



(19) 中華民國智慧財產局

(12) 發明說明書公開本

(11) 公開編號：TW 201426633 A

(43) 公開日：中華民國 103 (2014) 年 07 月 01 日

(21) 申請案號：102132849 (22) 申請日：中華民國 102 (2013) 年 09 月 11 日
(51) Int. Cl. : **G06T15/80 (2011.01)** **G06T1/00 (2006.01)**
(30) 優先權：2012/09/11 美國 13/609,698
(71) 申請人：輝達公司 (美國) NVIDIA CORPORATION (US)
美國
(72) 發明人：艾弗瑞特 凱斯 W EVERITT, CASS W (US) ; 莫爾頓 亨利 P MORETON, HENRY
P (US)
(74) 代理人：李宗德
申請實體審查：有 申請專利範圍項數：10 項 圖式數：5 共 25 頁

(54) 名稱

圖形處理器可程式化著色器的紋理資料結構與其運作方法

TEXEL DATA STRUCTURE FOR GRAPHICS PROCESSING UNIT PROGRAMMABLE SHADER
AND METHOD OF OPERATION THEREOF

(57) 摘要

本發明提供一種圖形處理子系統與一種著色方法。在一具體實施例中，該子系統包括：(1)一記憶體，其設置成包含一紋理資料結構，該紋理資料結構係根據對應於一特定複合紋理的多個基礎紋理有那些係包含在該記憶體的單一頁面中；及(2)一圖形處理器，其設置成經由一資料匯流排與該記憶體進行通訊，並執行一著色器，以提取包含在該單一頁面中的該等多個基礎紋理來產生該特定複合紋理。

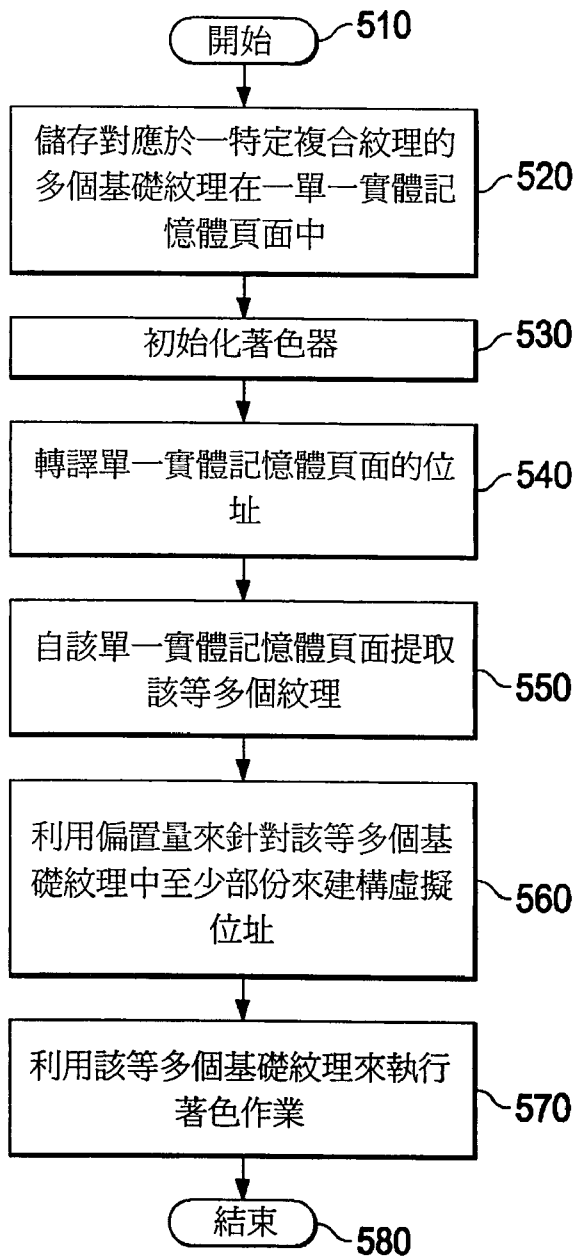


圖 5



(19) 中華民國智慧財產局

(12) 發明說明書公開本

(11) 公開編號：TW 201426633 A

(43) 公開日：中華民國 103 (2014) 年 07 月 01 日

(21) 申請案號：102132849

(22) 申請日：中華民國 102 (2013) 年 09 月 11 日

(51) Int. Cl. : **G06T15/80 (2011.01)**

G06T1/00 (2006.01)

(30) 優先權：2012/09/11 美國

13/609,698

(71) 申請人：輝達公司 (美國) NVIDIA CORPORATION (US)

美國

(72) 發明人：艾弗瑞特 凱斯 W EVERITT, CASS W (US) ; 莫爾頓 亨利 P MORETON, HENRY P (US)

(74) 代理人：李宗德

申請實體審查：有 申請專利範圍項數：10 項 圖式數：5 共 25 頁

(54) 名稱

圖形處理器可程式化著色器的紋理資料結構與其運作方法

TEXEL DATA STRUCTURE FOR GRAPHICS PROCESSING UNIT PROGRAMMABLE SHADER AND METHOD OF OPERATION THEREOF

(57) 摘要

本發明提供一種圖形處理子系統與一種著色方法。在一具體實施例中，該子系統包括：(1)一記憶體，其設置成包含一紋理資料結構，該紋理資料結構係根據對應於一特定複合紋理的多個基礎紋理有那些係包含在該記憶體的單一頁面中；及(2)一圖形處理器，其設置成經由一資料匯流排與該記憶體進行通訊，並執行一著色器，以提取包含在該單一頁面中的該等多個基礎紋理來產生該特定複合紋理。

發明摘要

※ 申請案號：102132849

※ 申請日：102 年 09 月 11 日

※IPC 分類：G06T15/80 (2011.01)

G06T1/00 (2006.01)

【發明名稱】(中文/英文)

圖形處理器可程式化著色器的紋理資料結構與其運作方法

TEXEL DATA STRUCTURE FOR GRAPHICS PROCESSING UNIT
PROGRAMMABLE SHADER AND METHOD OF OPERATION THEREOF

【中文】

本發明提供一種圖形處理子系統與一種著色方法。在一具體實施例中，該子系統包括：(1)一記憶體，其設置成包含一紋理資料結構，該紋理資料結構係根據對應於一特定複合紋理的多個基礎紋理有那些係包含在該記憶體的單一頁面中；及(2)一圖形處理器，其設置成經由一資料匯流排與該記憶體進行通訊，並執行一著色器，以提取包含在該單一頁面中的該等多個基礎紋理來產生該特定複合紋理。

【英文】

A graphics processing subsystem and a method of shading. In one embodiment, the subsystem includes: (1) a memory configured to contain a texel data structure according to which multiple primitive texels corresponding to a particular composite texel are contained in a single page of the memory and (2) a graphics processing unit configured to communicate with the memory via a data bus and execute a shader to fetch the multiple primitive texels contained in the single page to create the particular composite texel.

