

發明專利說明書

(本說明書格式、順序，請勿任意更動)

【發明名稱】(中文/英文)

映射多重速率著色至單一程式之技術 / MAPPING
MULTI-RATE SHADING TO MONOLITHIC PROGRAMS

【技術領域】

[0001]本發明係有關於對單片程式之映射多重速率著色技術。

【先前技術】

發明背景

[0002]在3D管線中的動力成本之顯著部分係歸因於像素著色。較高顯示器解析度及圖形品質(尤其在手持型裝置中)與使功率耗散最小化之需求不一致。更有效率地運行像素著色器將因此特別有利。

[0003]粗像素著色利用與表面著色相比幾何閉塞通常帶來較高頻率細節之事實，且按比可見度測試低的速度進行著色。粗像素著色較適合於高像素密度顯示器，其中減小的著色速率之效應僅自正常檢視距離即可辨別。在螢幕的模糊化或另外不大為使用者可察覺之區域(例如，受到運動或散焦模糊影響之區域，或在使用者之視窩視覺外部之區域)中，可進一步減小著色速率。

[0004]難以對電流圖形架構利用此等機會，在電流圖形架構之情況下，著色器按每個像素或每個樣本速率來執行。可藉由變化圖框緩衝器解析度來粗略地控制著色速

率，但此不允許仍耦合至可見度取樣速率的著色速率之更精細粒度級變化(例如，每個目標、每個三角形或每個影像區域)。

[0005]粗像素著色(CPS)為用於在點陣化管線中變化著色速率同時保持可見度取樣速率恆定之架構。

[0006]已提議許多方法來藉由按比可見度取樣速率低的速率取樣著色來改良著色效率。多樣本抗頻疊(MSAA)為常由圖形處理器硬體支援之一項此技術。藉由MSAA，每個像素儲存多個覆蓋樣本(亦叫作可見度樣本)，但僅針對由基元覆蓋之每一像素將像素著色器執行一次。此與每個覆蓋之樣本執行著色器一次的超取樣形成對比。

【發明內容】

[0007]依據本發明之一實施例，係特地提出一種方法，其包含：定義針對像素的一群組之一粗對像素(coarse-to-pixel)映射；進行粗像素著色；以及使用該粗對像素映射在用於該群組之一硬體處理器中進行像素著色。

【圖式簡單說明】

[0008]關於以下諸圖描述一些實施例：

圖1描繪根據一個實施例的粗四元素(quad)像素佈局及其4寬SIMD執行序列；

圖2描繪根據一個實施例的另一粗像素佈局及其4寬SIMD執行序列；

圖3A為根據一個實施例的用於核心處置粗及像素速率著色兩者之流程圖；

圖3B為根據一個實施例的用於固定功能加速器之流程圖；

圖4為後點陣化級之管線抽象化；

圖5描繪用於粗像素著色的圖形管線之一部分；

圖6描繪在4寬SIMD處理器上單一執行緒中排程的具有粗像素四元素之多重速率著色；

圖7為根據一個實施例的資料處理系統之方塊圖；

圖8為根據一個實施例的圖7展示之處理器之方塊圖；

圖9為根據一個實施例的圖7之圖形處理器之方塊圖。

圖10為根據一個實施例的圖形處理引擎之方塊圖；

圖11為根據另一實施例的圖形處理器之方塊圖；

圖12說明用於一個實施例之執行緒執行邏輯；

圖13為根據一個實施例的圖形處理器執行單元指令格式之方塊圖；以及

圖14為用於一個實施例之圖形軟體架構。

【實施方式】

較佳實施例之詳細例示

[0009]在粗像素著色(CPS)中，以解耦方式取樣著色及可見度。像素著色在影像-空間著色柵格中經遲鈍地評估，且臨時地儲存於記憶快取記憶體中，用於再用於規則或隨機點陣化(SR)。在另一粗像素取樣方法中，在最佳化之雷斯(Reyes)架構中的參數修補程式空間中，均勻地取樣著色。此等技術允許具有任意柵格間距之著色柵格，在一些實施例中，此實現著色速率之更靈活控制。

[0010]在多重速率著色中，在一些實施例中，將粗速率

著色階段加入於現有像素速率階段之上以在對影像品質具有最小影響之情況下顯著改良效能。可將以不同速率評估之兩個著色階段映射至在處理器圖形單指令多資料(SIMD)引擎上執行之一個單片程式。單片程式為其執行不取決於另一程式之單階層軟體。在一個實施例中，多重速率著色允許一單一呈現遍次(pass)以一或多個不同速率執行著色程式碼：每個像素群、每個像素及每個樣本。

[0011]新組合指令(至固定功能硬體加速器之介面)實現在於SIMD機器上執行之同一核心程式內程式執行自粗速率至像素速率之快速切換。

[0012]新固定功能邏輯加速了粗對像素映射，在一些實施例中，其具有比管理單獨程式所需要之邏輯低得多的複雜度/成本。同時，使用新邏輯之軟體附加項極少；因此在一些實施例中，其不妨礙單片程式之執行。

[0013]圖 1 例示映射至用於不同粗像素佈局之一個 SIMD 程式的 1×2 像素之粗像素四元素佈局之多重速率著色器執行之一實例。粗像素為共用單一粗像素著色器評估之結果的像素群。圖 2 例示用於 2×2 粗像素四元素佈局的多重速率著色器之一實例。為簡單起見，圖式展示一個 4 寬 SIMD 四元素，但此解決方案適用於映射至較寬 SIMD 機器之任何數目四元素。數字(0、1、2、3)表示粗像素，而大字字母(A、B、C、.....)表示像素。每一圖式之左側例示在點陣化後的像素之 2 維視圖，其中呈較淺色文字之數字/字母表示未點亮像素。右側展示單片程式在 SIMD 機器上的階段之執行次序。

[0014]在第一階段(「粗階段」)中，所有四個SIMD通道針對粗像素四元素執行粗速率指令。在下一個階段(像素階段#1)中，使用同一SIMD通道執行由粗四元素覆蓋的隨後像素四元素群組。像素著色器四元素及其至粗像素之映射的集合取決於粗像素佈局(例如， 1×1 、 1×2 、 2×1 、 2×2 、 2×4 、 4×2 、 4×4)及點陣化遮罩(一些四元素可完全未點亮)。此外，SIMD執行遮罩可對於每一群組而不同。

[0015]在階段一中，在前三個像素經點亮且第四像素未點亮之情況下，執行遮罩為(1、1、1、0)，使得不執行未點亮像素。針對階段#1之粗對像素映射為(0、1、0、1)，粗像素0映射至像素A，粗像素1映射至像素B，像素C亦映射至粗像素0，且像素D映射至粗像素1。對於階段#2，粗對像素映射為2、3、2、3。所有此等數條資訊在著色器編譯時係未知的，若使用現有組合指令之集合在軟體中實施粗對像素過渡，則此將對顯著的軟體附加項有影響。

[0016]在圖1及圖2中之右側的執行序列例示使用新組合指令(粗對像素速率切換)減少與階段轉變相關聯之軟體附加項的單片程式。新指令採用像素group_id作為輸入，且針對每一像素群組，其返回：

pixel_mask—指示在每一群組中哪些像素經點亮
pixel_to_coarse_mapping—指示每一像素之母粗像素，
(用以攪和來自粗階段輸出之階段間資料)
pixel_rate_bary—用以評估輸入屬性的在重心空間中之像素位置
next_group_id—處理的下一個像素群之識別符

[0017]新指令為至固定功能硬體加速器之介面。加速器判定粗四元素至像素四元素之2維映射，且管理此映射之其

可變擴展，此係因為處理的像素四元素之數目隨著粗四元素佈局及點陣化遮罩兩者變化。詳言之，若所有像素未點亮，則加速器抑制整個四元素。藉由此方法，單片著色器程式自特定粗四元素佈局及映射抽象化，且在一迴路中處理像素群，直至硬體加速器通知處理完成(next_group_id = 0)。

```

coarse_and_pixel_rate_shader(attribute_deltas, coarse_rate_bary)
{
    //規則著色器程式碼—與在單一速度解決方案中相同
    (simd) coarse_rate_inputs = interpolate(attribute_deltas, coarse_rate_bary)
    (simd) inter_phase_outputs = coarse_shader(coarse_rate_inputs)

    (1) group_id = 1
    (1) do {
        //像素階段切換導言—新
        (simd) (pixel_rate_bary, pixel_mask, pixel_to_coarse_mapping,
next_group_id) =
            coarse-to-pixel-rate-switch(group_id)
        (simd) set_execution_mask(pixel_mask)
        (simd) inter_phase_inputs = swizzle(inter_phase_outputs,
pixel_to_coarse_mapping)

        //規則像素著色器程式碼—與在單一速度解決方案中相同
        (simd) pixel_rate_inputs = interpolate(attribute_deltas,
pixel_rate_bary)
        (simd) pixel_rate_outputs = pixel_shader(pixel_rate_inputs,
inter_phase_inputs)

        //像素階段切換結尾—新
        group_id = next_group_id
    } until (group_id != 0)
}

```

[0018] 根據一些實施例，單一核心針對粗及像素速率著色兩者執行程式。在一些實施例中，可使用將同一通道用於粗四元素及由粗四元素覆蓋之像素四元素兩者的 SIMD 機器。

[0019] 圖 3A 中展示之序列可在軟體、韌體及/或硬體中實施。在軟體及韌體實施例中，其可由儲存在諸如磁性、

光學或半導體儲存器之一或多個非暫時性電腦可讀取媒體中的計算機執行指令實施。

[0020]圖3A之序列，如在區塊10中所指示，用於所有群組或四元素實施粗階段。此後接著為，在區塊12中使用粗對像素映射及執行遮罩針對每一群組的像素階段之實施，使得同一SIMD通道可用以執行粗四元素及由粗四元素覆蓋之像素四元素兩者。

[0021]接著，該序列接收一像素群組識別符(像素group_id)(區塊14)以識別諸如四元素之一特定像素群。基於由在圖3B中描繪之固定功能加速器判定之像素點亮遮罩，針對當前像素群組(區塊16)設定執行遮罩。接著，在區塊18中，使用粗對像素映射在對應的粗與像素階段之間傳遞使用者定義之級間資料。在區塊20中，判定每個像素輸入屬性值，且針對每一群組執行像素速率著色階段。在區塊22中，識別處理的下一個群組。

[0022]根據一些實施例，固定功能加速器可實施於軟體、韌體及/或硬體中。在軟體及韌體實施例中，加速器可由儲存在諸如磁性、光學或半導體儲存器之一或多個非暫時性電腦可讀取媒體中的計算機可讀指令實施。

[0023]序列30(如在圖3B中所展示)可開始於接收新像素群之識別符，如在區塊32中所指示。其判定二維粗對像素映射。基於此，其判定新像素群，管理可變擴展及抑制。對於彼四元素，判定哪些像素經點亮，如在區塊34中所指示。接著，在區塊36中，針對該群組定義粗像素映射，且

判定像素位置及重心形狀。可判定粗像素至像素四元素之 2D 映射，如在區塊 38 中所指示。接著，管理映射之可變擴展，如在區塊 40 中所指示。最後，若所有像素未點亮，則抑制整個四元素，如在區塊 42 中所指示。在接收到新像素群時，流程繼續循環。加速器返回新像素群、其執行遮罩(圖 3A 之區塊 16)及針對此群組之相對像素映射(圖 3A 之區塊 18)。

[0024] 在一個實施例中，軟體(SIMD 程式)與硬體(固定功能加速器)之間的功能性之唯一介面及分裂實現多重速率著色階段至一個單片 SIMD 程式之有效率的映射。在一個實施例中，此方法顯著降低硬體複雜度(邏輯、緩衝)，且不危害效能。

[0025] 為了允許以低於每個像素一次的速率著色，引入粗像素(CP)之概念。CP 為將共用單一粗像素著色器(CPS)評估之結果的 $N_x \times N_y$ 個像素的群組。此類似於多個可見度樣本藉由 MSAA 共用單一像素著色器評估之方式，差異為可藉由變化 CP 之大小來變化著色速率。叫作粗四元素(CQ)的 2×2 CP 之群組經一次著色以有助於使用有限差來計算導數。

[0026] 許多應用可受益於跨螢幕之不同區域變化著色速率的能力。為了實現著色速率之此變化，可將螢幕分成大小 $T_x \times T_y$ 之影像塊，同時允許用於每一影像塊的不同 CP 大小值。每一影像塊按待定 CP 大小映射至 CQ 之著色柵格。

[0027] 可藉由避免重疊著色柵格(其確保每一像素明確地屬於僅一個 CQ)來達成簡化之解耦取樣技術。此可藉由將

CP大小限制於確保著色柵格極佳地與影像塊邊界對準的值之有限集合來達成。藉由此假定，可將著色技術總結如下：

演算法1 藉由CPS的簡化之解耦取樣。

對於每一基元：

對於螢幕上每一被覆蓋之影像塊：

光柵化影像塊且儲存可見片段

判定影像塊之 N_x, N_y

將影像塊分成大小 $2N_x \times 2N_y$ 之CQ

對於具有可見片段之每一CQ：

對CQ著色

將輸出色彩寫入至所有覆蓋之像素

[0028] 為了確保在影像塊內部的CQ之極佳對準柵格，CP大小可受到限制，使得影像塊大小為所有允許之CQ大小的公倍數(例如，最小公倍數(LCM))。

[0029] 易於使用的少數模式可控制CP大小，但彼等模式足夠強大來支援一系列應用。在每一情況下，使用者不直接指定CP大小，而相反地，指定所要的CP大小之一對CP參數(s_x, s_y)。接著將CP參數量化至符合或超過所請求之著色速率的最近可用CP大小。基於選定模式，CP參數(s_x, s_y)可：

