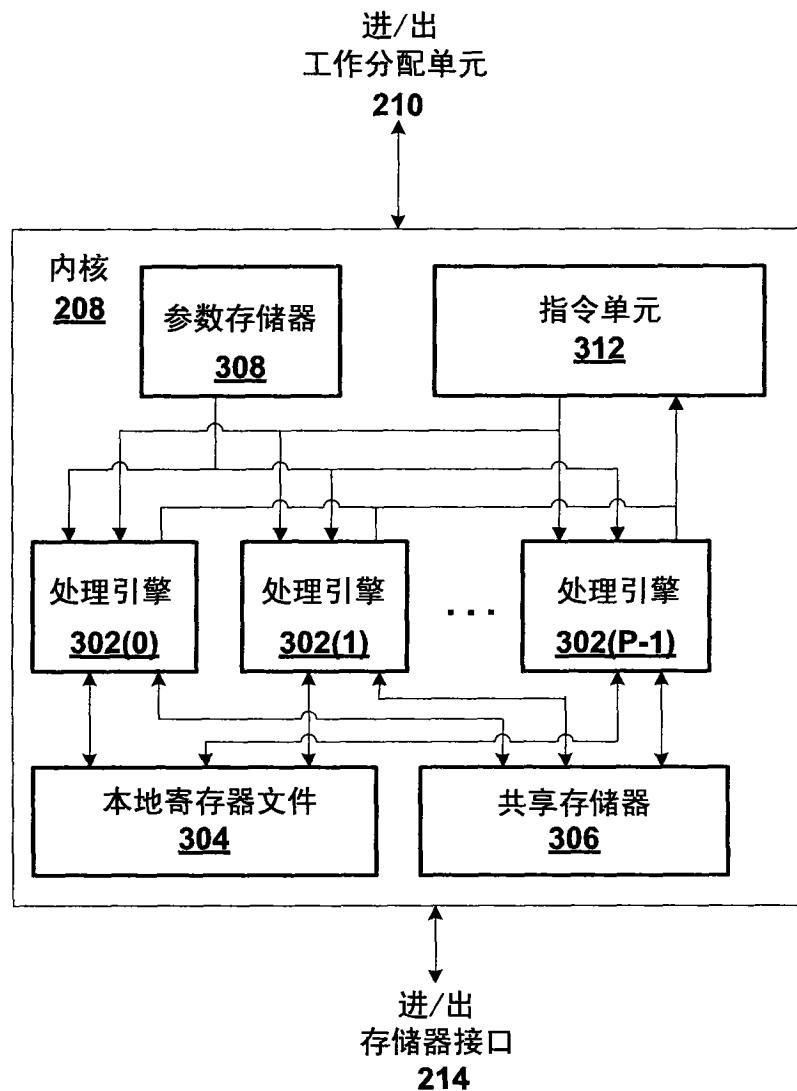


图 3

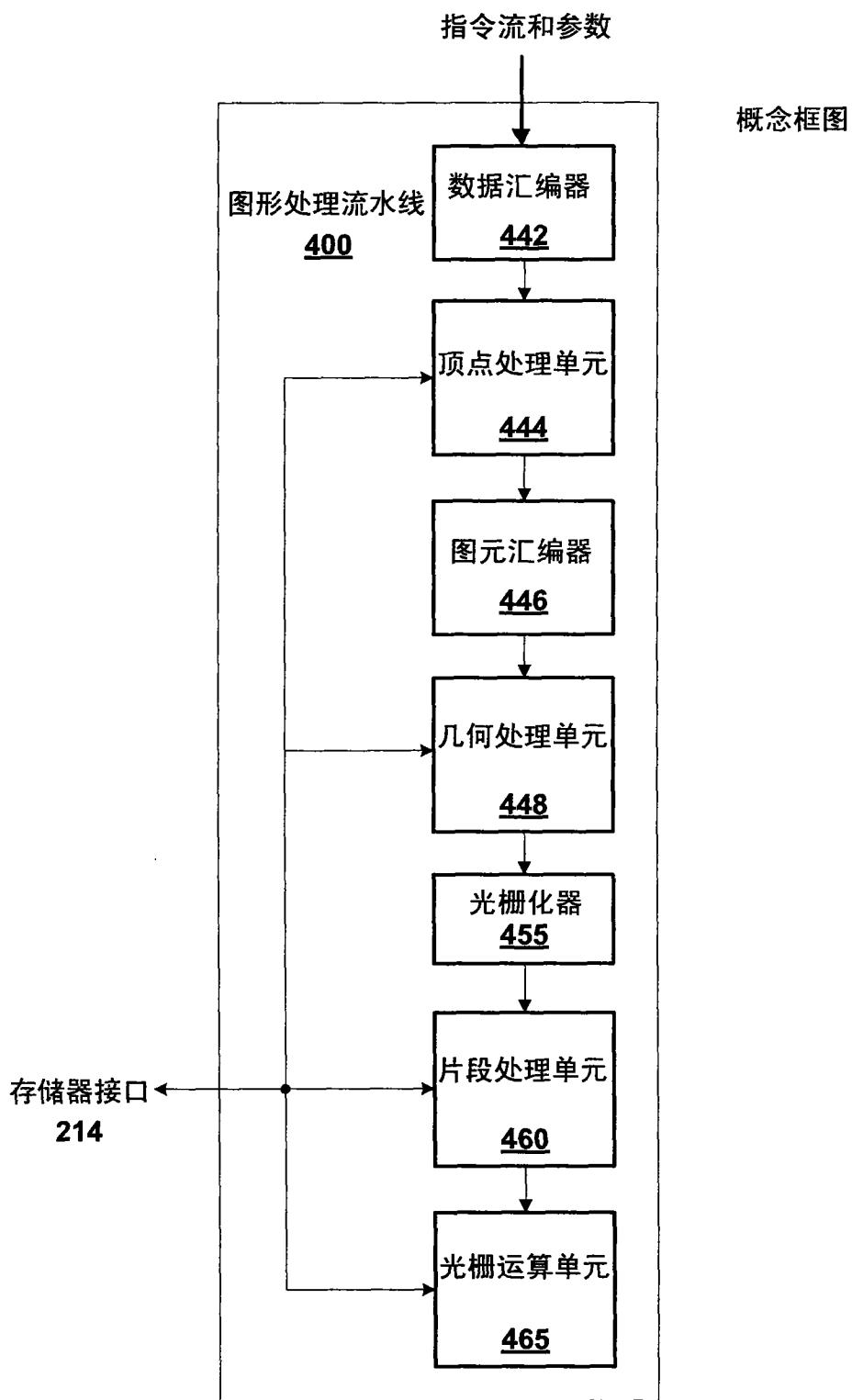