【代表圖】

【本案指定代表圖】：圖 5。

【本代表圖之符號簡單說明】：無。

【本案若有化學式時，請揭示最能顯示發明特徵的化學式】：無。

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)

圖形處理器可程式化著色器的紋理資料結構與其運作方法

TEXEL DATA STRUCTURE FOR GRAPHICS PROCESSING UNIT
PROGRAMMABLE SHADER AND METHOD OF OPERATION
THEREOF

【技術領域】

【0001】 本申請案概略關於電腦繪圖(computer graphics)，更特定而言係關於圖形處理器(GPU, “Graphics processing units”)的資料結構。

【先前技術】

【0002】 許多電腦圖形影像係由數學地模型化光線與由一給定視角的三維場景之互動所產生。此程序稱之為「顯像」(rendering)，可產生由該給定視角之場景的二維影像，並類似於拍攝現實世界場景的相片。

【0003】 對於電腦繪圖的需求增加，特別是即時電腦繪圖，可被調整來加速該顯像處理之具有圖形處理子系統的電腦系統已更為普遍。在這些電腦系統中，該顯像處理被分擔在一電腦的通用中央處理單元(CPU, “Central processing unit”)與該圖形處理子系統之間。基本上，CPU 執行高階運算，例如決定在一給定場景中物件的位置、動作與碰撞。由這些高階運算當中，該 CPU 產生一組顯像命令，以及定義該等需要的經顯像影像或多個影像之資料。例如，顯像命令與資料能夠定義一場景的場景幾何、照明、著色、紋理、動作及/或相機參數。圖形處理子系統由該組顯像命令與資料產生一或多個經顯像的影像。

【0004】 場景幾何基本上由幾何基元(geometric primitives)來表示，例如點、線、多邊形(例如三角形與四邊形)，以及由一或多個二維或三維頂點

所定義的曲面。每一頂點可具有額外的純量或向量屬性，用於定義品質，例如該頂點的色彩、透光度、照明、著色與動畫，以及其相關的幾何基元。

【0005】 許多圖形處理子系統可藉由一應用程式化介面(API, “Application programming interface”)成爲高度可程式化，除此之外能夠實作複雜的照明與著色演算法。爲了利用此可程式化特性，應用能夠包括一或多個圖形處理子系統程式，其由圖形處理子系統平行於由該 CPU 執行的一主程式來執行。雖然不只限定於實作著色和照明演算法，這些圖形處理子系統時常稱之爲「著色程式」、「可程式化著色器」、或僅稱之爲「著色器(shader)」。

【發明內容】

【0006】 本發明一種態樣提供一種圖形處理子系統。在一具體實施例中，該子系統包括：(1)一記憶體，其設置成包含一紋理資料結構，其根據對應於一特定複合紋理有那些多個基礎紋理來包含在該記憶體的單一頁面中；及(2)一圖形處理器，其設置成經由一資料匯流排與該記憶體進行通訊，並執行一著色器，以提取包含在該單一頁面中的該等多個基礎紋理來產生該特定複合紋理。

【0007】 在另一具體實施例中，該子系統包括：(1)一記憶體，其設置成包含一紋理資料結構，其根據對應於一特定複合紋理有那些多個基礎紋理來包含在該記憶體的單一頁面中；及(2)一圖形處理器，其設置成經由一資料匯流排與該記憶體進行通訊，且包括一複數個串流多處理器，該等串流多處理器中至少一者設置成執行一著色器。在一具體實施例中，該著色器設置成：(2a) 啓動該等多個基礎紋理的提取，該啓動牽涉到單一位址轉譯，以及(2b)利用該等多個基礎紋理來顯像該特定複合紋理。

【0008】 本發明另一種態樣提供一種著色方法。在一具體實施例中，該方法包括以下步驟：(1) 初始化一著色器來執行關於一特定複合紋理的一

著色作業，(2) 啟動多個基礎紋理的提取，該啟動造成對應於含有該等多個基礎紋理的一單一頁面的一位址轉譯，(3) 自該單一頁面提取該等多個基礎紋理，及(4)利用該等多個基礎紋理來執行該著色作業。

【圖式簡單說明】

【0009】 現在請參照以下配合該等附屬圖式的說明，其中：

【0010】 圖 1 為在其中可實作本發明之一或多種態樣之一種電腦系統的一具體實施例之方塊圖；

【0011】 圖 2 為駐在於圖 1 的該 GPU 之內的一可程式化圖形管線的一具體實施例之方塊圖；

【0012】 圖 3 為圖 2 之圖形管線的一幾何著色引擎之一具體實施例的方塊圖；

【0013】 圖 4 為圖 1 的該晶片上 GPU 記憶體之一部份的一具體實施例之方塊圖；及

【0014】 圖 5 為一種著色方法的一具體實施例之流程圖。

【實施方式】

【0015】 可瞭解到於進行它們的著色演算法時，可程式化著色器經常自一虛擬記憶體空間讀取或「提取」多個基礎紋理元件或「紋理」(texels)，以產生一或多個複合紋理。

【0016】 但是，因為一虛擬記憶體空間為實體記憶體空間的一種抽象概念，需要一種位址轉譯程序來轉譯在該虛擬記憶體空間中每一基礎紋理的該位址成為一實體位址。然後該實體位址用於自實體記憶體提取該基礎紋理。因為現今的電腦系統利用快取記憶體來更快地運作，包括含有該基礎紋理的該位址的一頁面亦被載入到快取記憶體(或簡稱為「被快取」)，以達到位在鄰近位址的資料能夠自該快取記憶體讀取之可能性。

【0017】 可惜地是，位址轉譯相當複雜，並消耗很多處理資源。爲此原因，較佳地是降低需要採取之位址轉譯的量。在此方面，已經發現到較佳地是使用一轉譯旁看緩衝器(TLB, “Translation lookaside buffer)來儲存新近轉譯的位址，並藉此提供一種機制而可重新使用包含在該 TLB 中的位址轉譯。可惜地是，當位址轉譯的頻率增加時，一「命中」(hit)(即 TLB 包含一個需要轉譯的虛擬位址之項目)的可能性即減少，因而降低 TLB 的有效性。