- 自每個頂點著色器輸出內插，
- 使用呈現狀態設定至恆定值，或
- 表達為螢幕座標之徑向函數。

[0030] 藉由著色器輸出控制CP參數高度靈活，且允許待表達之許多使用情況。

[0031] 為簡單起見，可啟用恆定CP參數。其為用於將

CPS添加至現有應用之最少侵入性方法。具體地包括使用徑向函數之能力以用於凹式呈現。因為不能使用每個頂點屬性之線性內插穩固地表達徑向函數，所以可將單獨模式用於此特殊情況。

[0032]為了始終符合所需之著色速度，藉由計算 $|s_x|$ 及 $|s_y|$ 在影像塊之保守下限且下捨入至最近的可用CP大小來判定CP大小 $N_x \times N_y$ 。此處絕對值之使用實現將負CP參數用於運動及散焦模糊。

[0033]存在CP大小之兩個量化源。首先，每個影像塊僅評估CP大小一次。其次，將CP大小量化至有限可用CP大小中之一者。此等源中之兩者造成在影像塊間移動的CP大小之不連續性，其可導致可見影像塊轉變。為了補償此等不連續性，擴增紋理取樣器細節層次(LOD)計算以反映所請求之(未量化之)CP大小。此可藉由縮放用以計算紋理LOD的紋理座標之有限差來進行：

$$\delta_x^{\text{new}} = \delta_x \frac{|s_x|}{N_x}, \delta_y^{\text{new}} = \delta_y \frac{|s_y|}{N_y},$$

其中 δ_x 及 δ_y 分別為沿著x軸及y軸的紋理座標之有限差。與每個影像塊評估一次的CP大小不同，針對每一CP評估 δ_x 及 δ_y 之值。補償紋理LOD創造影像細節之平滑變化，其遮蔽了CP大小之不連續性。

[0034]雖然在許多情況下LOD補償可有效，但存在其可能並不適用之一些情境，例如，在按程序產生之結構之情況下。在此等情況下，有可能可基於可用作著色器輸入之CP大小及CP參數來在使用者空間中應用LOD補償技術。

[0035]可在同一呈現遍次中按三個不同速率執行著色。可將著色器計算中之一些移動至比每個像素一次低的速率，同時可每個像素或甚至每個可見度樣本評估某些高頻率效應以減少頻疊。在當前圖形應用程式介面(API)中，此並不可能，因為像素著色器可經組配以按像素或樣本速率執行，但兩者相互排斥。概念上，將圖形管線之單一像素著色器級分成三個截然不同階段，每一速率對應一個階段(見圖2)。因此，在將一影像塊分成粗四元素後，按一或多個不同速率著色每一四元素：每個CP、每個像素及每個樣本。

[0036]圖4例示可將CPS整合至現有圖形管線內之程度。對於螢幕上之每一影像塊，光柵處理器41測試輸入基元以產生四元素片段，四元素片段經受早期深度測試(區塊43)且接著在影像塊緩衝器44中緩衝。在區塊46中，基於粗像素大小(區塊50)，將經緩衝之片段映射至粗像素及粗四元素，接著將粗像素及粗四元素著色(在區塊48中)且映射至輸出緩衝器。藉由簡化之解耦取樣技術，可在影像塊緩衝器44中緩衝針對一給定影像塊及基元之經光柵化樣本，且接著在完整的CQ上調用像素著色器48。影像塊緩衝器保留螢幕座標及所有必要資訊以解決可見度，諸如，深度值及覆蓋資訊。當光柵化新影像塊或新基元時，清空影像塊緩衝器，亦即，僅需要緩衝單一影像塊之資料價值。

[0037]圖5例示將CPS及多重速率著色整合至現有圖形架構(如Direct3D 11)之管線抽象化內以產生呈現目標66。每

一著色階段能夠存取內插之每個頂點屬性IA，且可經由使用者定義之著色器輸出將少量資料自一個階段傳達至下一個階段；允許之資料量為特定實施之極限。階段(CP 60、像素62、樣本64)中之任一者亦可寫入將在稍後管線級(例如，圖框緩衝器摻合)中消耗之輸出。

[0038]在多重速率著色中，計算每個粗像素之低頻擴散項，且計算每個像素之高頻鏡面項。經由頂點著色器控制CP大小。在一個實施例中，單一概念管線級可執行自高達三個不同入口點編譯之單一著色器，每一速率對應一個入口點，如下所展示：

```
struct VS_OUT { //輸出屬性
    float2 cpcize: SV_CoarsePixelSize;
};
VS_OUT VertexShader (VS_IN In){
    VS_OUT Out;
    Out.cpsize = ComputeCPSIZE (In)
    ...
    return Out;
}
[shadingphase ("coarse-pixel")]
[nextshaderfunc ("PixelShader")]
float 4 CoarseShader (VS_OUT In)    {
    return ComputeDiffuse (In)
}
[shadingphase ("pixel")]
float 4 PixelShader (VS_OUT VSIn, float 4 CPIn)    {
    return ComputeSpecular (VSIn, CPIn);
}
```

[0039]此方法給程式設計師提供對在每一速率下操作何程式碼以及何資料在階段之間流動的高可見度，及對在每一速率下操作何程式碼以及何資料在階段之間流動的控制。在可針對易用性犧牲控制之情況下，以上所描述之較簡單模型可仍經實施為分層抽象化。

[0040] 當正藉由著色器程式碼控制CP大小時，可使額外系統解譯值SV_CoarsePixelSize用於在光柵處理器前之最後一個著色級。此輸出為二組分向量，且對應於CP參數 (δ_x, δ_y) 。通常將在頂點著色器中設定此輸出，但亦在外殼、域或幾何著色器(若使用此等者)中定義此輸出。

[0041] 此外，按CP速率執行之著色器程式碼能夠存取針對在量化前的CP參數之經內插值及實際粗像素大小兩者的輸入：分別為SV_RequestedCoarsePixelSize及SV_CoarsePixelSize。此等兩個值一起足夠用於著色器計算其自身LOD補償，用於在不涉及紋理取樣器之計算(例如，預先濾波程序紋理)中使用。

[0042] 存在CPS可用以顯著減少像素著色工作量之若干重要使用情況。此絕非詳盡的清單，因為存在更多得多的使用情況。

[0043] 按高DPI顯示器之天然解析度呈現常為對於圖形處理單元(GPU)過於苛求之任務。典型的補救為降低呈現之解析度且對影像增加取樣，其導致沿著目標廓影之可察覺品質降級，同時表面內部之改變並非如此顯而易見。取而代之，藉由使用CPS且將CP參數設定至恆定值(諸如， 2×2 個像素)，吾人可達成著色速率之急劇減小，同時保留多數細節。

[0044] 藉由對著色速率之更靈活控制，可實現更寬應用範圍，如下文所論述。

[0045] 一些材料具有大量表面細節，而其他不具有。藉

由取決於材料類型來選擇CP參數，可節省計算，在該情況下，視覺影響最小。舉例而言，用於呈現煙霧之粒子系統可相當均質，且經按低速率著色，而具有文字之標識可保證高解析度著色。類似地，在完全陰影中之物件有可能按比在明亮日光下之物件低的速率來著色。

[0046] CPS使按凹式呈現有效率地著色更容易，從而避免在多個呈現遍次上重新發送幾何大小。可組配徑向函數藉由少數參數來控制著色速率：對應於注視中心之點 c 、縱橫比 a 、內較小半徑 r^i 與外較小半徑 r^o ，及內CP參數 s^{\min} 與外CP參數 s^{\max} 。對於凹式呈現，可將 r^i 設定至表示約 5° 之視角的大小，且對於圓形下降函數， $a = 1$ 。

[0047]雖然在一些實施例中，著色系統支援用於高解析度區域之任意位置，但將 c 固定在螢幕中心處且使用較寬縱橫亦產生良好結果，最值得注意地，當自第一人員視角呈現時。與僅在注視追蹤之情況下可能的恰當凹式呈現相對比，此技術可被叫作周邊CPS呈現。

[0048]螢幕之具有運動或散焦模糊的區域通常具有窄頻率回應，且可按較低速率著色。藉由CPS，吾人可藉由在頂點著色器中設定與螢幕空間速度或在彼頂點處之模糊圈成比例之CP參數來控制此區域中之著色速率。由於在剪輯前評估頂點著色器，因此在相機後或在零深度處可存在頂點。對於此等情況，不減小著色速率以確保穩固性。

[0049]可針對 x 軸及 y 軸分開來判定CP參數，以便針對運動模糊產生各向異性著色速率。藉由指派用於在焦平面

前之頂點的負 CP 參數及用於在焦平面後之頂點的正 CP 參數，CP 參數將在焦平面處內插至零。類似地，針對速度指派帶正負號之 CP 參數確保在移動基元內部之靜止點處的零值。由於螢幕空間模糊圈半徑及速度在螢幕空間中為線性函數，因此在此等情況下，應針對 CP 參數停用視角正確內插。

[0050] 存在多重速率著色提供用於縮放品質對效能之有價值的工具之若干不同應用。使用立體像素錐形追蹤每一 2×2 像素計算高品質環境閉塞項，而按像素速率評估擴散紋理查找以保留表面細節中之多數。類似地，亦可按較低速率評估複雜的低頻照明計算(諸如，間接照明)。

[0051] 另一實例為局部增大困難區域中之著色速率。此可經進行以按像素或樣本速率計算僅在鏡面特徵周圍及其他較低處之著色。當今即時工作負荷使用兩遍次方法實施一類多重速率著色相當普遍。在此情況下，第一遍次按像素速率執行(MSAA)，但拋棄在模板緩衝器中標記之困難像素。第二遍次接著每個樣本執行著色，但僅針對在模板緩衝器中標記之像素。使用 CPS，可將此演算法轉換至單一遍次，其選擇性地計算像素及樣本速率著色器中之結果，在可能時，按甚至更低速率執行結果。

[0052] CPS 多重速率著色亦可用以基於每個 CP(例如， 4×4 像素)進行剔除。

[0053] 一個 CPS 軟體實施為基於功能 Direct3D 11 模擬器之中心處理器中的管線。對於此實施，可使用 16×16 個像

素之影像塊大小，及CP大小寬度，及1、2或4之高度。為了支援CPS程式設計師抽象化，擴展DirectX高階著色語言，且可引入新API函數。模擬器可配備儀器以量測動態指令計數，以便給出著色之成本的指示。可追蹤對色彩緩衝器之所有記憶體存取，以用於量測色彩頻寬。

[0054]除了像素著色之外，可在管線中引入粗像素(CP)以及取樣速率著色階段。取決於跨多個處理器及執行緒排程此等額外階段之方式，存在用於多重速率著色之若干潛在實施策略。在一排程策略中，在同一執行緒上連續地執行CP、像素及取樣速率著色階段。以此方式限制排程實現簡單的實施，其需要顯著較少的對管線之改變。其消除了跨階段輸送資料之需求，此係因為資料可駐留於同一暫存器中。其亦避免用於調節階段間資料之複雜流量控制機構。

[0055]圖6展示在具有通道70之4寬SIMD處理器上排程的具有 2×2 之CP大小之粗四元素。首先，跨SIMD通道在72處同時執行粗四元素(例如，0123、4567、.....12131415)中之四個CP，其有助於有限差之計算。在粗階段後，處理器在粗四元素內部在74處同時排程 2×2 像素四元素，從而在所有覆蓋之四元素上循環。由於此排程方案需要在跨階段轉變時資料跨SIMD通道之移動，因此吾人假定處理器指令可用於以有效率的方式來有助於此。

[0056]對於大於4(比方說，8或12)之SIMD寬度，可同時排程多個粗四元素。然而，由於每一粗四元素可具有不同數目個覆蓋之像素，因此較高SIMD寬度亦可導致一些

SIMD通道之減少利用，此係因為一些像素階段迴路較早地終止了。為了具有大SIMD寬度之改良效率，吾人可選擇跨SIMD通道或甚至分開的執行緒更均勻地分佈像素階段工作之更佳排程方案。

[0057] 按比每個像素一次低的速率著色對於色彩緩衝器壓縮工作得良好程度以減小記憶體頻寬具有影響。在 2×2 或更大像素塊內的色彩值之較高均勻性程度通常減小了熵，且使壓縮更有效率。存在可受益於CPS的用於色彩壓縮之許多現有方法。為了例示此在實務上之工作，可使用Pool等人之方案，其計算像素之間的色差，且接著使用費布那(Fibonacci)編碼器對該差應用熵寫碼。Pool等人，「在GPU上的可變精確度浮點緩衝器之無損壓縮 (Lossless Compression of Variable Precision Floating-Point Buffers on GPUs)」，關於互動式3D圖形及遊戲之研討會(2012年)，第47-54頁。

[0058] 可修改Pool等人之方案，使得始終使用預定義之希伯特(Hilbert)曲線以階層式方式來訪問像素。結果，將首先在 2×2 四元素內部且接著在 4×4 區域內部等等計算差異。當實現CPS時，在此等區域中之一些內部的若干色彩有可能相同，從而產生零差異，可使用費布那編碼器有效率地編碼該等差異。吾人使用128個位元組之快取線大小，其等效於 8×4 個像素(對於RGBA8呈現目標)。當自色彩快取記憶體收回線時，其經壓縮，且若所得大小小於或等於64個位元組，那壓縮成功，且將線標記為在單獨的控制緩衝器中經

壓縮，且僅將一個 64 位元組異動發送至記憶體。否則，在兩個 64 位元組異動中將未壓縮之資料發送至記憶體。藉由最近最少使用(LRU)替換策略來模型化 64 kB 色彩快取記憶體。

[0059] 粗像素著色(CPS)可顯著降低著色之成本，對影像品質具有極少或無可察覺的影響。CPS 自然地擬合即時圖形管線之演進，此係因為其經由可規劃著色速率引入新的靈活性程度，同時仍解決了對於用於手持型裝置之快速生長市場之能效的真實需求。