图 4

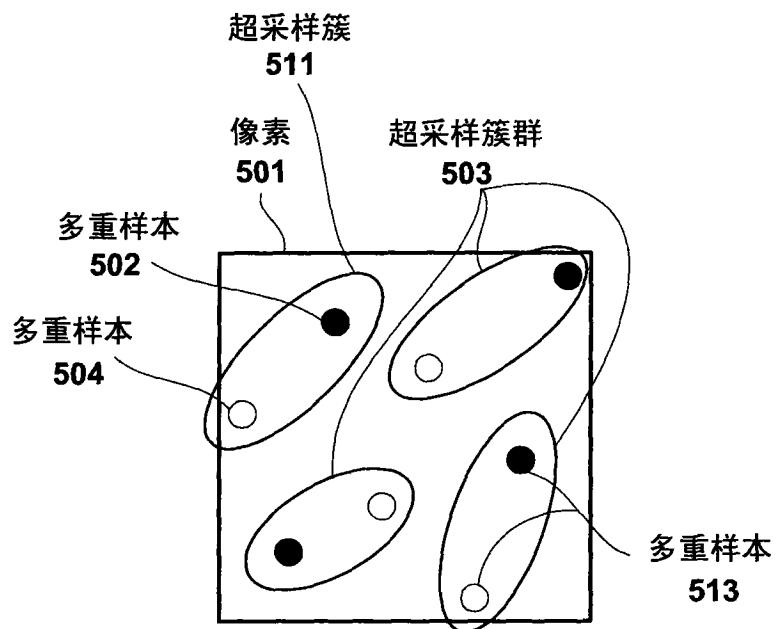


图 5A