【0018】 另外，自主記憶體載入頁面至快取記憶體會消耗匯流排頻寬。爲此原因，較佳地是降低需要採取之快取化的量。

【0019】 在此可瞭解到用於儲存基礎紋理的習用資料結構，其中基礎紋理根據物件群組化，習用資料結構幾乎總是會造成對應於一特定複合紋理的該等多個基礎紋理會跨越記憶體的多個頁面。因此，提取要產生一特定複合紋理所需要的該等多個基礎紋理通常不僅牽涉到過多的位址轉譯，亦有過多的頁面快取化。

【0020】 在此可瞭解到如果重新形成該等基礎紋理資料結構，使得關於一特定複合紋理的多個基礎紋理被包含在一單一記憶體頁面中時，即可減少位址轉譯，並可能大量地減少。因此，本文將介紹了一基礎紋理資料結構的多種具體實施例，及一種運作一基礎紋理資料結構的方法，其中包含有對應於一特定複合紋理的多個基礎紋理，或是「交錯(interleaved)」在一單一記憶體頁面中。因此，一單一位址轉譯(或 TLB 命中)可允許取得多個基礎紋理。在某些具體實施例中，對應於一特定複合紋理的所有該等基礎紋理被交錯在一單一記憶體空間中，而允許關於該特定複合紋理的所有基礎紋理之取得可利用一單一位址轉譯(或 TLB 命中)來提取。

【0021】 亦可瞭解到當交錯多個基礎紋理時，一特定複合紋理在一給定頁面中佔用的空間之比例會隨著該紋理的大小增加而增加。換言之，交錯可減少一給定頁面能夠包含的複合紋理之數目。因此，當提取一特定頁

面時，有可能包含較少之不需要的基礎紋理。這樣比較好，係因為提取不需要的基礎紋理會浪費頻寬與暫時性儲存器(例如快取)的空間。當產生稀疏紋理的影像或其部份(即那些具有較簡單的複合紋理者)時，交錯即特別具有好處。在這些情況下，可使該等基礎紋理較小(它們的粒度降低)，增加可包含在一記憶體頁面中的基礎紋理的數目，並進一步降低浪費。例如，如果一未交錯的基礎紋理之磁磚大小為 256×256 ，將其與四個其它的基礎紋理可允許其磚(tile)大小降低到 128×128 。如果一著色器僅需要該原始 256×256 磚的一個象限來產生一特定複合紋理，該基礎紋理可用該原始空間的四分之一來表示。同樣地，較容易地追蹤已被提取的該等基礎紋理。

【0022】 該基礎紋理資料結構與方法的一些具體實施例基於 2 的次方來將該等基礎紋理交錯在該等頁面之內。例如，如果基礎紋理被交錯在 4 kB 區塊中，該等 4 kB 基礎紋理區塊之間的距離或「步幅(stride)」亦為 4 kB 的 2 次方的倍數，在一給定空間中對於該等基礎紋理區塊之每一者建構一位址會牽涉到將一或多個最有效位元(MSB, “Most significant bits”)插入到該最低的 4 kB 基礎紋理區塊的位址。該最低的 4 kB 基礎紋理區塊之位址可被適當地視為一基礎位址，且該經插入的 MSB 可被適當地視為一偏置量。因此，舉例來說，可插入一單一”0” MSB 位址偶數步幅，並插入一單一”1” MSB 位址奇數步幅。

【0023】 在說明該基礎紋理資料結構與方法的多種具體實施例之前，將說明可於其中實施或執行該基礎紋理資料結構的一電腦系統。

【0024】 圖 1 為在其中可實作本發明之一或多種態樣之一種電腦系統 100 的一具體實施例之方塊圖。電腦系統 100 包括一系統資料匯流排 132、一中央處理單元(CPU) 102、輸入裝置 108、一系統記憶體 104、一圖形處理子系統 106、及顯示裝置 110。在其它具體實施例中，CPU 102、圖形處理子系統 106 的部份、系統資料匯流排 132 或其任何組合可被整合到一單一處理單元中。另外，圖形處理子系統 106 的功能可被包括在一晶片組中，

或是在某種其它型式的特定目的處理單元或共處理器中。

【0025】 如所示，系統資料匯流排 132 連接 CPU 102、輸入裝置 110、系統記憶體 104、與圖形處理子系統 106。在其它具體實施例中，系統記憶體 100 可直接連接到 CPU 102。CPU 102 自輸入裝置 108 接收使用者輸入、執行儲存在系統記憶體 104 中的程式化指令、運作儲存在系統記憶體 104 中的資料，並設置圖形處理子系統 106 來在該圖形管線中執行特定工作。系統記憶體 104 基本上包括用於儲存由 CPU 102 與圖形處理子系統 106 處理的程式化指令與資料的動態隨機存取記憶體(DRAM, “Dynamic random access memory”)。圖形處理子系統 106 接收由 CPU 102 傳送的指令，並處理該等指令來在顯示裝置 110 上顯像與顯示圖形影像。

【0026】 亦如所示，系統記憶體 110 包括一應用程式 112、一應用程式化介面(API)114、與一圖形處理器(GPU)驅動器 116。應用程式 112 產生對 API 114 的呼叫，藉以產生一組所需要的結果(其基本上為一系列圖形影像的型式)。應用程式 112 亦傳送零個或多個高階著色程式到 API 114 在 GPU 驅動器 116 內進行處理。該等高階著色程式基本上為高階程式化指令的來源碼文字，其被設計成在圖形處理子系統 106 內的一或多個著色引擎上運作。API 114 的功能基本上係實作在 GPU 驅動器 116 之內。GPU 驅動器 116 設置成轉譯該高階著色程式成為機器碼著色程式，其基本上針對一特定種類的著色引擎進行最佳化(例如頂點、幾何或片段)。

【0027】 圖形處理子系統 106 包括一圖形處理器(GPU)118、一晶片上 GPU 記憶體 122、一晶片上 GPU 資料匯流排 136、一 GPU 本地記憶體 120、與一 GPU 資料匯流排 134。GPU 118 設置成經由晶片上 GPU 資料匯流排 136 與晶片上 GPU 記憶體 122 進行通訊，並經由 GPU 資料匯流排 134 與 GPU 本地記憶體 120 進行通訊。GPU 118 可接收自 CPU 102 傳送的指令、處理該等指令來顯像圖形資料與影像、並儲存這些影像在 GPU 本地記憶體 120 中。然後，GPU 118 可以在顯示裝置 110 上顯示儲存在 GPU 本地記憶

