

發明專利說明書

(本說明書格式、順序及粗體字，請勿任意更動，※記號部分請勿填寫)

※ 申請案號：97116868

※ 申請日期：97.5.7

※IPC 分類：~~G06T~~

G06T 1/50 (2006.01)

一、發明名稱：(中文/英文)

描繪後圖形重疊

POST-RENDER GRAPHICS OVERLAYS

二、申請人：(共 1 人)

姓名或名稱：(中文/英文)

美商高通公司

QUALCOMM INCORPORATED

代表人：(中文/英文)

湯瑪仕 R 勞斯

ROUSE, THOMAS R.

住居所或營業所地址：(中文/英文)

美國加州聖地牙哥市摩豪斯大道5775號

5775 MOREHOUSE DRIVE SAN DIEGO, CA 92121-1714, U.S.A.

國 籍：(中文/英文)

美國 U.S.A.

三、發明人：(共 4 人)

姓 名：(中文/英文)

1. 史帝芬 泰德 威布魯
WEYBREW, STEVEN TODD
2. 布萊恩 亞利斯
ELLIS, BRIAN
3. 巴伯克 艾密斯
ELMIEH, BABACK
4. 賽門 威爾森
WILSON, SIMON

國 籍：(中文/英文)

1. 美國 U.S.A.
2. 美國 U.S.A.
3. 加拿大 CANADA
4. 英國 U.K.

四、聲明事項：

☐ 主張專利法第二十二條第二項 ☐ 第一款或 ☐ 第二款規定之事實，其事實發生日期為： 年 月 日。

☒ 申請前已向下列國家（地區）申請專利：

【格式請依：受理國家（地區）、申請日、申請案號 順序註記】

☒ 有主張專利法第二十七條第一項國際優先權：

1. 美國；2007年05月07日；60/916,303

2. 美國；2008年05月06日；12/116,056

☐ 無主張專利法第二十七條第一項國際優先權：

1.

2.

☐ 主張專利法第二十九條第一項國內優先權：

【格式請依：申請日、申請案號 順序註記】

☐ 主張專利法第三十條生物材料：

☐ 須寄存生物材料者：

國內生物材料 【格式請依：寄存機構、日期、號碼 順序註記】

國外生物材料 【格式請依：寄存國家、機構、日期、號碼 順序註記】

☐ 不須寄存生物材料者：

所屬技術領域中具有通常知識者易於獲得時，不須寄存。

九、發明說明：

【發明所屬之技術領域】

本揭示案係關於圖形處理，且更特定言之，係關於在描繪過程之後圖形表面的重疊。

本申請案主張2007年5月7日申請之美國臨時申請案第60/916,303號之權利，該案之全文以引用的方式併入本文中。

【先前技術】

圖形處理器被廣泛用來描繪用於諸如視訊遊戲、圖形程式、電腦輔助設計(CAD)應用程式、模擬及可視化工具及成像之各種應用的二維(2D)及三維(3D)影像。接著可使用顯示處理器顯示經描繪之輸出。

用於此等應用中之圖形處理器、顯示處理器或多媒體處理器可經組態以執行對資料之並列及/或向量處理。具有或不具有SIMD(單指令，多資料)擴展之通用CPU(中央處理單元)亦可經組態以處理資料。在SIMD向量處理中，單指令同時對多個資料項操作。

OpenGL[®](開放圖形程式庫)為界定可在寫入產生2D及3D圖形之應用程式時使用的跨平台API(應用程式設計介面)之標準規格。諸如Java之其他語言可經由其本身的標準過程來界定與OpenGL API之結合。API包括當實施於圖形應用程式中時可用於根據簡單圖元繪製景物的多個函數。圖形處理器、多媒體處理器及甚至通用CPU可執行使用OpenGL函數調用而被寫入之應用程式。OpenGL ES(嵌入式

系統)為OpenGL之變體，其經設計用於諸如行動電話、PDA或視訊遊戲機之嵌入式器件。OpenVG™(開放向量圖形)為另一標準API，其主要設計用於硬體加速2D向量圖形。

EGL™(嵌入式圖形程式庫)為介於描繪API(諸如，OpenGL ES、Open VG及若干其他標準多媒體API)與下伏之平台多媒體設施之間的平台介面層。EGL可處理圖形內容管理、描繪表面產生及描繪同步且啟用高效能、硬體加速，及2D與3D混合模式描繪。

對於描繪表面產生，EGL提供用於產生螢幕上表面(例如，窗口表面)與螢幕外表面(例如，pbuffer、pixmap)的機制，用戶端API可在該等表面上繪製且用戶端API可共用該等表面。螢幕上表面通常被直接描繪至活動窗口之圖框緩衝器記憶體中。螢幕外表面通常經描繪至螢幕外緩衝器中以供稍後使用。Pbuffer為螢幕外記憶體緩衝器，其可儲存(例如)於與OpenGL伺服器側(驅動程式)操作相關聯之記憶體空間中。Pixmap為螢幕外記憶體區域，其通常儲存(例如)於與用戶端應用程式相關聯之記憶體空間中。