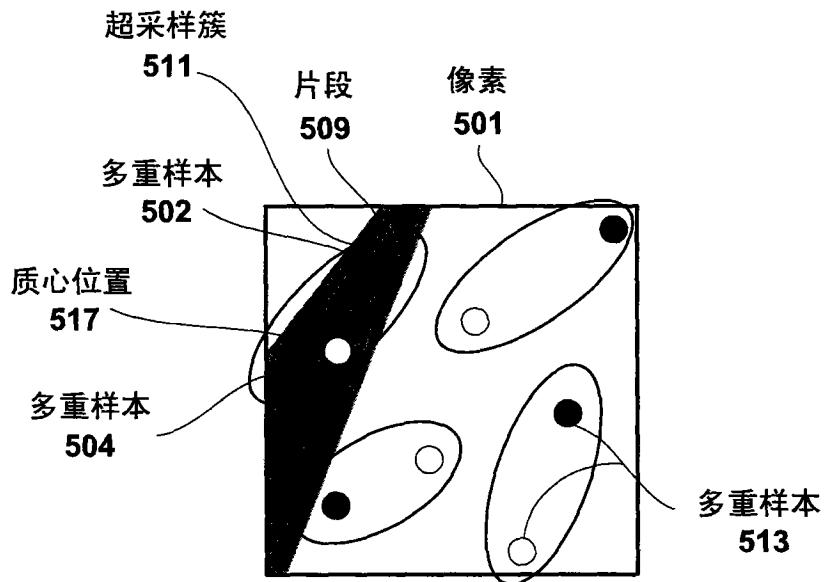


图 5B

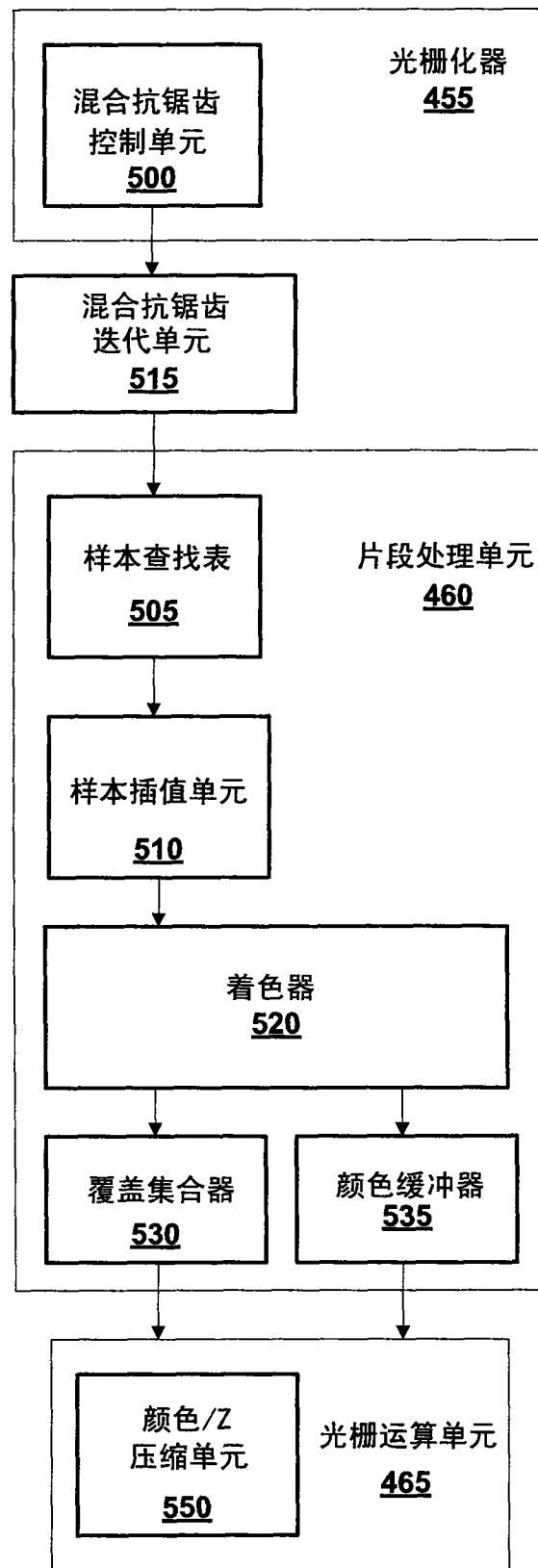