體 120 中的某些圖形影像。

【0028】 GPU 118 包括一或多個串流多重處理器 124。串流多重處理器 124 之每一者能夠同時間執行相當大量的執行緒。較佳地是，串流多重處理器 124 之每一者可被程式化來執行關於許多種應用之處理工作，其包括但不限於線性與非線性資料轉換、視訊及/或音訊資料的過濾、模型化作業(例如應用物理來決定物件的位置、速度與其它屬性)等等。另外，串流多重處理器 124 之每一者可設置成一著色引擎，其中包括一或多個可程式化著色器，其每一者執行一機器碼著色程式(即一執行緒)來執行影像顯像作業。GPU 118 可具有任何數量的晶片上 GPU 記憶體 122 與 GPU 本地記憶體 120，包括沒有任何記憶體，並可使用晶片上 GPU 記憶體 122、GPU 本地記憶體 120 與系統記憶體 104 以任何組合來進行記憶體作業。

【0029】 晶片上 GPU 記憶體 122 設置成包括 GPU 可程式碼 128 與晶片上緩衝器 130。GPU 可程式碼 128 可自 GPU 驅動器 116 經由系統資料匯流排 132 傳送到晶片上 GPU 記憶體 122。GPU 可程式碼 128 可包括一機器碼(machine code)頂點(vertex)著色程式、一機器碼幾何著色程式、一機器碼片段著色程式、或其每一者任何數目的變化。晶片上緩衝器 130 基本上用於儲存需要快速存取之著色資料，藉此降低在該圖形管線中該等著色引擎的遲滯。因為晶片上 GPU 記憶體 122 會佔用寶貴的晶粒面積，因此相當昂貴。

【0030】 GPU 本地記憶體 120 基本上包括較便宜的晶片外(off-chip)動態隨機存取記憶體(DRAM)，亦用於儲存 GPU 118 使用的資料與程式。如所示，GPU 本地記憶體 120 包括一像框緩衝器 126。像框緩衝器 126 儲存可用於驅動顯示裝置 110 的至少一二維表面的資料。另外，像框緩衝器 126 可包括一個以上的二維表面，所以 GPU 118 能夠顯像到一二維表面，而使用一第二二維表面來驅動顯示裝置 110。

【0031】 顯示裝置 110 為一種能夠發出對應於一輸入資料信號的視

覺影像之一或多種輸出裝置。例如，一顯示裝置可使用一陰極射線管(CRT, “Cathode ray tube”)監視器、一液晶顯示器、或任何其它適當顯示系統來建構。傳送到顯示裝置 110 的該等輸入資料信號基本上藉由掃描輸出儲存在像框緩衝器 126 中的一或多個影像資料之像框的該等內容來產生。

【0032】 圖 2 為駐在於圖 1 的 GPU 100 內的一可程式化圖形管線 200 之一具體實施例的方塊圖。圖形管線 200 轉換 3-D 影像成爲 2-D 影像。如所示，圖形管線 200 包括一資料組合器 202、一頂點著色引擎 204、一基元組合器 206、一幾何著色引擎 208、一掃描場解析器 210、一片段著色引擎 212、一掃描場作業單元 214、以及圖 1 的像框緩衝器 126。

【0033】 資料組合器 202 爲一種固定功能的單元，用於收集高級數表面、基元和類似者的頂點資料，並輸出頂點資料到頂點著色引擎 204。資料組合器 202 可匯集來自儲存在系統記憶體 104、GPU 本地記憶體 120 與晶片上 GPU 記憶體 122 內的緩衝器之資料，以及來自用於指定頂點屬性的應用程式 112 之 API 呼叫。頂點著色引擎 204 爲一可程式化執行單元，例如串流多重處理器 124-1，其設置成執行一機器碼頂點著色程式用於處理由該頂點著色程式指定的頂點資料。例如，頂點著色引擎 204 可被程式化來轉換來自一物件式座標表示(物件空間)的該頂點資料到另外一種座標系統，例如世界空間(world space)或正規化的裝置座標(NDC, “Normalized device coordinates”)空間。頂點處理單元 204 可讀取及寫入資料，該資料係儲存在 GPU 本地記憶體 120 與晶片上 GPU 記憶體 122 中。

【0034】 基元組合器 206 爲一種固定功能單元，其接收來自頂點著色引擎 204 的已處理頂點資料，並建構圖形基元，例如點、線、三角形或類似者，用於由幾何著色引擎 208 做處理。在其它具體實施例中，一第二基元組合器(未示出)可被包括在資料流經 GPU 118 當中幾何著色引擎 208 之後。幾何著色引擎 208 爲一種可程式化執行單元，例如串流多重處理器 124-1，其設置成執行一機器碼幾何著色程式，用於處理由該幾何著色程式所指定

自基元組合器 206 接收的圖形基元。幾何著色引擎 208 可被程式化來產生零個或多個新的圖形基元並計算參數，例如平面公式係數，其用於掃描場化該等新的圖形基元。例如，用於立方映射顯像時，幾何著色引擎 208 可設置成處理輸入基元，使該等輸入基元變為六倍，並發出最多到六組的輸出基元，每一者用於一立方體的一面。幾何著色引擎 208 可讀取與寫入儲存在 GPU 本地記憶體 120 與晶片上 GPU 記憶體 122 中的資料。幾何著色引擎 208 輸出該等參數及新的圖形基元到掃描場解析器 210。掃描場解析器 210 為一種固定功能單元，其掃描該等新的圖形基元，並輸出片段與覆蓋資料到片段著色引擎 212。

【0035】 片段著色引擎 212 為一可程式化執行單元，例如串流多重處理器 124-1，其設置成執行一機器碼片段著色程式用於處理由該機器碼片段著色程式所指定的自掃描場解析器 210 接收的片段。例如，片段著色引擎 212 可被程式化來執行作業，例如透視修正、紋理映射、著色、混色及類似者，以產生被輸出到掃描場解析器作業單元 214 的經著色的片段。片段著色引擎 212 可讀取與寫入儲存在 GPU 本地記憶體 120 與晶片上 GPU 記憶體 122 中的資料。掃描場解析器作業單元 214 選擇性地執行固定功能運算，例如近與遠平面裁切和掃描場解析器運算，例如網版(stencil)、z 測試(z test) 與類似者，並輸出畫素資料做為已處理的圖形資料用於儲存在 GPU 本地記憶體 120 中的一緩衝器中，例如像框緩衝器 126。