[0060] 圖 7 為根據一個實施例的資料處理系統 100 之方塊圖。資料處理系統 100 包括一或多個處理器 102 及一或多個圖形處理器 108，且可為單一處理器桌上型系統、多處理器工作站系統或具有大量處理器 102 或處理器核心 107 之伺服器系統。在一個實施例中，資料處理系統 100 為系統單晶片積體電路(SOC)，用於在行動、手持型或內嵌型裝置中使用。

[0061] 資料處理系統 100 之一實施例可包括基於伺服器之遊戲平台、遊戲控制台(包括遊戲及媒體控制台、行動遊戲控制台、手持型遊戲控制台或線上遊戲控制台)，或併入於前述各者內。在一個實施例中，資料處理系統 100 為行動電話、智慧型手機、平板計算裝置或行動網際網路裝置。資料處理系統 100 亦可包括可佩戴裝置(諸如，智慧型手錶可佩戴裝置、智慧型護目鏡裝置、擴增實境裝置或虛擬實境裝置)、與可佩戴裝置耦接或整合於可佩戴裝置內。在一個實施例中，資料處理系統 100 為具有一或多個處理器 102

及由一或多個圖形處理器108產生之一圖形介面的電視或機上盒裝置。

[0062] 該一或多個處理器102各包括一或多個處理器核心107以處理在經執行時進行用於系統及使用者軟體之操作的指令。在一個實施例中，該一或多個處理器核心107中之每一者經組配以處理特定指令集109。指令集109可有助於複雜指令集計算(CISC)、精簡指令集計算(RISC)或經由超長指令字(VLIW)之計算。多個處理器核心107可各處理可包含有助於其他指令集之仿真的指令之不同指令集109。處理器核心107亦可包括其他處理裝置，諸如，數位信號處理器(DSP)。

[0063] 在一個實施例中，處理器102包括快取記憶體104。取決於架構，處理器102可具有單一內部快取記憶體或多層內部快取記憶體。在一個實施例中，在處理器102之各種組件當中共用快取記憶體。在一個實施例中，處理器102亦使用可使用已知快取記憶體同調性技術在處理器核心107當中共用之一外部快取記憶體(例如，第3層(L3)快取記憶體或最後一層快取記憶體(LLC))(未展示)。暫存器檔案106另外包括於處理器102中，該處理器可包括不同類型之暫存器以用於儲存不同類型之資料(例如，整數暫存器、浮點暫存器、狀態暫存器及指令指標暫存器)。一些暫存器可為通用暫存器，而其他暫存器可具體針對處理器102之設計。

[0064] 處理器102耦接至處理器匯流排110以在處理器102與系統100中之其他組件之間傳輸資料信號。系統100使

用一示範性「集線器」系統架構，其包括一記憶體控制器集線器116及一輸入輸出(I/O)控制器集線器130。記憶體控制器集線器116有助於記憶體設備與系統100之其他組件之間的通訊，而I/O控制器集線器(ICH)130提供經由局部I/O匯流排至I/O裝置之連接。

[0065]記憶體裝置120可為動態隨機存取記憶體(DRAM)裝置、靜態隨機存取記憶體(SRAM)裝置、快閃記憶體裝置或具有充當處理序記憶體之合適效能的某一其他記憶體裝置。記憶體120可儲存資料122及指令121用於在處理器102執行一處理序時使用。記憶體控制器集線器116亦與一可選外部圖形處理器112耦接，該可選外部圖形處理器可與處理器102中之一或多個圖形處理器108通訊以進行圖形及媒體操作。

[0066]ICH 130使周邊設備能夠經由高速I/O匯流排連接至記憶體120及處理器102。I/O周邊設備包括一音訊控制器146、一韌體介面128、一無線收發器126(例如，Wi-Fi、藍芽)、一資料儲存裝置124(例如，硬碟機、快閃記憶體等)及用於將舊版(例如，個人系統2 (PS/2))裝置耦接至系統之舊版I/O控制器。一或多個通用串列匯流排(USB)控制器142連接輸入裝置，諸如，鍵盤與滑鼠144組合。網路控制器134亦可耦接至ICH 130。在一個實施例中，高效能網路控制器(未展示)耦接至處理器匯流排110。

[0067]圖8為具有一或多個處理器核心202A-N、一整合式記憶體控制器214及一整合式圖形處理器208的處理器

200之一實施例之方塊圖。處理器200可包括高達且包括由虛線框表示之額外核心202N之額外核心。核心202A-N中之每一者包括一或多個內部快取記憶體單元204A-N。在一個實施例中，每一核心亦能夠存取一或多個共用快取單元206。

[0068]內部快取記憶體單元204A-N及共用快取記憶體單元206表示在處理器200內之快取記憶體階層。快取記憶體階層可包括在每一核心內之至少一層指令及資料快取記憶體，及一或多個層共用層間快取記憶體，諸如，第2層(L2)、第3層(L3)、第4層(L4)或其他層之快取記憶體，其中在外部記憶體前之最高層快取記憶體經分類為最後一層快取記憶體(LLC)。在一個實施例中，快取記憶體同調性邏輯維持各種快取記憶體單元206與204A-N之間的同調性。

[0069]處理器200亦可包括一或多個匯流排控制器單元216及一系統代理210之一集合。該一或多個匯流排控制器單元管理一組周邊匯流排，諸如，一或多個周邊組件互連匯流排(例如，PCI、高速PCI)。系統代理210為各種處理器組件提供管理功能性。在一個實施例中，系統代理210包括一或多個整合之記憶體控制器214以管理對各種外部記憶體裝置(未展示)之存取。

[0070]在一個實施例中，核心202A-N中之一或多者包括對同時多緒執行之支援。在此實施例中，系統代理210包括用於在多緒執行之處理期間協調及操作核心202A-N之組件。系統代理210可另外包括一功率控制單元(PCU)，其包

括調節核心202A-N及圖形處理器208之功率狀態的邏輯及組件。

[0071] 處理器200另外包括一圖形處理器208以執行圖形處理操作。在一個實施例中，圖形處理器208與共用快取記憶體單元206及系統代理單元210(包括一或多個整合之記憶體控制器214)之集合耦接。在一個實施例中，顯示控制器211與圖形處理器208耦接以驅動至一或多個耦接之顯示器的圖形處理器輸出。顯示控制器211可為經由至少一個互連件與圖形處理器耦接的單獨模組，或可整合於圖形處理器208或系統代理210內。

[0072] 在一個實施例中，基於環形之互連單元212用以耦接處理器200之內部組件，然而，可使用一替代性互連單元，諸如，點對點互連件、切換式互連件或其他技術，包括此項技術中熟知之技術。在一個實施例中，圖形處理器208經由I/O鏈路213與環形互連件212耦接。

[0073] 示範性I/O鏈路213表示多種I/O互連件中之至少一者，包括有助於各種處理器組件與高效能內嵌型記憶體模組218(諸如，eDRAM模組)之間的通訊之封裝上I/O互連件。在一個實施例中，核心202-N中之每一者及圖形處理器208使用內嵌型記憶體模組218作為共用最後一層快取記憶體。

[0074] 在一個實施例中，核心202A-N為執行同一指令集架構之均質核心。在另一實施例中，就指令集架構(ISA)而言，核心202A-N係異質的，其中核心202A-N中之一或多

者執行第一指令集，而其他核心中之至少一者執行第一指令集或不同指令集之一子集。

[0075] 處理器 200 可為一或多個基體之一部分或使用許多製程技術(例如，互補金屬氧化物半導體(CMOS)、雙極接面/互補金屬氧化物半導體(BiCMOS)或 N 型金屬氧化物半導體邏輯(NMOS))中之任一者實施於一或多個基體上。另外，處理器 200 可實施於一或多個晶片上，或實施為系統單晶片(SOC)積體電路，除了其他組件之外，該積體電路亦具有所例示之組件。

[0076] 圖 9 為可為精密圖形處理單元或可為與多個處理核心整合之圖形處理器的圖形處理器 300 之一個實施例之方塊圖。在一個實施例中，經由至圖形處理器上之暫存器之記憶體映射 I/O 介面及經由置放至處理器記憶體內之命令來與圖形處理器通訊。圖形處理器 300 包括一記憶體介面 314 以存取記憶體。記憶體介面 314 可為至局部記憶體、一或多個內部快取記憶體、一或多個共用外部快取記憶體及/或至系統記憶體之介面。

[0077] 圖形處理器 300 亦包括一顯示控制器 302 以驅動至顯示裝置 320 之顯示輸出資料。顯示控制器 302 包括用於一或多個覆疊平面之硬體，該一或多個覆疊平面用於顯示器及多個視訊層與使用者介面元件之組成。在一個實施例中，圖形處理器 300 包括一視訊編碼解碼器引擎 306 以將媒體編碼至一或多個媒體編碼格式、自一或多個媒體編碼格式解碼媒體或在一或多個媒體編碼格式之間轉碼媒體，該

一或多個媒體編碼格式包括(但不限於)諸如MPEG-2之動畫專業團體(MPEG)格式、諸如H.264/MPEG-4 AVC之進階視訊寫碼(AVC)格式以及美國電影與電視工程師學會(SMPTE) 421M/VC-1，及諸如JPEG及運動JPEG(MJPEG)格式之聯合照相專業團體(JPEG)格式。

[0078]在一個實施例中，圖形處理器300包括一區塊影像轉移(BLIT)引擎304以進行包含(例如)位元邊界區塊轉移之二維(2D)光柵處理器操作。然而，在一個實施例中，使用圖形處理引擎(GPE)310之一或多個組件進行2D圖形操作。圖形處理引擎310為用於進行圖形操作(包括三維(3D)圖形操作及媒體操作)之計算引擎。

[0079]GPE 310包括一3D管線312以用於進行3D操作，諸如，使用作用於3D基元形狀(例如，矩形、三角形等)之處理函數呈現三維影像及場景。3D管線312包括進行元件內之各種任務及/或將執行緒產生至3D/媒體子系統315的可規劃及固定功能元件。而3D管線312可用以進行媒體操作，GPE 310之一實施例亦包括一媒體管線316，其特定地用以進行媒體操作(諸如，視訊後處理及影像增強)。

[0080]在一個實施例中，媒體管線316包括代替或代表視訊編碼解碼器引擎306進行一或多個專業化媒體操作(諸如，視訊解碼加速、視訊解交錯及視訊編碼加速)之固定功能或可規劃邏輯單元。在一個實施例中，媒體管線316另外包括一執行緒產生單元以產生用於在3D/媒體子系統315上執行之執行緒。產生之執行緒進行用於在3D/媒體子系統中

包括之一或多個圖形執行單元上的媒體操作之計算。

[0081] 3D/媒體子系統315包括用於執行由3D管線312及媒體管線316產生之執行緒的邏輯。在一個實施例中，管線將執行緒執行請求發送至3D/媒體子系統315，該3D/媒體子系統包括用於仲裁各種請求及調度各種請求至可用執行緒執行資源之執行緒調度邏輯。執行資源包括一陣列圖形執行單元以處理3D及媒體執行緒。在一個實施例中，3D/媒體子系統315包括用於執行緒指令及資料之一或多個內部快取記憶體。在一個實施例中，子系統亦包括共用記憶體(包括暫存器及可定址記憶體)以在執行緒之間共用資料且儲存輸出資料。

[0082] 圖10為用於圖形處理器的圖形處理引擎410之一實施例之方塊圖。在一個實施例中，圖形處理引擎(GPE)410為圖9中展示的GPE 310之一版本。GPE 410包括一3D管線412及一媒體管線416，其中之每一者可不同於或類似於圖9之3D管線312及媒體管線316之實施。

[0083] 在一個實施例中，GPE 410與將命令流提供至GPE 3D及媒體管線412、416之命令串流器403耦接。命令串流器403耦接至記憶體，該記憶體可為系統記憶體或內部快取記憶體及共用快取記憶體中之一或多者。命令串流器403自記憶體接收命令且將命令發送至3D管線412及/或媒體管線416。3D及媒體管線藉由經由各別管線內之邏輯進行操作或藉由將一或多個執行緒調度至執行單元陣列414來處理命令。在一個實施例中，執行單元陣列414可縮放，使

得基於 GPE 410 之目標功率及效能等級，該陣列包括可變數目個執行單元。

[0084] 取樣引擎 430 與記憶體(例如，快取記憶體或系統記憶體)及執行單元陣列 414 耦接。在一個實施例中，取樣引擎 430 提供用於可縮放執行單元陣列 414 之一記憶體存取機構，其允許執行陣列 414 自記憶體讀取圖形及媒體資料。在一個實施例中，取樣引擎 430 包括進行用於媒體之專業化影像取樣操作的邏輯。

[0085] 取樣引擎 430 中之專業化媒體取樣邏輯包括一去雜訊/解交錯模組 432、一運動估計模組 434 及一影像縮放及濾波模組 436。去雜訊/解交錯模組 432 包括對經解碼視訊資料進行去雜訊或解交錯演算法中之一或多者的邏輯。解交錯邏輯將經交錯之視訊內容的交替欄位組合至單一視訊圖框內。去雜訊邏輯減少或移除來自視訊及影像資料之資料雜訊。在一個實施例中，去雜訊邏輯及解交錯邏輯為運動自適應性，且基於在視訊資料中偵測到之運動量使用空間或時間濾波。在一個實施例中，去雜訊/解交錯模組 432 包括專用運動偵測邏輯(例如，在運動估計引擎 434 內)。

[0086] 運動估計引擎 434 藉由對視訊資料進行諸如運動向量估計及預測之視訊加速功能來提供用於視訊操作之硬體加速。運動估計引擎判定描述連續視訊圖框之間的影像資料之變換之運動向量。在一個實施例中，圖形處理器媒體編碼解碼器使用視訊運動估計引擎 434 對在巨集區塊層面之視訊進行操作，該等操作可另外在計算上集中以使用

通用處理器進行。在一個實施例中，運動估計引擎434通常可為圖形處理器組件所用，以輔助對視訊資料內的運動之方向或量值敏感或自適應之視訊解碼及處理功能。

[0087] 影像縮放及濾波模組436進行影像處理操作以增強產生之影像及視訊的視覺品質。在一個實施例中，縮放及濾波模組436在將資料提供至執行單元陣列414前在取樣操作期間處理影像及視訊資料。