【發明內容】

一般而言，本揭示案描述用於將一組經描繪或預先描繪之圖形表面重疊或組合至單一圖形圖框上的各種技術。在一態樣中，器件包括第一處理器，其在器件將複數個經描繪之圖形表面中之任一者輸出至顯示器之前選擇用於該等經描繪之圖形表面中之每一者的表面等級。該器件進一步包括第二處理器，其檢索經描繪之圖形表面，根據選定之

表面等級中之每一者將經描繪之圖形表面重疊至圖形圖框上，且將所得的圖形圖框輸出至顯示器。

在另一態樣中，方法包括檢索複數個經描繪之圖形表面。該方法進一步包括在將經描繪之圖形表面中之任一者輸出至顯示器之前選擇用於經描繪之圖形表面中之每一者的表面等級。該方法進一步包括根據選定之表面等級中之每一者將經描繪之圖形表面重疊至圖形圖框上。該方法進一步包括將所得的圖形圖框輸出至顯示器。

在另一態樣中，電腦可讀媒體包括用於造成一或多個可程式處理器執行以下操作之指令：檢索複數個經描繪之圖形表面，在將經描繪之圖形表面中之任一者輸出至顯示器之前選擇用於經描繪之圖形表面中之每一者的表面等級，根據選定之表面等級中之每一者將經描繪之圖形表面重疊至圖形圖框上，及將所得的圖形圖框輸出至顯示器。

在隨附圖式及以下描述中闡述本揭示案之一或多個態樣的細節。其他特徵、目標及優點將自該描述及諸圖式及自申請專利範圍而顯而易見。

【實施方式】

一般而言，本揭示案描述用於將一組經描繪或預先描繪之圖形表面重疊或組合至單一圖形圖框上的各種技術。圖形表面可為二維(2D)表面、三維(3D)表面，及/或視訊表面。2D表面可藉由實施2D API(諸如，OpenVG)之函數之軟體或硬體來產生。3D表面可藉由實施3D API(諸如，OpenGL ES)之函數之軟體或硬體來產生。視訊表面可藉由視訊解碼器

來產生，諸如符合ITU H.264或MPEG4(動畫專家群版本4)之視訊解碼器。經描繪之圖形表面可為螢幕上表面(諸如，窗口表面)或螢幕外表面(諸如，pbuffer表面或pixmap表面)。此等表面中之每一者可被顯示為靜態影像或一組活動影像(諸如，視訊或合成動畫)之一部分。在本揭示案中，預先描繪之圖形表面可指代(1)藉由應用程式而描繪及保存至影像檔案且隨後可藉由具有或不具有進一步處理之圖形應用而自該影像檔案載入的內容；或(2)作為圖形應用之初始化過程之一部分而不在圖形應用之主要動畫執行時間循環期間藉由圖形應用描繪的影像。此外，經描繪之圖形表面可指代預先描繪之圖形表面或指代界定或包括經描繪之資料的任何種類之資料結構。

在一態樣中，每一圖形表面可具有與表面相關聯之表面等級。該表面等級確定每一圖形表面重疊至圖形圖框上之等級。表面等級在某些狀況下可被界定為任何數字，其中數字愈高，在所顯示圖形圖框上該表面的顯示層級愈高。換言之，具有較高表面等級之表面可能看來似乎較接近顯示器之觀察者。亦即，具有較高表面等級之表面中所含有的物件可能看來似乎在具有較低表面等級之表面中所含有的其他物件之前面。作為簡單實例，用於桌上型電腦上之背景影像或"底色圖案"將具有比桌上之圖示較低的表面等級。

在一態樣中，顯示處理器可根據一或多個複合或摻合模式組合圖形表面。該等複合模式之實例包括(1)覆寫、(2) α

摻合、(3)不具有 α 摻合之色彩鍵控及(4)具有 α 摻合之色彩鍵控。根據兩個表面之部分重疊的覆寫複合模式，可顯示具有較高表面等級之表面的重疊部分而不顯示具有較低表面等級之任何表面的重疊部分。

在某些狀況下，顯示處理器可根據界定複數個層之重疊堆疊(每一層對應於一不同層等級)組合該等表面。每一層可包括結合至該層之一或多個表面。顯示處理器接著可遍曆該重疊堆疊以確定組合該等表面之次序。在一態樣中，使用者程式或API函數可選擇性地啟用或停用重疊堆疊內之個別表面，且接著顯示處理器僅組合已啟用之彼等表面。在另一態樣中，使用者程式或API函數可選擇性地啟用或停用重疊堆疊內之全部層，且接著顯示處理器僅組合結合至經啟用之層的彼等經啟用之表面。