【0036】 圖形管線 200 的某些具體實施例包括一鑲嵌狀格局 (tessellation)引擎(未示出)，其設置成轉換低細節細分表面成為較高細節的基元。概言之，鑲嵌狀格局磚化高級數表面成為更適於顯像的結構。

【0037】 圖 3 為圖 2 的圖形管線 200 之一幾何著色引擎 208 的一具體實施例之方塊圖。如所示，幾何著色引擎 208 為一串流多重處理器 124，其設置成根據一幾何著色程式執行幾何著色作業。在串流多重處理器 124 中每一串流處理器設置成一個別的幾何著色器 302 用於執行一單一執行緒，

其為該幾何著色程式的一個實例。在一具體實施例中，串流多重處理器 124 包括 32 個串流處理器，藉以提供給幾何著色引擎 208 有 32 個個別的幾何著色器 302。在其它具體實施例中，有多於或少於 32 個串流處理器，以及幾何著色器 302。

【0038】 在說明於其中可實施或進行該基礎紋理資料結構的一種電腦系統之後，將說明該等基礎紋理資料結構與方法之多種具體實施例。

【0039】 圖 4 為圖 1 的該晶片上 GPU 記憶體 122 之一部份的一具體實施例之方塊圖。GPU 記憶體 122 被區分成頁面(pages)。一個頁面 410 含有多個對應於一特定複合紋理的基礎紋理的區塊。在多種具體實施例中，該等基礎紋理包括至少一個基礎紋理屬性，例如：正常、擴散或鏡面色彩、閃亮、反射、透光、白熾、半透明、折射及顛簸。

【0040】 在圖 4 的具體實施例中，頁面 410 包含對應於一特定複合紋理 1 的多個基礎紋理的區塊(即區塊 411-415)。因此，對應於複合紋理 1 的多個(兩個或更多)基礎紋理可利用一單一位址轉譯或 TLB 命中來提取。同時在圖 4 的該具體實施例中，頁面 410 包含對應於紋理 1 的所有該等基礎紋理。因此，一單一位址轉譯皆需要提取對應於複合紋理 1 的區塊 411-415。

【0041】 頁面 410 亦包含對應於另一個特定複合紋理，即複合紋理 2 的多個基礎紋理，即區塊 421-425。頁面 410 的某些具體實施例包含對應於其它複合紋理的多個基礎紋理(未示出)。當區塊 421-425 連同對應於紋理 1 的該等基礎紋理被提取時，基本上可預期頁面 410 所浪費的程度由於此處所述之該基礎紋理資料結構的好處而降低。另外，當要對於紋理 2 執行著色時，對應於頁面 410 的一個項目可能仍駐在於該 TLB 中。因此，即可避免將需要提取區塊 421-415 之該位址轉譯。

【0042】 在圖 4 的具體實施例中，區塊 411-415 與區塊 421-425 之每一者係包含在一致(uniform)(例如 4 kB)的區塊中。它們彼此交錯(interleaved)，因此區塊 421 儲存在區塊 411 與區塊 422 之間等等。另外，在圖 4 的具體

實施例中，411 的位址亦為頁面 410 的位址。因此，適當長度的 MSB 被預先加入到一基礎位址中(例如在 4 kB 區塊大小的案例中為 12 位元)。例如，如果每一複合紋理具有 8 個相對應的基礎紋理，適當地是有三個 MSB 來定址包含該等 8 個相對應基礎紋理的該等區塊。依此方式定址該等基礎紋理區塊並非必須，但可需要較少的位址轉譯。

【0043】 圖 4 亦顯示有對應於一複合紋理 N 的區塊 431-435 之一頁面 420。在圖 4 的具體實施例中，區塊 431-435 被包含在 4 kB 的區塊中。區塊 431-435 被交錯，使得區塊 432 儲存在區塊 431 與 433 之間等等。另外，在圖 4 的具體實施例中，區塊 431 的位址自頁面 420 的該位址偏置一 2 的次方之倍數。因此，適當長度的 MSB 被預先加入到一 12 位元的基礎位址。

【0044】 在該等前述的具體實施例中，該等基礎紋理之每一者對應於一單一複合紋理。但是，並不一定非要為此不可。在另一具體實施例中，一特定基礎紋理對應於並用於產生多個複合紋理。例如，包含在區塊 431-435 中一或多個該等基礎紋理對應於例如四個複合紋理。因此此具體實施例利用較少的基礎紋理來建構相同數目的複合紋理，其需要較少的記憶體來儲存該等基礎紋理，並可減少執行關於一給定數目的複合紋理之著色作業所需要的位址轉譯或 TLB 查找之數目。

【0045】 同時在該等前述的具體實施例中，該等基礎紋理假設是為相同的大小。但是，亦不一定非要為此不可。在另一具體實施例中，該等基礎紋理具有不同的大小。例如，一些基礎紋理可以覆蓋一 4 x 8 的磁磚，而其它的則可覆蓋一 2 x 4 的磁磚。同樣地，該等複合紋理可具有不同的大小。另外，該等基本紋理的大小不需要相等於它們相對應之複合紋理的大小。

【0046】 因為基礎紋理係參照為具有一高度的空間本地性(space locality)，熟此技藝者將可瞭解到在 GPU 支援多重執行緒執行的具體實施例中有可能可以橫跨多個執行緒來同時地使用鄰接的基礎紋理。因此較佳地是基礎紋理被相鄰地儲存。但是，當使用一給定頁面內多個基礎紋理來產

生一複合紋理時，該等基礎紋理較佳地是在該頁面之內粗略地交錯，而非微細地交錯。此可稱之為「陣列結構的陣列」(array of structures of arrays)。該「結構的陣列」為該複合紋理，該結構為該記憶體頁面，而「陣列的結構」為該等個別的基礎紋理。

【0047】 圖 5 為一種著色方法的一具體實施例之流程圖。該方法開始於步驟 510。在步驟 520，對應於一特定複合紋理的多個基礎紋理被儲存在一單一頁面。在步驟 530，至少一著色器被初始化來執行關於一特定複合紋理的至少一著色作業。在某個時間點，該著色器需要能夠存取到對應於該特定複合紋理的至少一些基礎紋理。因此在步驟 540，該著色器啟動該等基礎紋理的提取，造成對應於該單一頁面的一位址轉譯。在另一具體實施例中，代替該位址轉譯，一個項目可由一 TLB 取得。在步驟 550，多個基礎紋理自該單一頁面提取。在一具體實施例中，該頁面包含所有該著色器需要的該等基礎紋理，釋放該著色器使其不需要啓始其它頁面的任何提取。在步驟 560，利用至少一偏置量(offset)來定址包含在該頁面中的該等多個基礎紋理中至少一者。然後在步驟 570 該著色器利用該等多個基礎紋理來執行該著色作業。該方法結束於一結束步驟 580。