[0088] 在一個實施例中，圖形處理引擎410包括一資料埠444，其提供用於圖形子系統之額外機構以存取記憶體。資料埠444有助於用於包括呈現目標寫入、恆定緩衝器讀取、暫時記憶體空間讀取/寫入及媒體表面存取之操作的記憶體存取。在一個實施例中，資料埠444包括快取記憶體空間以快取對記憶體之存取。快取記憶體可為單一資料快取記憶體或分成用於經由資料埠存取記憶體之多個子系統的多個快取記憶體(例如，呈現緩衝器快取記憶體、恆定緩衝器快取記憶體等)。在一個實施例中，在執行單元陣列414中之執行單元上執行的執行緒藉由經由耦接圖形處理引擎410之子系統中之每一者的資料分佈互連件交換訊息來與資料埠通訊。

[0089] 圖11為圖形處理器之另一實施例之方塊圖。在一個實施例中，圖形處理器包括一環形互連件502、一管線前端504、一媒體引擎537及圖形核心580A-N。環形互連件502將圖形處理器耦接至其他處理單元，包括其他圖形處理器或一或多個通用處理器核心。在一個實施例中，圖形處理

器為整合於多核心處理系統內的許多處理器中之一者。

[0090] 圖形處理器經由環形互連件 502 接收成批之命令。傳入命令由管線前端 504 中之命令串流器 503 解譯。圖形處理器包括可縮放執行邏輯以經由圖形核心 580A-N 進行 3D 幾何處理及媒體處理。對於 3D 幾何處理命令，命令串流器 503 將命令供應至幾何管線 536。對於至少一些媒體處理命令，命令串流器 503 將命令供應至與媒體引擎 537 耦接之視訊前端 534。媒體引擎 537 包括用於視訊及影像後處理之一視訊品質引擎 (VQE) 530，及一多格式編碼/解碼 (MFX) 533 引擎以提供硬體加速之媒體資料編碼及解碼。幾何管線 536 及媒體引擎 537 各產生用於由至少一個圖形核心 580A 提供之執行緒執行資源的執行緒。

[0091] 圖形處理器包括特徵為模組化核心 580A-N (有時被稱作核心切片) 之可縮放執行緒執行資源，每一核心具有多個子核心 550A-N、560A-N (有時被稱作核心子切片)。圖形處理器可具有任何數目個圖形核心 580A 至 580N。在一個實施例中，圖形處理器包括一圖形核心 580A，其具有至少一第一子核心 550A 及一第二子核心 560A。在另一實施例中，圖形處理器為具有單一子核心 (例如，550A) 之低功率處理器。在一個實施例中，圖形處理器包括多個圖形核心 580A-N，每一者包括第一子核心 550A-N 之一集合及第二子核心 560A-N 之一集合。第一子核心 550A-N 之集合中的每一子核心包括執行單元 552A-N 及媒體/紋理取樣器 554A-N 之至少一第一集合。第二子核心 560A-N 之集合中的

每一子核心包括執行單元562A-N及取樣器564A-N之至少一第二集合。在一個實施例中，每一子核心550A-N、560A-N共用一組共用資源570A-N。在一個實施例中，共用資源包括共用快取記憶體及像素操作邏輯。其他共用資源亦可包括於圖形處理器之各種實施例中。

[0092] 圖12例示包括在圖形處理引擎之一個實施例中使用的一陣列處理元件之執行緒執行邏輯600。在一個實施例中，執行緒執行邏輯600包括一執行緒調度器604、指令快取記憶體606、一可縮放執行單元陣列(包括多個執行單元608A-N)、一取樣器610、一資料快取記憶體612及一資料埠614。在一個實施例中，包括之組件係經由連接至組件中之每一者的互連網狀架構來互連。執行緒執行邏輯600包括經由指令快取記憶體606、資料埠614、取樣器610及執行單元陣列608A-N中之一或多者至諸如系統記憶體或快取記憶體之記憶體的一或多個連接。在一個實施例中，每一執行單元(例如，608A)為能夠執行多個同時執行緒且針對每一執行緒並行地處理多個資料元素的個別向量處理器。執行單元陣列608A-N包括任何數目個別執行單元。

[0093] 在一個實施例中，執行單元陣列608A-N主要地用以執行「著色器」程式。在一個實施例中，陣列608A-N中之執行單元執行包括對於許多標準3D圖形著色器之原生支援的指令集，使得來自圖形程式庫(例如，Direct 3D及OpenGL)之著色器程式按最小平移來執行。執行單元支援頂點及幾何處理(例如，頂點程式、幾何程式、頂點著色器)、

像素處理(例如，像素著色器、片段著色器)及通用處理(例如，計算及媒體著色器)。

[0094] 執行單元陣列608A-N中之每一執行單元對資料元素陣列操作。資料元素之數目為「執行大小」，或用於指令的頻道之數目。執行頻道為針對指令內之資料元素存取、遮蔽及流量控制的執行之邏輯單元。頻道之數目可獨立於用於一特定圖形處理器的實體ALU或FPU之數目。執行單元608A-N支援整數及浮點資料類型。

[0095] 執行單元指令集包括單指令多資料(SIMD)指令。可將各種資料元素作為封裝資料類型儲存於暫存器中，且執行單元將基於元素之資料大小處理各種元素。舉例而言，當對256位元寬向量操作時，將向量之256個位元儲存於暫存器中，且該執行單元對該向量將其作為四個分開的64位元封裝資料元素(四元素-字(QW)大小資料元素)、八個分開的32位元封裝資料元素(雙字(DW)大小資料元素)、十六個分開的16位元封裝資料元素(字(W)大小資料元素)或三十二個分開的8位元資料元素(位元組(B)大小資料元素)操作。然而，不同向量寬度及暫存器大小係可能的。

[0096] 一或多個內部指令快取記憶體(例如，606)包括於執行緒執行邏輯600中以快取用於執行單元之執行緒指令。在一個實施例中，包括一或多個資料快取記憶體(例如，612)以在執行緒執行期間快取執行緒資料。包括一取樣器610以提供用於3D操作之紋理取樣及用於媒體操作之媒體取樣。在一個實施例中，取樣器610包括專業化紋理或

媒體取樣功能性以在將取樣之資料提供至執行單元前在取樣過程期間處理紋理或媒體資料。

[0097] 在執行期間，圖形及媒體管線經由執行緒產生及調度邏輯將執行緒起始請求發送至執行緒執行邏輯600。執行緒執行邏輯600包括一局部執行緒調度器604，其仲裁來自圖形及媒體管線之執行緒起始請求，且在一或多個執行單元608A-N上具現化所請求之執行緒。舉例而言，幾何管線(例如，圖10之536)調度頂點處理、棋盤形格局或幾何處理執行緒至執行緒執行邏輯600。執行緒調度器604亦可處理來自執行著色器程式之執行環境執行緒產生請求。

[0098] 一旦一群幾何物件已經處理且光柵化至像素資料，則調用像素著色器602以進一步計算輸出資訊且使結果被寫入至輸出表面(例如，色彩緩衝器、深度緩衝器、模板緩衝器等)。在一個實施例中，像素著色器602計算待跨經光柵化目標內插的各種頂點屬性之值。像素著色器602接著執行供應API之像素著色器程式。為了執行像素著色器程式，像素著色器602經由執行緒調度器604調度執行緒至執行單元(例如，608A)。像素著色器602使用取樣器610中之紋理取樣邏輯存取儲存在記憶體中之紋理映射中的紋理資料。對紋理資料及輸入幾何資料之算術運算計算每一幾何片段之像素彩色資料，或拋棄一或多個像素作進一步處理。

[0099] 在一個實施例中，資料埠614提供一記憶體存取機構，以用於執行緒執行邏輯600將經處理之資料輸出至記憶體以用於在圖形處理器輸出管線上處理。在一個實施例

中，資料埠614包括或耦接至一或多個快取記憶體(例如，資料快取記憶體612)以快取用於經由資料埠的記憶體存取之資料。

[0100]圖13為例示根據一個實施例的圖形處理器執行單元指令格式之方塊圖。可將對較早描述之固定功能加速器的存取映射至此處描述之指令(發送訊息)。在一個實施例中，圖形處理器執行單元支援具有呈多個格式之指令的指令集。實線框例示通常包括於執行單元指令中之組分，而虛線包括可選或僅包括於該等指令之子集中的組分。描述及例示之指令格式係巨集指令，因為其為供應至執行單元之指令，如與一旦指令經處理而自指令解碼產生之微操作相反。

[0101]在一個實施例中，圖形處理器執行單元原生地支援呈128位元格式710之指令。64位元壓緊指令格式730可用於基於選定指令、指令選項及運算元之數目的一些指令。原生128位元格式710提供對所有指令選項之存取，而一些選項及操作在64位元格式730中受到限制。在64位元格式730中可用之原生指令按實施例變化。在一個實施例中，使用索引欄位713中之一組索引值來部分地壓緊指令。執行單元硬體基於索引值參考一組壓緊表，且使用壓緊表輸出重建構呈128位元格式710之一原生指令。

[0102]對於每一格式，指令作業碼712定義執行單元將進行之操作。執行單元跨每一運算元之多個資料元素並行地執行每一指令。舉例而言，回應於添加指令，執行單元

跨表示紋理元素或像元之每一色彩頻道進行同時添加操作。按預設，執行單元跨運算元之所有資料頻道進行每一指令。指令控制欄位712實現對諸如頻道選擇(例如，預測)及資料頻道階數(例如，攪和)之某些執行選項之控制。對於128位元指令710，exec-size欄位716限制將並行地執行的資料頻道之數目。exec-size欄位716不可用於按64位元緊湊指令格式730使用。

[0103] 一些執行單元指令具有高達三個運算元，包括兩個源運算元src0 720、src1 722及一個目的地718。在一個實施例中，執行單元支援雙目的地指令，其中該等目的地中之一者經暗示。資料操縱指令可具有第三源運算元(例如，SRC2 724)，其中指令作業碼JJ12判定源運算元之數目。指令之最後一個源運算元可為與指令一起傳遞之立即(例如，經硬寫碼)值。

[0104] 在一個實施例中，基於作業碼位元欄位將指令分群以簡化作業碼解碼740。對於8位元作業碼，位元4、5及6允許執行單元判定作業碼之類型。展示之精密作業碼分群為示範性的。在一個實施例中，移動及邏輯作業碼群組742包括資料移動及邏輯指令(例如，mov、cmp)。移動及邏輯群組742共用五個最高有效位元(MSB)，其中移動指令呈0000xxxxb(例如，0x0x)之形式，且邏輯指令呈0001xxxxb(例如，0x01)之形式。流量控制指令群組744(例如，呼叫、jmp)包括呈0010xxxxb(例如，0x20)之形式的指令。雜項指令群組746包括指令之混合，包括呈0011xxxxb(例如，0x30)之形

式的同步指令(例如，等待、發送)。平行數學指令群組748包括呈0100xxxxb(例如，0x40)之形式的逐個組分算術指令(例如，加、mul)。平行數學群組748跨資料頻道並行地進行算術運算。向量數學群組750包括呈0101xxxxb(例如，0x50)之形式的算術指令(例如，dp4)。向量數學群組對向量運算元進行諸如點乘計算之算術。

[0105]圖14例示根據一實施例的用於資料處理系統之示範性圖形軟體架構。軟體架構包括一3D圖形應用程式1010、一作業系統1020及至少一個處理器1030。處理器1030包括一圖形處理器1032及一或多個通用處理器核心1034。圖形應用程式1010及作業系統1020各在資料處理系統之系統記憶體1050中執行。

[0106]在一個實施例中，3D圖形應用程式1010含有包括著色器指令1012之一或多個著色器程式。著色器語言指令可呈高階著色器語言，諸如，高階著色器語言(HLSL)或OpenGL著色器語言(GLSL)。該應用程式亦包括呈適合於由通用處理器核心1034執行之機器語言的可執行指令1014。該應用程式亦包括由頂點資料定義之圖形物件1016。

[0107]作業系統1020可為來自微軟公司(Microsoft Corporation)之Microsoft®Windows®作業系統、類似專屬UNIX之作業系統或使用Linux核心之變體的類似開放源UNIX之作業系統。當Direct3D API在使用中時，作業系統1020使用前端著色器編譯器1024將HLSL下之任何著色器指令1012編譯成較低階著色器語言。編譯可為剛好及時

(just-in-time)編譯，或該應用程式可進行共用預先編譯。在一個實施例中，在3D圖形應用程式1010之編譯期間將高階著色器編譯成低階著色器。

[0108]使用者模式圖形驅動器1026可含有一後端著色器編譯器1027以將著色器指令1012轉換成硬體特定表示。著色器編譯器將粗及像素速率階段融合至一個單片程式內，且在周圍添加「膠結程式碼(glue code)」，如結合較早單片程式描述所描述。當OpenGL API在使用中時，將GLSL高階語言下之著色器指令1012傳遞至使用者模式圖形驅動器1026供編譯。使用者模式圖形驅動器使用作業系統核心模式功能1028與核心模式圖形驅動器1029通訊。核心模式圖形驅動器1029與圖形處理器1032通訊以調度命令及指令。

[0109]就本文中描述各種操作或功能而言，其可經描述或定義為硬體電路、軟體程式碼、指令、組配及/或資料。內容可體現於硬體邏輯中，或直接體現為可執行軟體(「目標」或「可執行」形成)、原始程式碼、經設計用於在圖形引擎上執行之高階著色器程式碼或用於特定處理器或圖形核心之指令集中的低階組合語言程式碼。本文所述的實施例的軟件內容可經由其上存儲有內容之製品提供，或經由操作通訊介面以經由通訊介面發送資料之方法提供。