图 5C

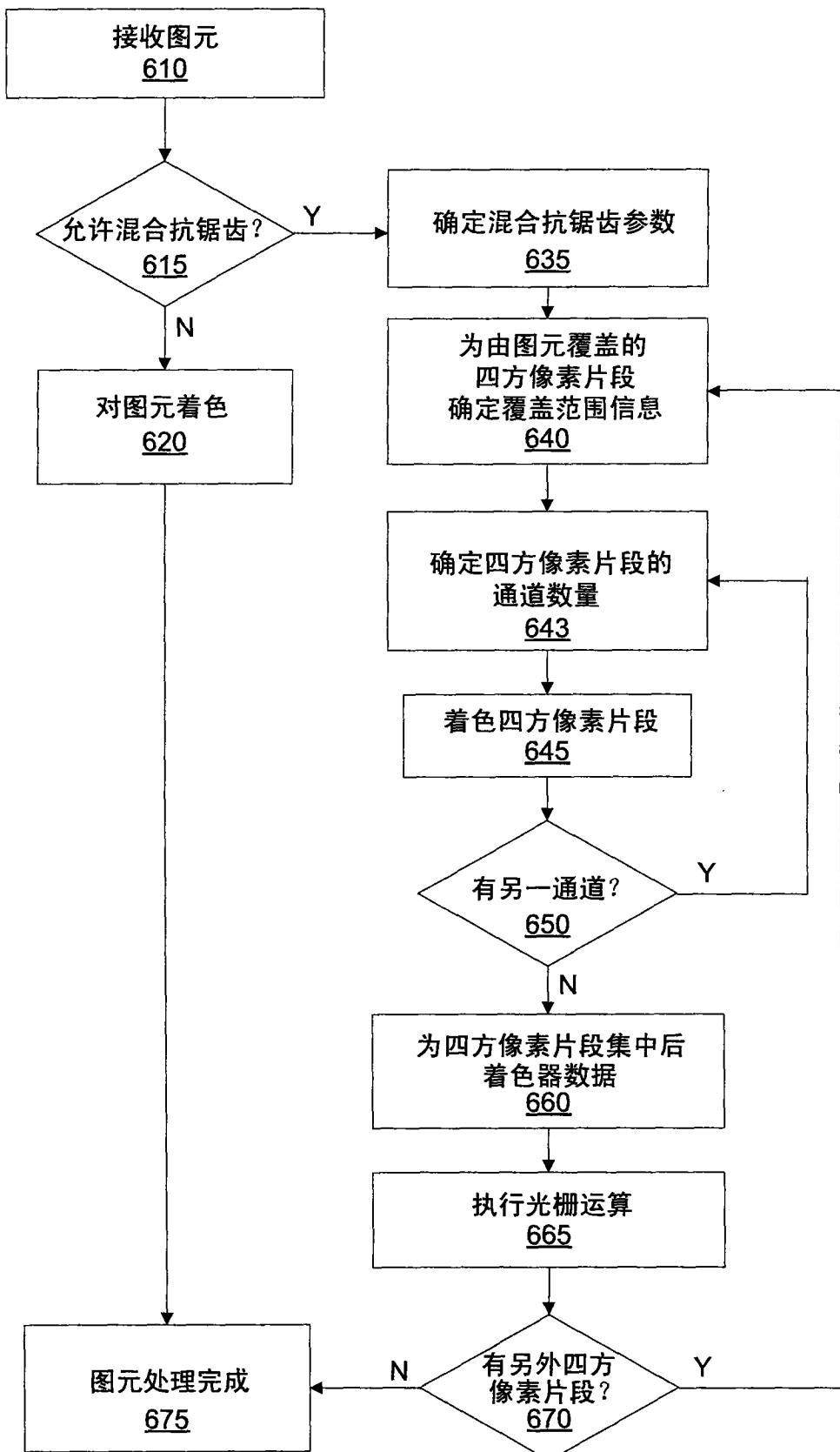


图 6