【0048】 與本申請案相關的技術專業人士將可瞭解到可能對該等描述的具體實施例進行其它與進一步的加入、刪除、取代和修改。在不脫離本發明精神或必要特性的情況下，可以其他特定形式來體現本發明。應將所述具體實施例各方面僅視為解說性而非限制性。因此，本發明的範疇如隨附申請專利範圍所示而非如前述說明所示。所有落在申請專利範圍之等效意義及範圍內的變更應視為落在申請專利範圍的範疇內。

【符號說明】

100	電腦系統
102	中央處理單元

104	系統記憶體
106	圖形處理子系統
108	輸入裝置
110	顯示裝置
112	應用程式
114	應用程式化介面
116	圖形處理器驅動器
118	圖形處理單元
120	圖形處理器本地記憶體
122	晶片上圖形處理器記憶體
124	串流多重處理器
126	像框緩衝器
128	圖形處理器程式碼
130	晶片上緩衝器
132	資料匯流排系統
134	圖形處理器資料匯流排
136	晶片上圖形處理器資料匯流排
200	圖形管線
202	資料組合器
204	頂點著色引擎
206	基元組合器
208	幾何著色引擎
210	掃描場解析器
212	片段著色引擎
214	掃描場作業單元
300	幾何著色器群組

302	幾何著色器
304	輸入三角形
306	輸出三角形
308	工作識別
310	輸出限制
410	實體頁面
411-415	複合紋理，基礎紋理
421-425	複合紋理，基礎紋理
430	實體頁面
431-435	複合紋理，基礎紋理
510, 520, 530, 540, 550, 560, 570, 580	步驟

【生物材料寄存】

國內寄存資訊【請依寄存機構、日期、號碼順序註記】

國外寄存資訊【請依寄存國家、機構、日期、號碼順序註記】

【序列表】 (請換頁單獨記載)

申請專利範圍

1. 一種圖形處理子系統，其包含：
 - 一記憶體，其設置成包含一紋理資料結構，該紋理資料結構係根據對應於一特定複合紋理的多個基礎紋理中那些係包含在該記憶體的一單一頁面中；及
 - 一圖形處理器，其設置成經由一資料匯流排與該記憶體進行通訊，並執行一著色器以提取包含在該單一頁面中的該等多個基礎紋理，藉此產生該特定複合紋理。
2. 如申請專利範圍第 1 項之子系統，其中該單一頁面包含該著色器要產生該特定複合紋理所需要的所有基礎紋理。
3. 如申請專利範圍第 1 項之子系統，其中該圖形處理器設置成利用偏置量來針對該等多個基礎紋理中至少部份來建構虛擬位址。
4. 如申請專利範圍第 1 項之子系統，其中該等多個基礎紋理係包含在該單一頁面內一致的區塊中。
5. 如申請專利範圍第 1 項之子系統，其中該單一頁面包含對應於多個複合紋理的多個基礎紋理。

6. 如申請專利範圍第 5 項之子系統，其中對應於該等多個複合紋理的該等多個基礎紋理被交錯。
7. 如申請專利範圍第 1 項之子系統，其中該圖形處理器設置成執行一複數個著色器。
8. 一種著色方法，該方法包含：
 - 初始化一著色器來對於一特定複合紋理執行一著色作業；
 - 啓動多個基礎紋理的提取，該啓動造成對應於包含該等多個基礎紋理的一單一頁面之一位址轉譯；
 - 自該單一頁面提取該等多個基礎紋理；及
 - 利用該等多個基礎紋理來執行該著色作業。
9. 如申請專利範圍第 8 項之方法，其中該頁面包含該著色器要執行該著色作業所需要的所有基礎紋理。
10. 如申請專利範圍第 8 項之方法，另包含利用偏置量來針對該等多個基礎紋理中至少部份來建構虛擬位址。