[0110]非暫時性機器可讀儲存媒體可使機器進行所描述之功能或操作，且包括以可由機器(例如，計算裝置、電子系統等)存取之形式儲存資訊的任何機構，諸如，可記錄

/不可記錄媒體(例如，唯讀記憶體(ROM)、隨機存取記憶體(RAM)、磁碟儲存媒體、光學儲存媒體、快閃記憶體裝置等)。通訊介面包括與固線式、無線、光學等媒體中之任何者介面連接以與另一裝置通訊的任何機構，諸如，記憶體匯流排介面、處理器匯流排介面、網際網路連接、磁碟控制器等。藉由提供組配參數或發送信號以準備通訊介面提供描述軟體內容之資料信號來組配通訊介面。可經由發送至通訊介面之一或多個命令或信號來存取通訊介面。

[0111] 以下條款及/或實例係關於另外實施例：

一個實例實施例可為一種方法，其包括：定義針對一像素群之一粗對像素映射，進行粗像素著色，及使用該粗對像素映射在用於該群組之一硬體處理器中進行像素著色。該方法亦可包括判定該群組中之哪些像素經點亮。該方法亦可包括將同一核心用於粗及像素著色。該方法亦可包括使用一單指令多資料著色器。該方法亦可包括將同樣的通道用於粗及像素著色兩者。該方法亦可包括在針對該群組進行粗著色後針對該群組進行像素著色。該方法亦可包括將一指令用於在使一單指令多資料機器運作之同一核心程式中將程式執行自粗速率切換至像素速率。該方法亦可包括一次粗著色多個四元素，且接著連續地像素著色該等四元素中之每一者。

[0112] 另一實例實施例可為一種方法，其包括：藉由階段將粗及像素著色映射至一個單片單指令多資料程式內，判定重心空間中之像素位置，及若一像素群中之所有像素

未點亮，則抑制該群組。該方法可包括在一固定功能加速器中編譯粗對像素映射、像素點亮遮罩、像素群及重心。該方法可包括將像素速率階段實施為由該固定功能加速器判定之像素群組上之一迴路。該方法可包括在每一迴路中設定一執行遮罩，使用粗對像素映射攪和粗對像素速率階段之間的級間資料，及在每一迴路遍次中使用重心內插每個像素屬性。

[0113] 在另一實例實施例中，一或多個非暫時性電腦可讀取媒體儲存由一處理器所執行以進行包括以下操作之一序列的指令：定義針對一像素群之一粗對像素映射，進行粗像素著色，及使用該粗對像素映射在用於該群組之一硬體處理器中進行像素著色。該媒體可包括該序列包括判定該群組中之哪些像素經點亮。該媒體可包括該序列包括將同一核心用於粗及像素著色。該媒體可包括該序列包括使用一單指令多資料著色器。該媒體可包括該序列包括將同樣的通道用於粗及像素著色兩者。該媒體可包括該序列包括在針對該群組進行粗著色後針對該群組進行像素著色。該媒體可包括該序列包括將一指令用於在使一單指令多資料機器運作之同一核心程式中將程式執行自粗速率切換至像素速率。該媒體可包括該序列包括一次粗著色多個四元素，且接著連續地像素著色該等四元素中之每一者。

[0114] 另一實例實施例可為一種設備，其包括：一硬體處理器，其定義針對一像素群之一粗對像素映射，進行粗像素著色，及使用該粗對像素映射在用於該群組之一硬體

處理器中進行像素著色；及一儲存器，其耦接至該處理器。該設備可包括該處理器判定該群組中之哪些像素經點亮。該設備可包括該處理器將同一核心用於粗及像素著色。該設備可包括該處理器使用一單指令多資料著色器。該設備可包括該處理器將同樣的通道用於粗及像素著色兩者。該設備可包括該處理器在針對該群組進行粗著色後針對該群組進行像素著色。該設備可包括該處理器將一指令用於在使一單指令多資料機器運作之同一核心程式中將程式執行自粗速率切換至像素速率。該設備可包括該處理器一次粗著色多個四元素，且接著連續地像素著色該等四元素中之每一者。該設備可包括一記憶體控制器集線器。該設備可包括一無線收發器。

[0115]所描述之各種組件可為用於進行所描述之操作或功能之構件。本文中描述之每一組件包括軟體、硬體或此等之組合。該等組件可實施為軟體模組、硬體模組、專用硬體(例如，特殊應用硬體、特殊應用積體電路(ASIC)、數位信號處理器(DSP)等)、內嵌型控制器、固線式電路等。除了本文中所描述之內容外，可對本發明的揭示之實施例及實施進行各種修改，而不脫離其範疇。因此，本文中之例示及實例應以例示性而非限制性意義來解釋。本發明之範疇應僅藉由參考接下來之申請專利範圍來量測。

[0116]本文中所描述之圖形處理技術可實施於各種硬體架構中。舉例而言，圖形功能性可整合於晶片組內。替代地，可使用一離散圖形處理器。如又一實施例，可由通

用處理器(包括多核心處理器)實施圖形功能。

[0117]貫穿本說明書對“一個實施例”或“一實施例”之提及意謂結合該實施例所描述之一特定特徵、結構或特性包括在涵蓋於本發明內之至少一個實施中。因此，片語「一個實施例」或「在一實施例中」之出現未必指同一實施例。此外，可以不同於例示之特定實施例的其他合適形式來設立特定特徵、結構或特性，且所有此等形式可涵蓋於本申請案之申請專利範圍內。

[0118]雖然已描述有限數目個實施例，但熟習此項技術者應瞭解自其之眾多修改及變化。意欲隨附申請專利範圍涵蓋如在本發明之真實精神及範疇內的所有此等修改及變化。

【符號說明】

10-22、30-42、43、46、50...區塊

41...光柵處理器

44...影像塊緩衝器

48...像素著色器

60...CP

62...像素

64...樣本

66...呈現目標

70...通道

72...粗像素

74...像素四元素

100...資料處理系統

102、200、1030...處理器
104...快取記憶體
106...暫存器檔案
107...處理器核心
108、300、1032...圖形處理器
109...指令集
110...處理器匯流排
112...外部圖形處理器
116...記憶體控制器集線器
120...記憶體裝置/記憶體
121...指令
122...資料
124...資料儲存裝置
126...無線收發器
128...韌體介面
130...輸入輸出(I/O)控制器集線器/ICH
134...網路控制器
140...舊版I/O控制器
142...通用串列匯流排(USB)控制器
144...鍵盤與滑鼠
146...音訊控制器
202A-202N...處理器核心
204A-204N...內部快取記憶體單元
206...共用快取記憶體單元
208...整合式圖形處理器

- 210...系統代理/系統代理單元
- 211、302...顯示控制器
- 212...基於環形之互連單元/環形互連件
- 213...I/O鏈路
- 214...整合式記憶體控制器
- 216...匯流排控制器單元
- 218...內嵌型記憶體模組
- 304...區塊影像轉移(BLIT)引擎
- 306...視訊編碼解碼器引擎
- 310、410...圖形處理引擎(GPE)
- 312、412...3D管線
- 314...記憶體介面
- 315...3D/媒體子系統
- 316、416...媒體管線
- 320...顯示裝置
- 403、503...命令串流器
- 414...執行單元陣列
- 430...取樣引擎
- 432...去雜訊/解交錯模組
- 434...運動估計模組
- 436...影像縮放及濾波模組
- 444...資料埠
- 502...環形互連件
- 504...管線前端
- 530...視訊品質引擎(VQE)

533...多格式編碼/解碼(MFX)
534...視訊前端
536...幾何管線
537...媒體引擎
550A-550N、560A-560N...子核心
552A-552N、562A-562N...執行單元
554A-554N...媒體/紋理取樣器
564A-564N...取樣器
570A-570N...共用資源
580A-580N...圖形核心
600...執行緒執行邏輯
602...像素著色器
604...執行緒調度器
606...指令快取記憶體
608A-608N...執行單元/執行單元陣列
610...取樣器
612...資料快取記憶體
614...資料埠
710...128位元格式
712...指令作業碼
713...索引欄位
714...指令控制欄位
716...exec-size欄位
718...目的地
720...src0

722...src1

724...SRC2

730...64位元壓緊指令格式/64位元格式

740...作業碼解碼

742...移動及邏輯群組

744...流量控制指令群組

746...雜項指令群組

748...平行數學群組

750...向量數學群組

1010...3D圖形應用程式

1012...著色器指令

1014...可執行指令

1016...圖形物件

1020...作業系統

1024...前端著色器編譯器

1026...使用者模式圖形驅動器

1027...後端著色器編譯器

1028...作業系統核心模式功能

1029...核心模式圖形驅動器

1034...通用處理器核心

1050...系統記憶體

I646502

發明摘要

雙面影印

※ 申請案號：104106522

※ 申請日：104.3.2

※IPC 分類：G06T 15/80 (2011.01)
G06T 1/20 (2006.01)**【發明名稱】(中文/英文)**

映射多重速率著色至單一程式之技術

MAPPING MULTI-RATE SHADING TO MONOLITHIC PROGRAMS

【中文】

在多重速率著色中，將一粗速率著色階段加入於現有像素速率階段之上以在對影像品質具有最小影響之情況下顯著改良效能。可將以不同速率評估之兩個著色階段映射至在處理器圖形單指令多資料 (SIMD) 引擎上執行之一個單片程式。在一個實施例中，多重速率著色允許一單一呈現遍次以一或多個不同速率執行著色程式碼：每個像素群、每個像素及每個樣本。

【英文】

In multi-rate shading, a coarse-rate shading phase is added on top of existing pixel-rate phase to significantly improve performance with minimum impact to image quality. Two shading phases evaluated at different rates may be mapped to one monolithic program running on processor graphics single instruction multiple data (SIMD) engines. In one embodiment, multi-rate shading allows a single rendering pass to execute shading code at one or more different rates: per group of pixels, per pixel, and per sample.

【代表圖】

【本案指定代表圖】：第 (3A) 圖。

【本代表圖之符號簡單說明】：

10-22...區塊

【本案若有化學式時，請揭示最能顯示發明特徵的化學式】：

(無)

申請專利範圍

1. 一種用於圖形處理之方法，其包含：
 - 定義針對像素的一群組之一粗對像素 (coarse-to-pixel) 映射；
 - 使用內插屬性進行粗像素著色；
 - 使用來自粗像素著色之資料及該內插屬性在用於該群組之一硬體處理器中進行像素著色；以及
 - 使用來自像素著色之資料及該內插屬性進行樣本著色。
2. 如請求項1之方法，其包括判定該群組中之哪些像素經點亮。
3. 如請求項1之方法，其包括使用同一核心於粗及像素著色。
4. 如請求項3之方法，其包括使用一單指令多資料著色器。
5. 如請求項4之方法，其包括使用同樣的通道於粗及像素著色兩者。
6. 如請求項1之方法，其包括在針對該群組進行粗著色後針對該群組進行像素著色。
7. 如請求項1之方法，其包括在運作一單指令多資料機器之同一核心程式中使用一指令用以將程式執行自粗速率切換至像素速率。
8. 如請求項1之方法，其包括一次粗著色多個四元素 (quad)，且接著依次地像素著色該等四元素中之每一者。

9. 一種用於圖形處理之方法，其包含：
 - 將粗及像素著色以階段方式映射至一單個單指令多資料程式內；
 - 判定重心空間中之像素位置；以及
 - 若於一群組中之所有像素未經點亮，則抑制該群組之像素。
10. 如請求項9之方法，其包括在一固定功能加速器中編譯粗對像素映射、像素點亮遮罩、像素群及重心。
11. 如請求項10之方法，其包括將像素速率階段實施為由該固定功能加速器所判定之像素群組上之一迴路。
12. 如請求項11之方法，其包括在每一迴路中設定一執行遮罩，使用粗對像素映射以攪和粗對像素速率階段之間的級間資料，及在每一迴路遍次(pass)中使用重心以內插每個像素屬性。
13. 一種儲存指令之一或多個非暫時性電腦可讀取媒體，該等指令由一處理器所執行以進行包括以下一序列之操作：
 - 定義針對像素的一群組之一粗對像素映射；
 - 使用內插屬性進行粗像素著色；
 - 使用來自粗像素著色之資料及該內插屬性在用於該群組之一硬體處理器中進行像素著色；以及
 - 使用來自像素著色之資料及該內插屬性進行樣本著色。
14. 如請求項13之媒體，該序列包括判定該群組中之哪些像