圖式

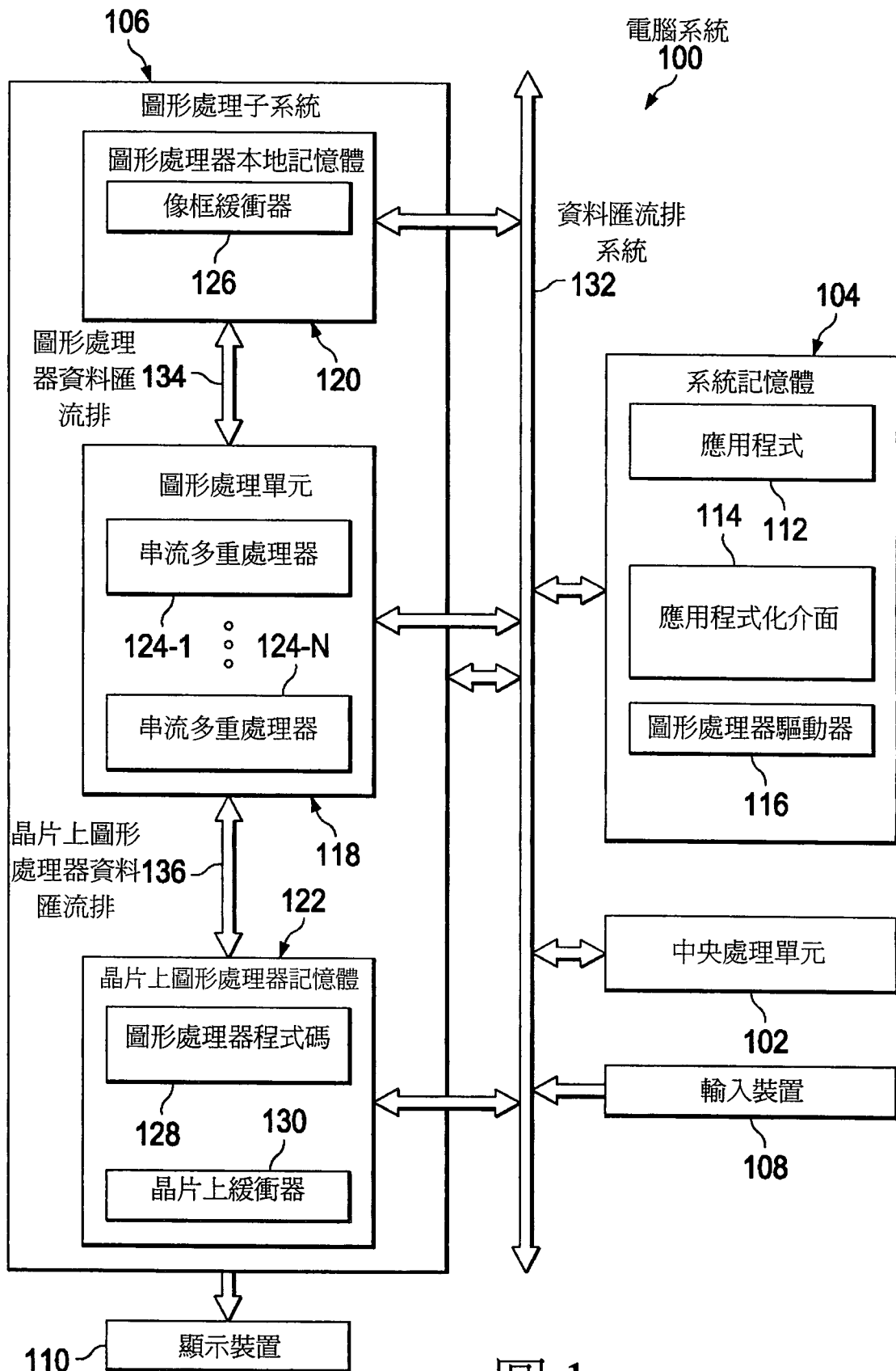


圖 1

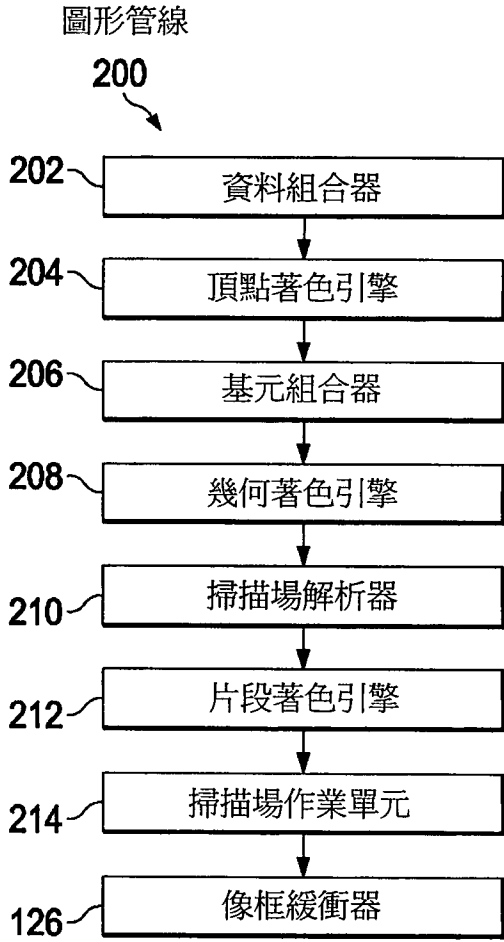


圖 2

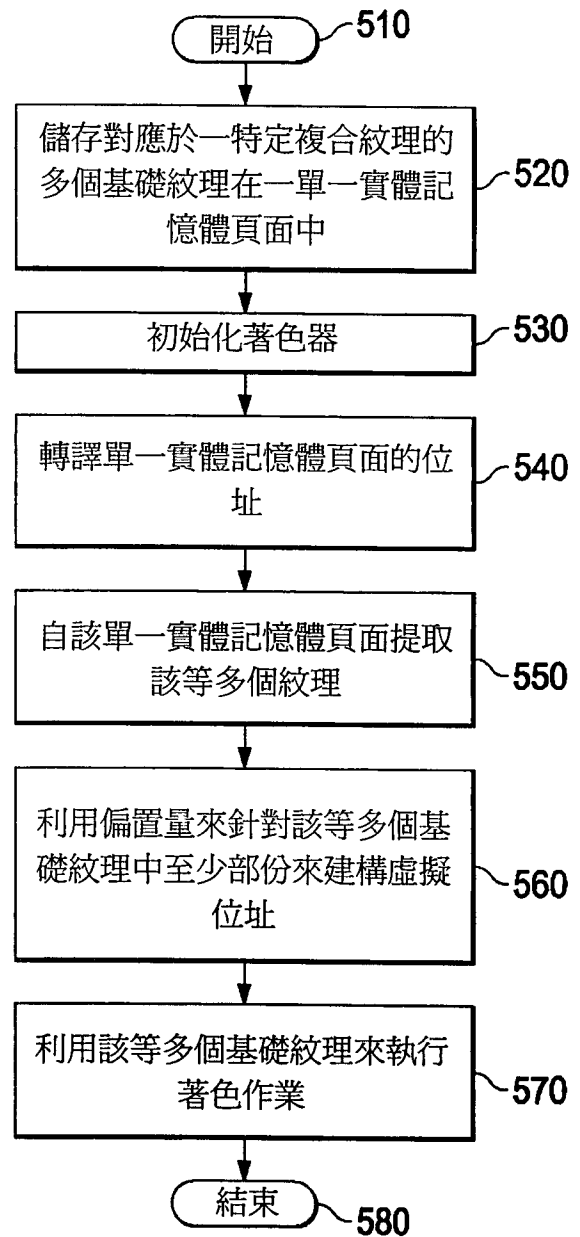