素經點亮。

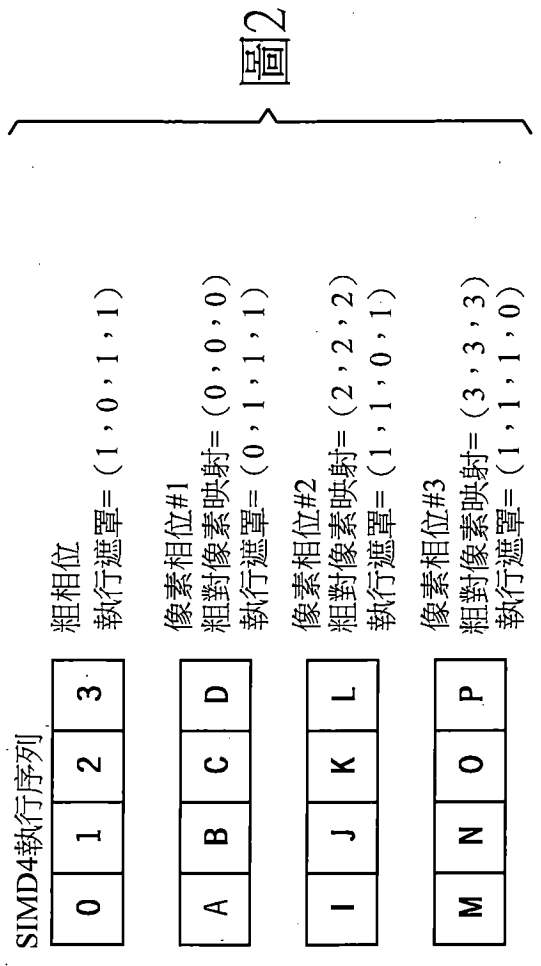
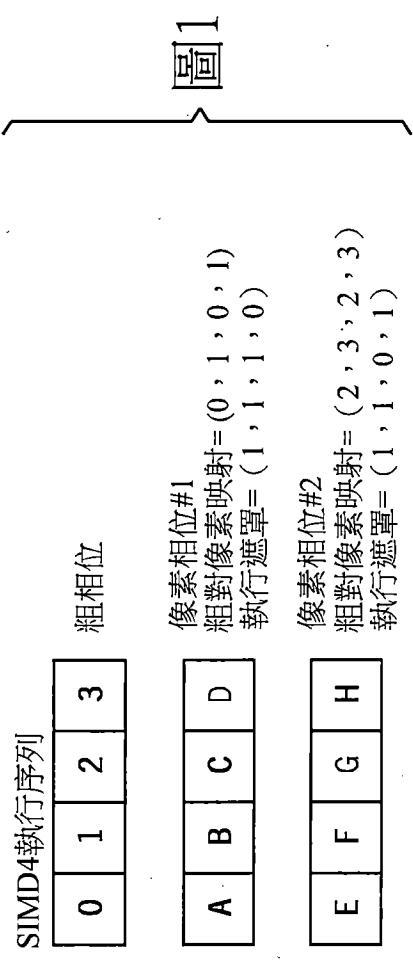
15. 如請求項13之媒體，該序列包括使用同一核心於粗及像素著色。
16. 如請求項15之媒體，該序列包括使用一單指令多資料著色器。
17. 如請求項16之媒體，該序列包括使用同樣的通道於粗及像素著色兩者。
18. 如請求項13之媒體，該序列包括在針對該群組進行粗著色後針對該群組進行像素著色。
19. 如請求項13之媒體，該序列包括在運作一單指令多資料機器之同一核心程式中使用一指令用以將程式執行自粗速率切換至像素速率。
20. 如請求項13之媒體，該序列包括一次粗著色多個四元素，且接著依次地像素著色該等四元素中之每一者。
21. 一種運算設備，其包括：
 - 一硬體處理器，其用以定義針對像素的一群組之一粗對像素映射，使用內插屬性進行粗像素著色，使用來自粗像素著色之資料及該內插屬性在用於該群組之一硬體處理器中進行像素著色，使用來自像素著色之資料及該內插屬性進行樣本著色；以及
 - 一儲存器，其耦接至該處理器。
22. 如請求項21之設備，該處理器用以判定該群組中之哪些像素經點亮。
23. 如請求項21之設備，該處理器用以使用同一核心於粗及

像素著色。

24. 如請求項23之設備，該處理器用以使用一單指令多資料著色器。
25. 如請求項24之設備，該處理器用以使用同樣的通道於粗及像素著色兩者。
26. 如請求項21之設備，該處理器用以在針對該群組進行粗著色後針對該群組進行像素著色。
27. 如請求項21之設備，該處理器用以在運作一單指令多資料機器之同一核心程式中使用一指令用以將程式執行自粗速率切換至像素速率。
28. 如請求項21之設備，該處理器用以一次粗著色多個四元素，且接著依次地像素著色該等四元素中之每一者。
29. 如請求項21之設備，其包括一記憶體控制器集線器。
30. 如請求項21之設備，其包括一無線收發器。

圖式

第104106522號專利申請案
圖式替換本
修正日期：107年9月26日



粗像素佈局: 1x2

0A	1B
C	D
2E	3F
G	H

粗像素佈局: 2x2

0A	B	1E	F
C	D	G	H
2I	J	3M	N
K	L	O	P

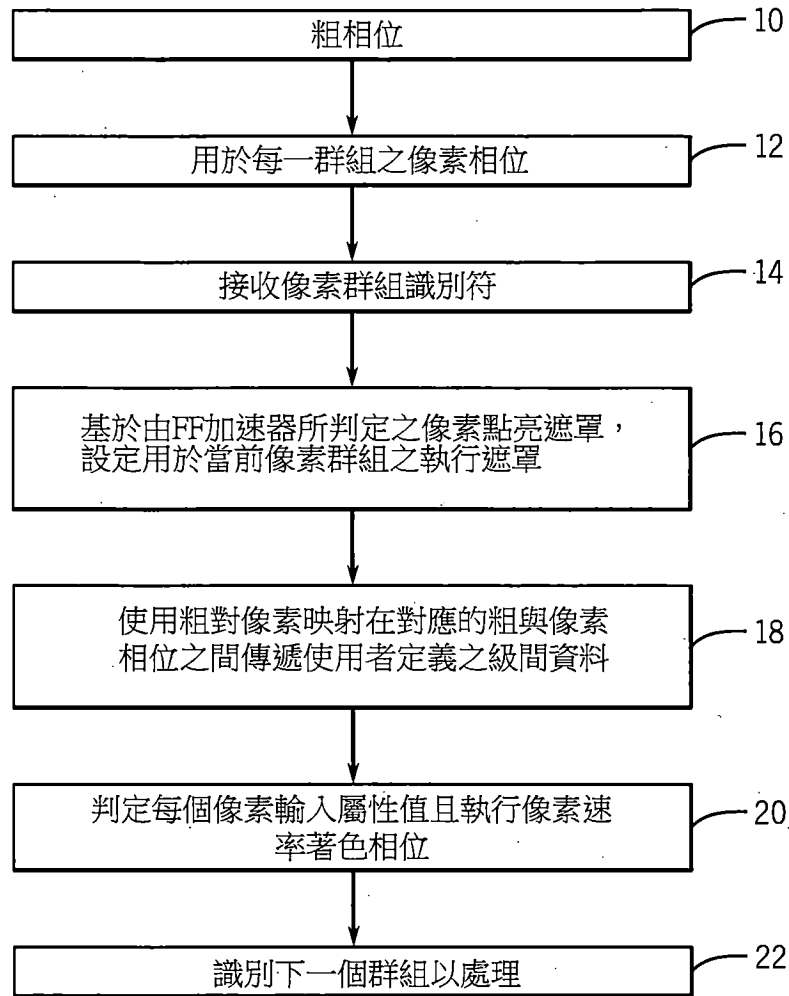


圖3A

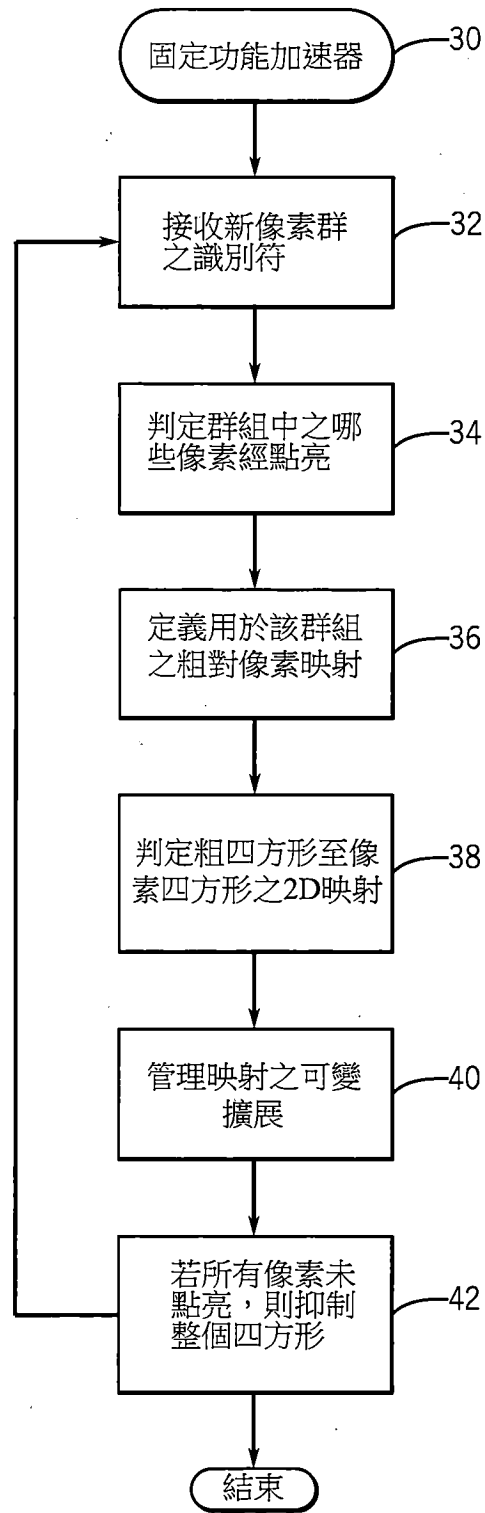


圖3B

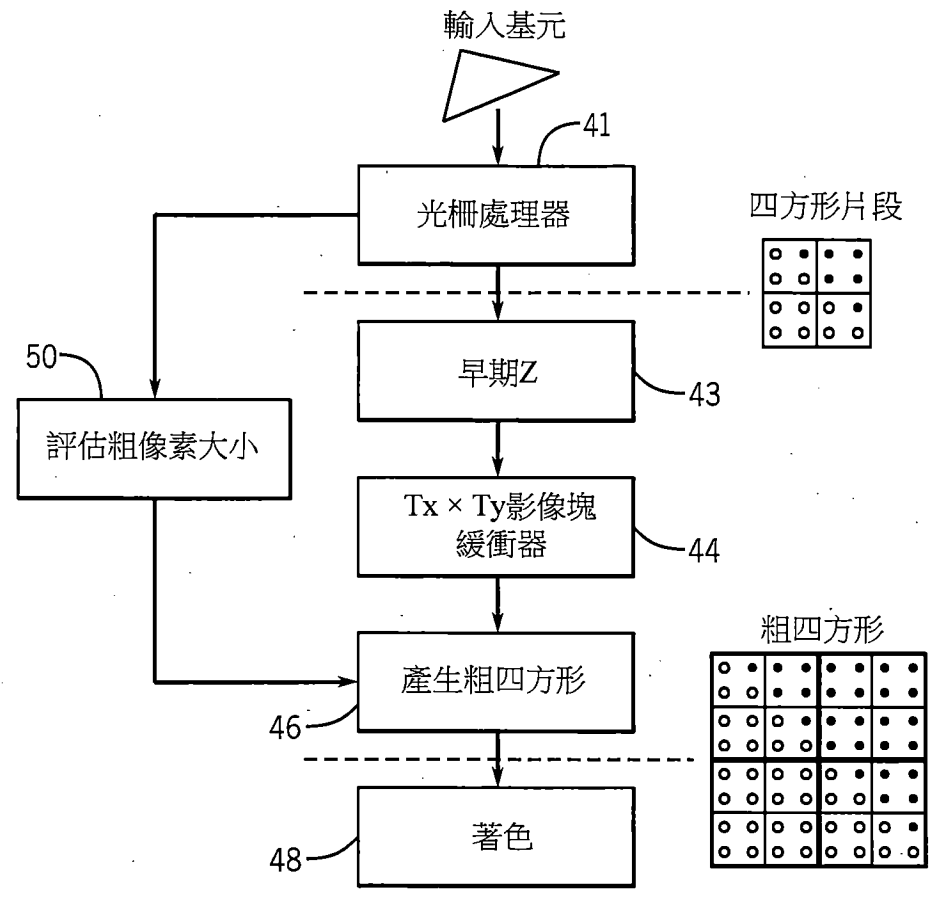


圖4

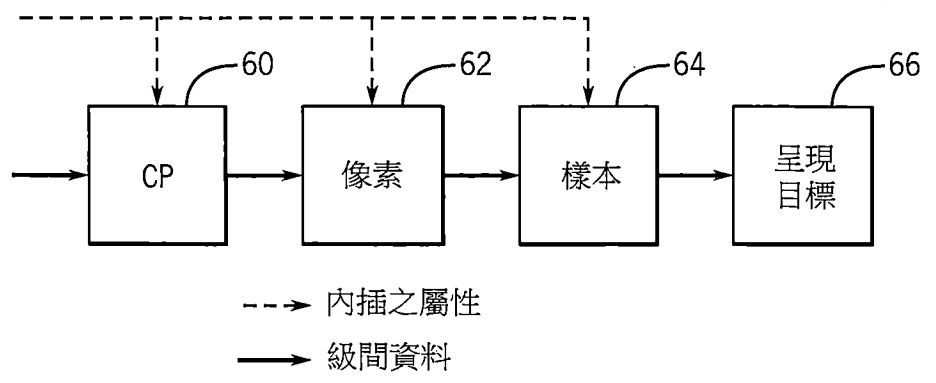


圖5

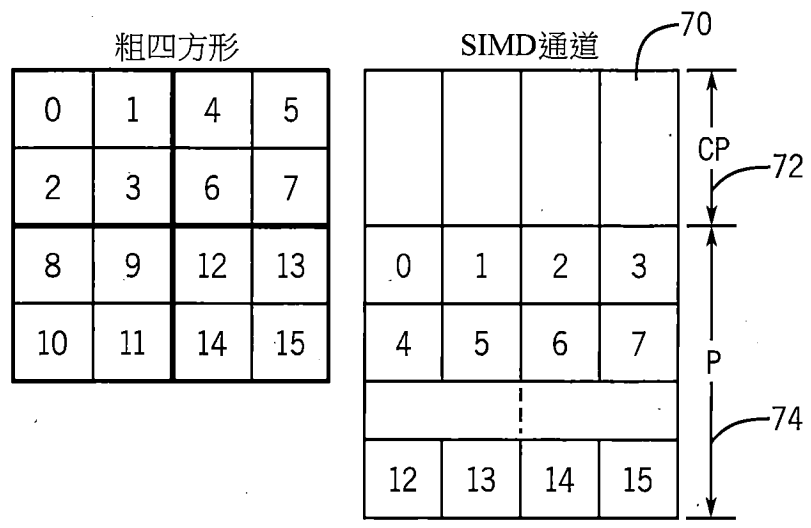


圖6

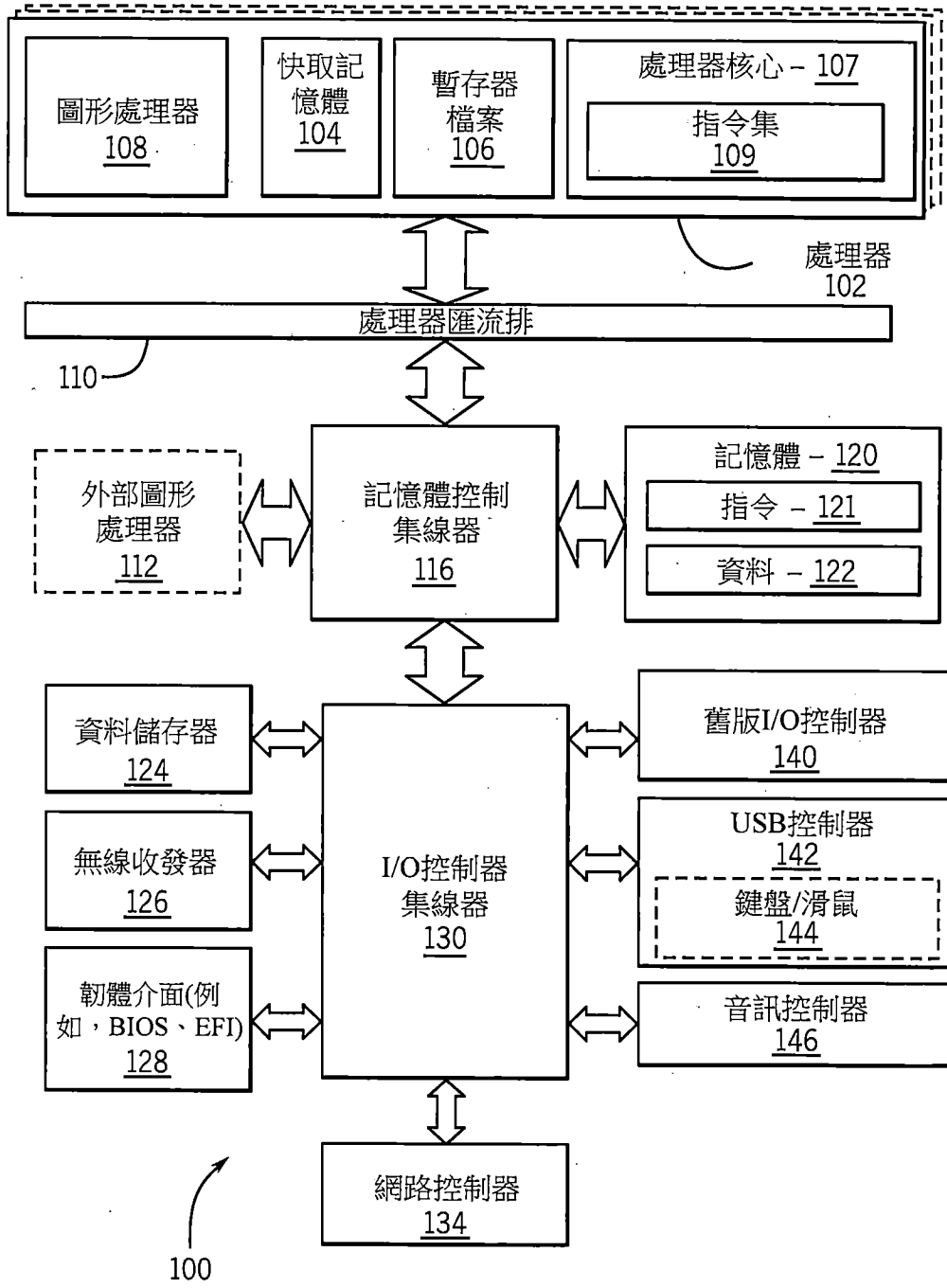


圖7

處理器200

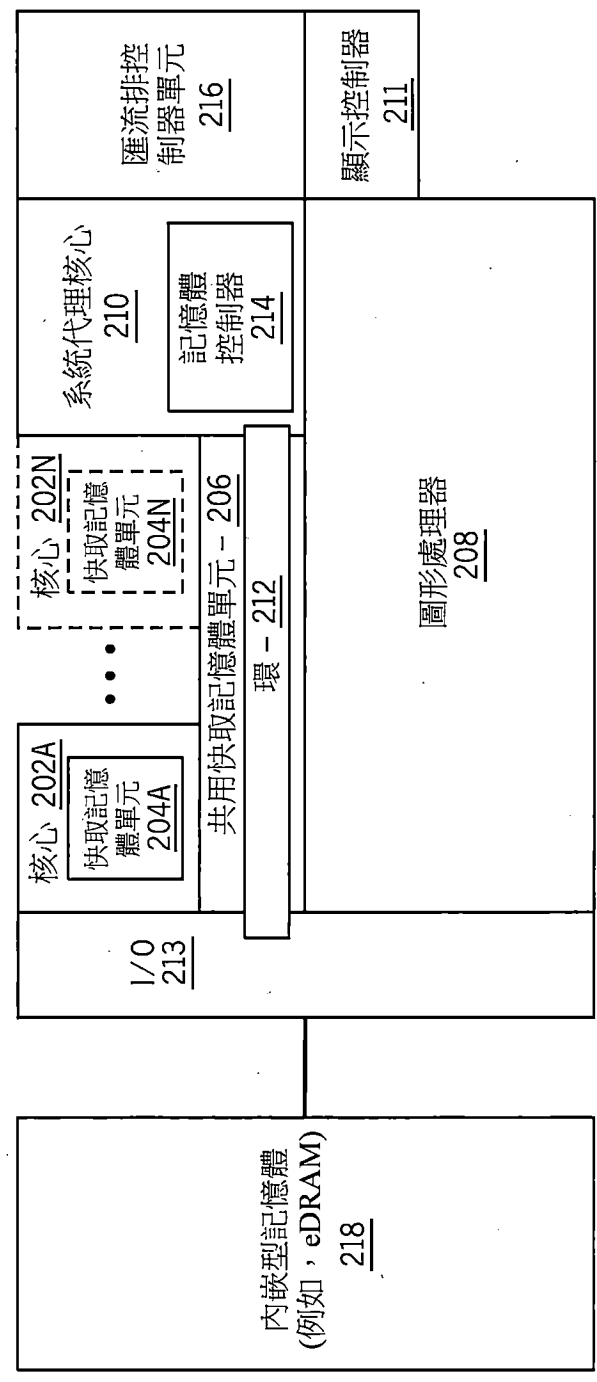
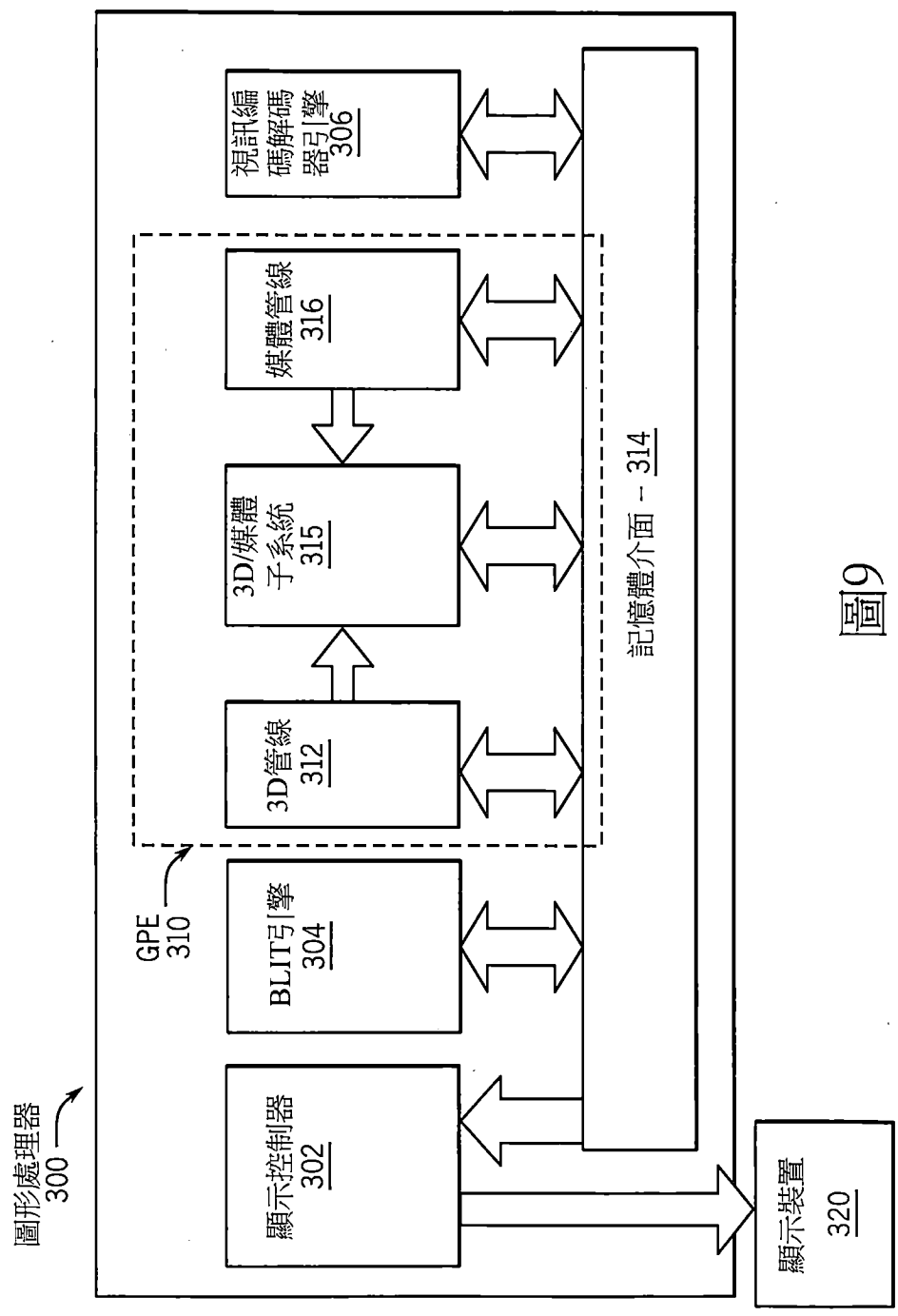