圖 5

幾何著色引擎
208

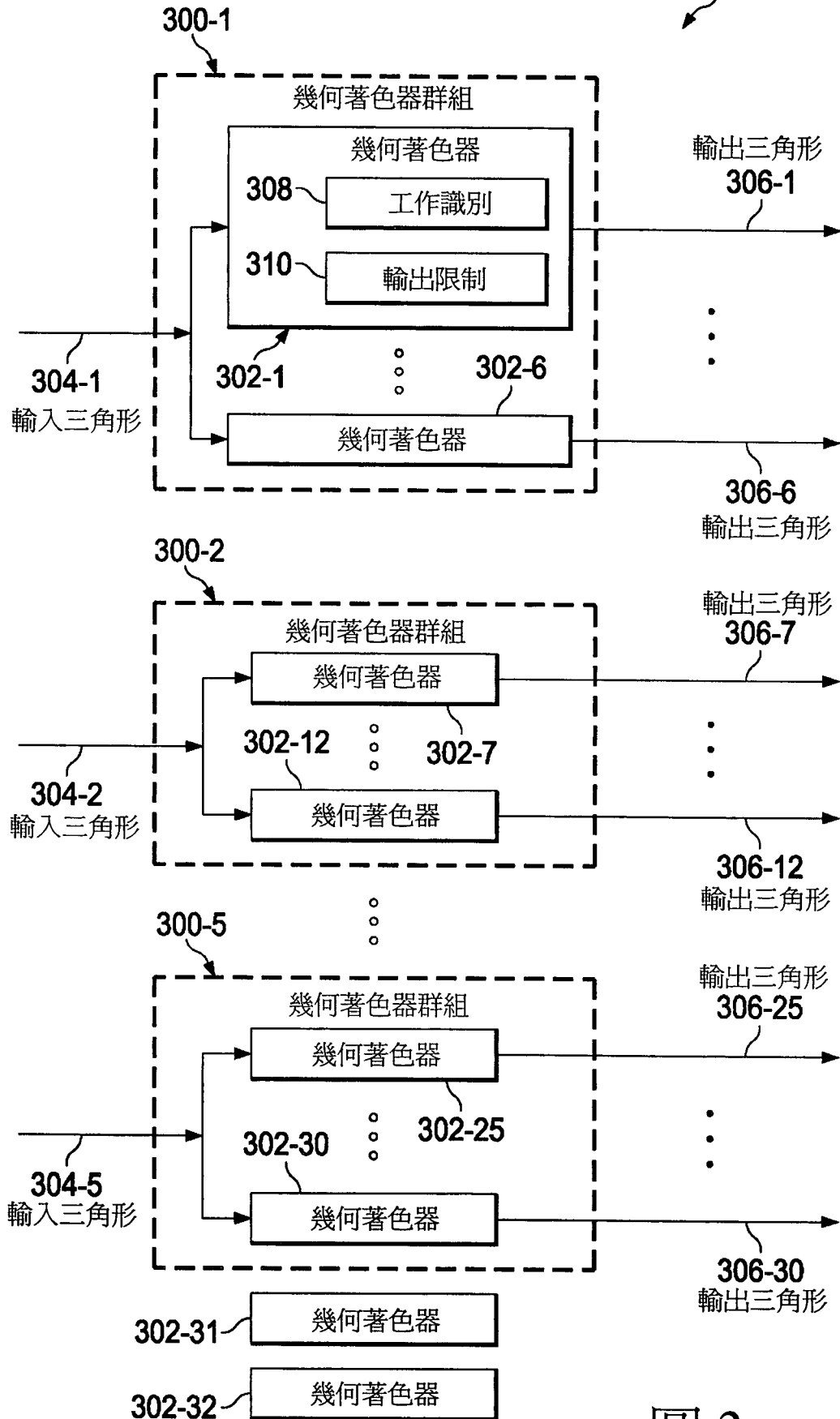


圖 3

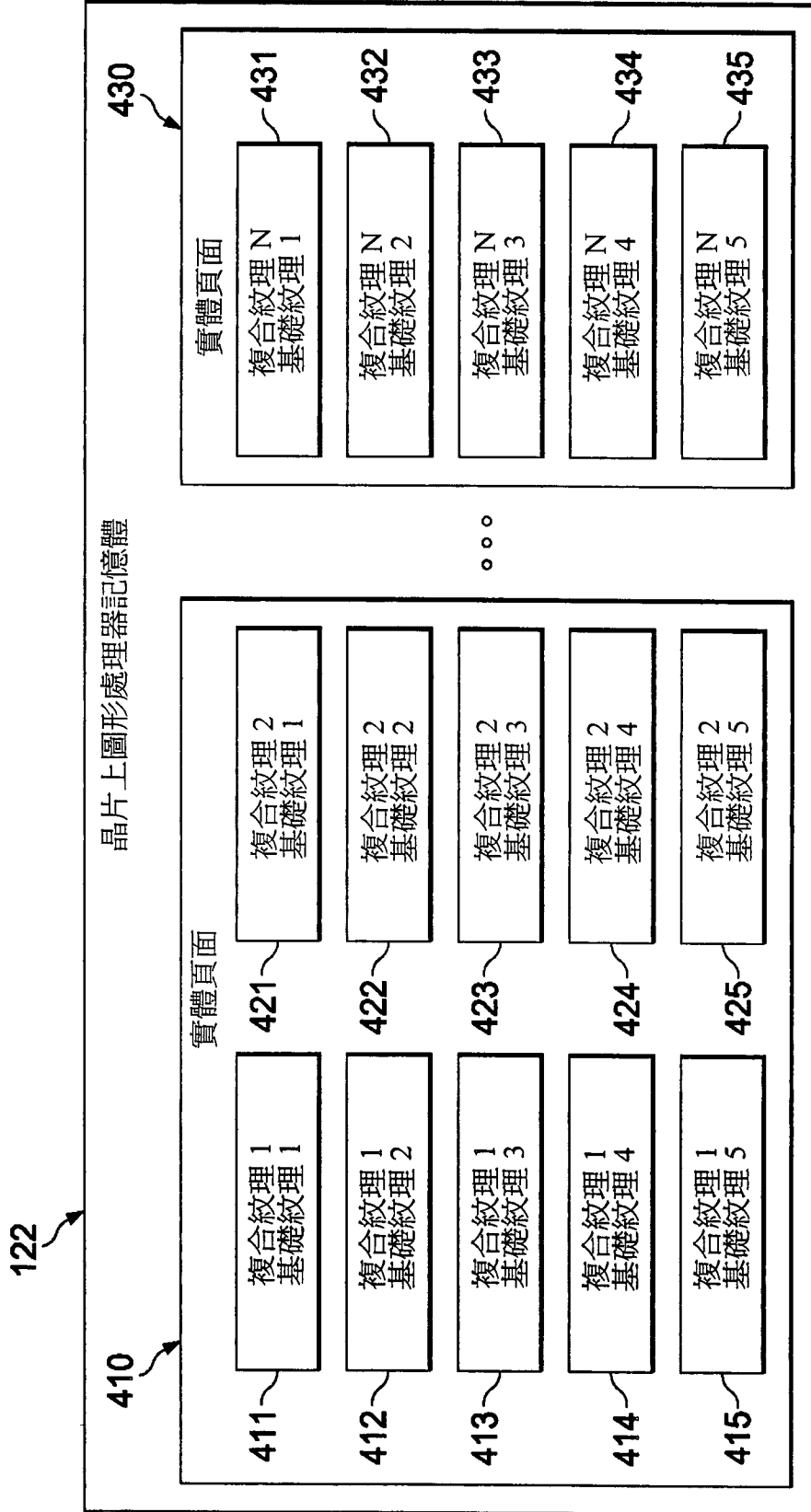


圖 4