圖8



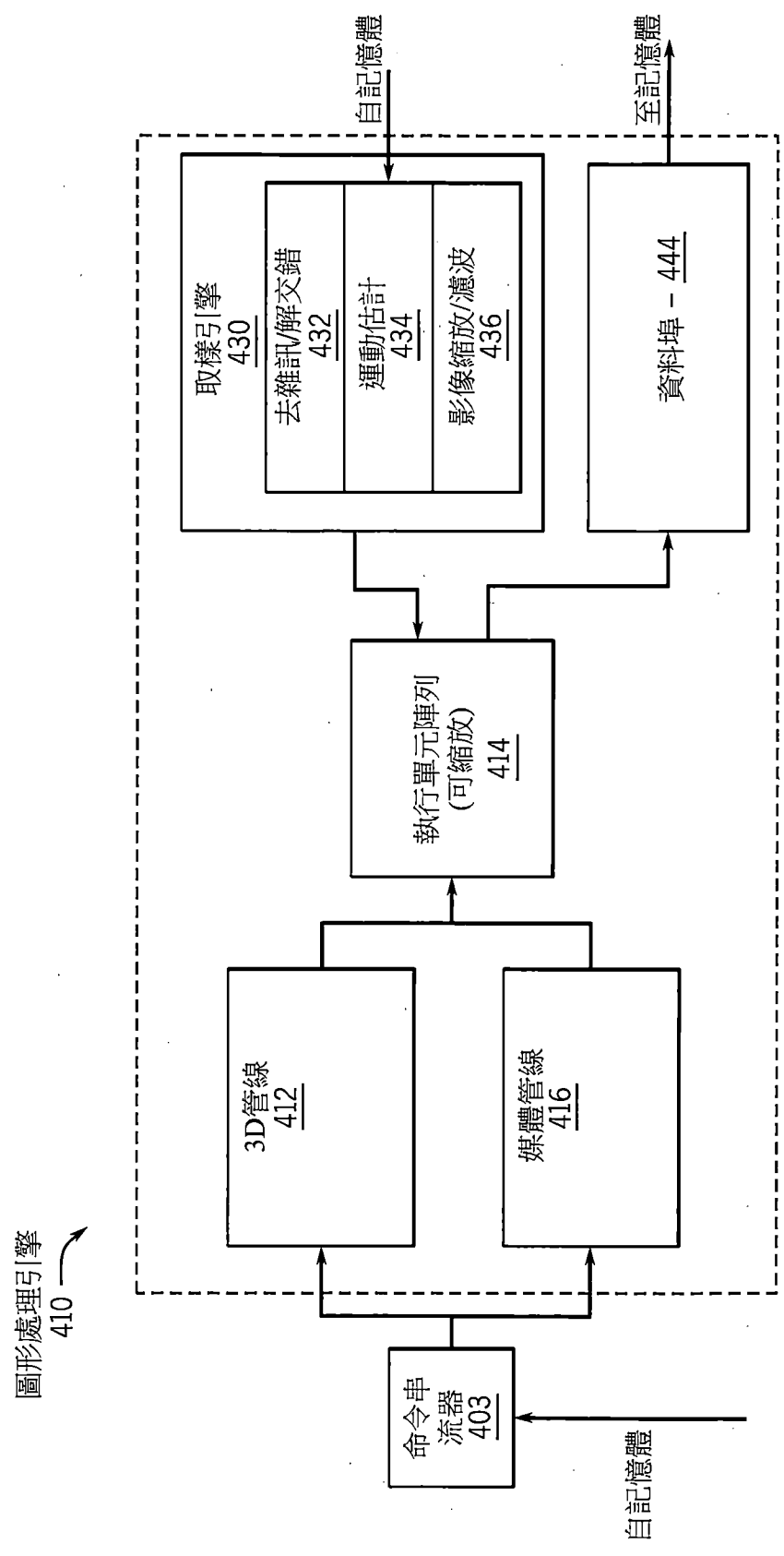


圖10

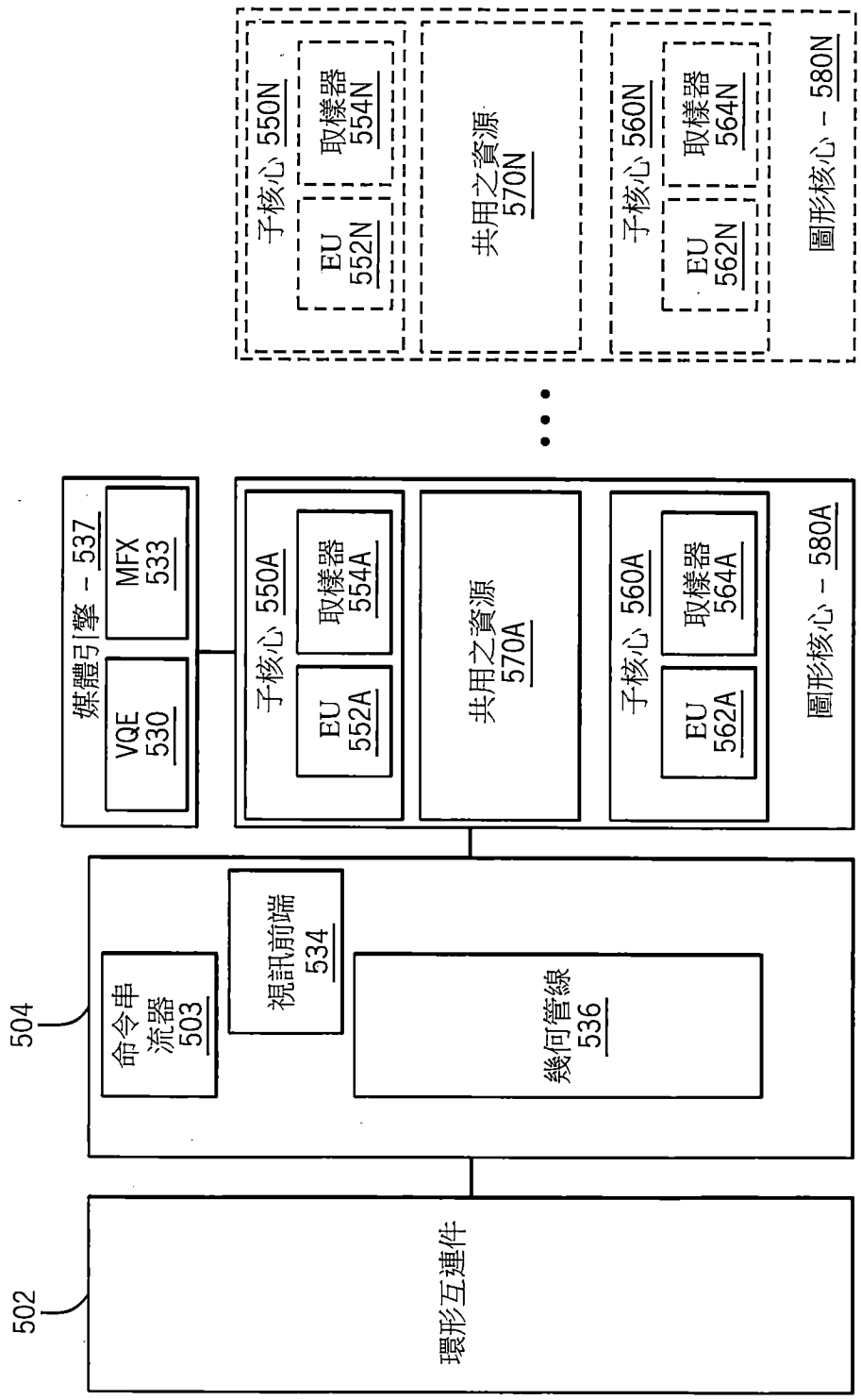


圖11

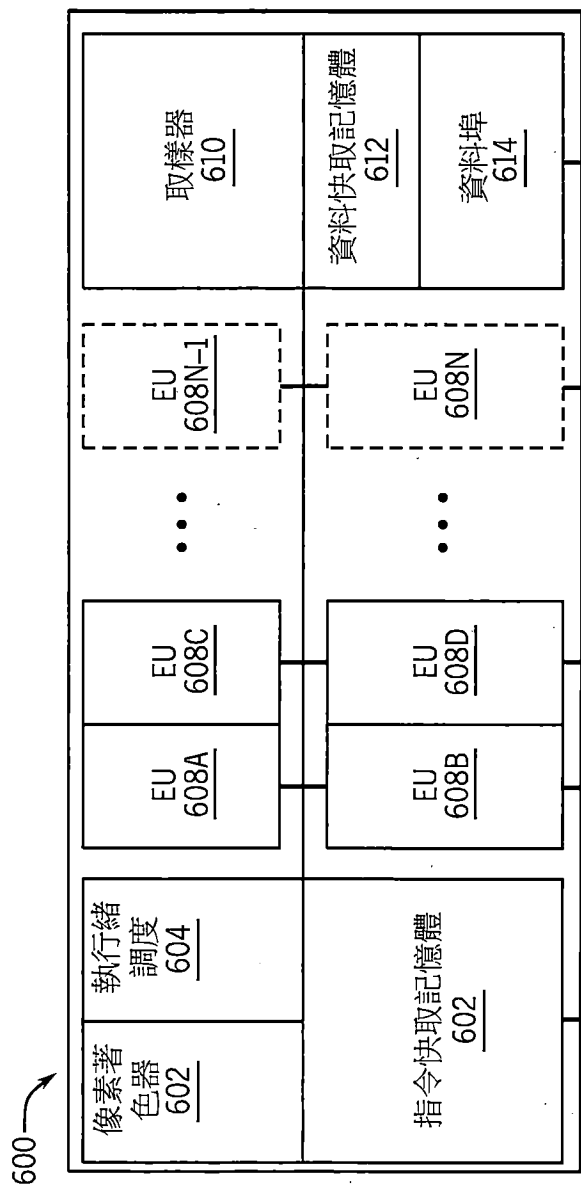


圖12

圖形核心指令格式

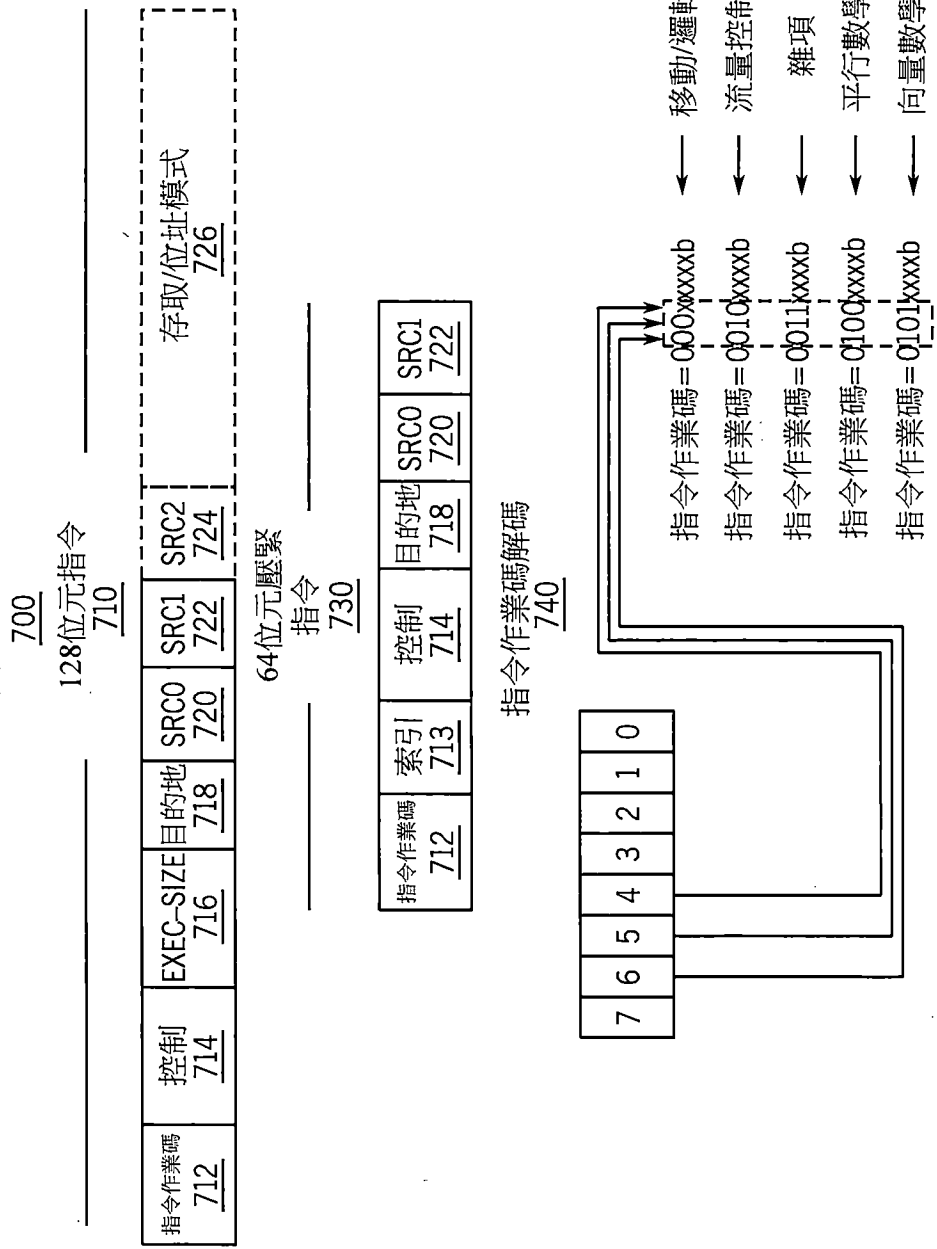


圖13

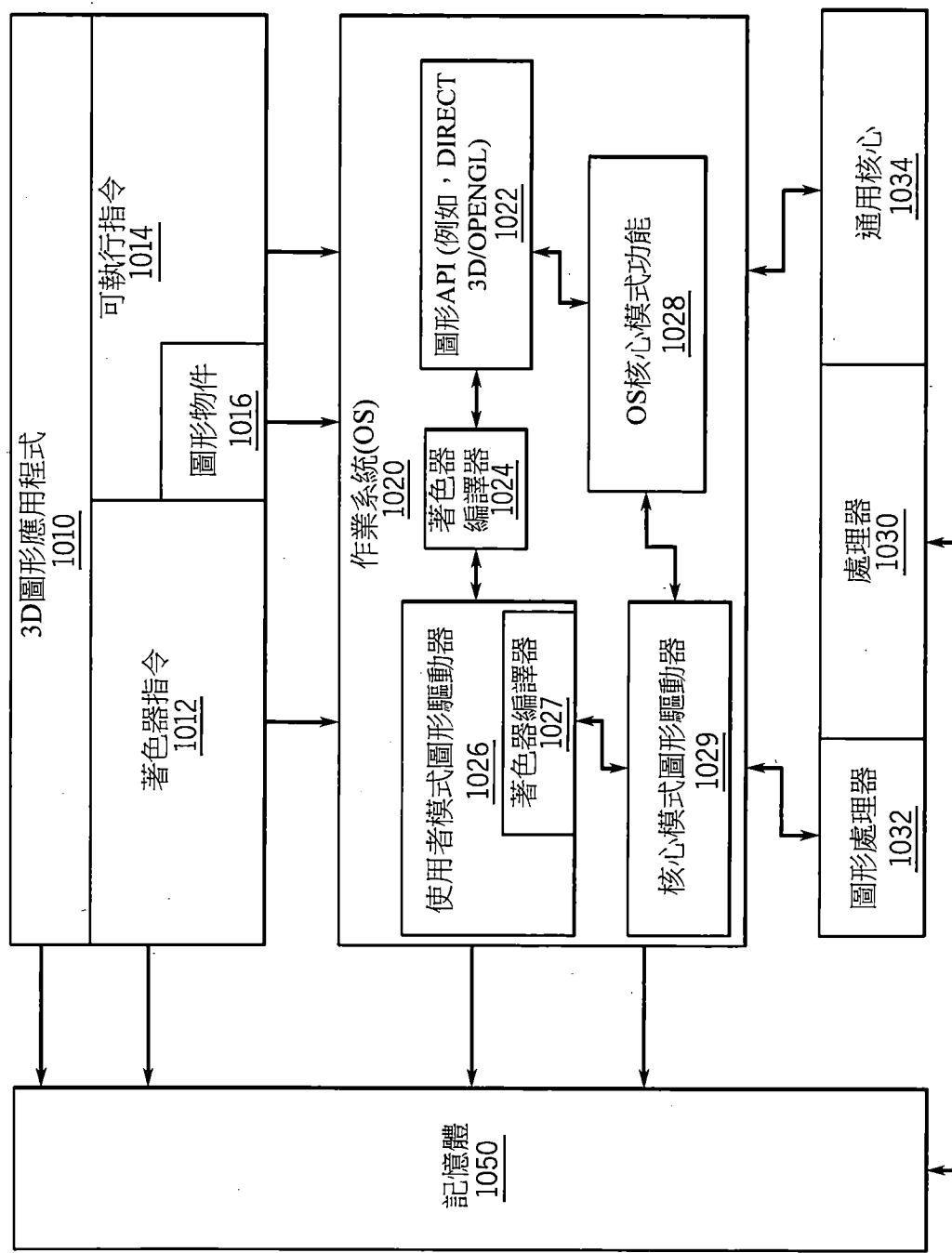


圖14