在諸如視訊遊戲之圖形密集型應用中，待重疊之圖形表面中之許多者為藉由2D API之軟體及硬體實施而產生的2D表面(而不是藉由3D硬體產生)。然而，用於描繪3D圖形之許多標準不包括用於將該等2D圖形表面重疊至3D圖形表面上之規格。因而，能夠使2D表面之描繪與3D表面之描繪同步的複雜描繪引擎有時用於該等應用，但該等描繪引擎之複雜性可不利地影響該等系統之總系統成本、圖形效能及功率消耗。

本揭示案中之重疊及複合技術可提供一或多個優點。舉例而言，視訊遊戲可顯示複雜3D圖形以及簡單圖形物件(諸如，2D圖形及相對靜態之物件)。可與複雜3D圖形分開地將

此等簡單圖形物件描繪為螢幕外表面。可將經描繪之螢幕外表面與複雜3D圖形表面重疊(亦即，組合)以產生最終的圖形圖框。因為簡單圖形物件可能不需要3D圖形描繪能力，所以可使用消耗圖形處理系統中之較少硬體資源之技術描繪該等物件。舉例而言，該等物件可藉由使用通用處理管道之處理器來描繪或藉由具有2D圖形加速能力之處理器來描繪。此外，該等物件可經預先描繪及儲存以供以後在圖形處理系統內使用。藉由與複雜3D圖形分開地描繪或預先描繪簡單圖形物件，可減少3D圖形描繪硬體上之負載。此在行動通信領域中尤其重要，其中3D圖形硬體上減少的負載可導致行動器件之功率節省及/或增加的總效能(亦即，較高的持續圖框速率)。

此外，藉由將圖形處理劃分成執行複雜描繪之圖形處理器及組合螢幕上表面與螢幕外表面之顯示處理器，且藉由並列操作此等兩個處理器，可減少圖形處理器之時鐘率。此外，因為功率消耗隨著圖形處理器之時鐘率非線性地增加，所以可在圖形處理系統中達成額外功率節省。

圖1為說明根據本揭示案之態樣之器件100的方塊圖，器件100可用於將一組經描繪之圖形表面重疊或組合至圖形圖框上。器件100可為獨立器件或可為較大系統之一部分。舉例而言，器件100可包含無線通信器件(諸如，無線手機)，或可為數位相機、數位多媒體播放機、個人數位助理(PDA)、視訊遊戲機、行動遊戲器件或其他視訊器件之一部分。在一態樣中，器件100可包含或為個人電腦(PC)或膝上

型器件之一部分。器件100亦可包括於一或多個積體電路或晶片 中。

器件100可能夠執行各種不同應用，諸如圖形應用、視訊應用或其他多媒體應用。舉例而言，器件100可用於圖形應用、視訊遊戲應用、視訊應用、組合視訊與圖形之應用、數位相機應用、即時訊息傳遞應用、行動應用、視訊電話或視訊串流應用。

器件100可能夠處理多種不同資料類型及格式。舉例而言，器件100可處理靜態影響資料、活動影像(視訊)資料或其他多媒體資料，如下文將更詳細描述的。在圖1之實例中，器件100包括圖形處理系統102、記憶體104及顯示器件106。可程式處理器108、110及114可包括於圖形處理系統102內。可程式處理器108可為控制或通用處理器，且可包含系統CPU(中央處理單元)。可程式處理器110可為圖形處理器，且可程式處理器114可為顯示處理器。控制處理器108可能夠控制圖形處理器110與顯示處理器114。處理器108、110及114可為純量或向量處理器。在一態樣中，器件100可包括其他形式之多媒體處理器。

在一態樣中，圖形處理系統102可實施於彼此實體分開之若干不同子系統或組件上。在此種狀況下，可程式處理器108、110、114之一或多者可實施於不同組件上。舉例而言，圖形處理系統102之一實施可包括在第一組件或子系統上之控制處理器108及顯示處理器114，及在第二組件或子系統上之圖形處理器110。

在裝置100中，圖形處理系統102可不僅耦接至記憶體104而且耦接至顯示器件106。記憶體104可包括能夠儲存指令及/或資料之任何永久或揮發性記憶體。顯示器件106可為能夠顯示3D影像資料、2D影像資料或用於顯示目的之視訊資料的任何器件，諸如LCD(液晶顯示器)或標準電視顯示器件。

圖形處理器110可為用以描繪、操縱及顯示電腦化圖形之專用圖形描繪器件。圖形處理器110可實施各種複雜的與圖形有關之演算法。舉例而言，複雜演算法可對應於二維或三維電腦化圖形之表示。圖形處理器110可實施許多所謂的"圖元"圖形操作，諸如形成點、線、三角形或其他多邊形以產生用於呈現於顯示器(諸如，顯示器件106)上之複雜、三維影像。

在本揭示案中，術語"描繪"可指代3D及/或2D描繪。作為實例，圖形處理器110可利用OpenGL指令來描繪3D圖形表面，或可利用OpenVG指令來描繪2D圖形表面。然而，在各種態樣中，用於描繪圖形之任何標準、方法或技術可由圖形處理器110來利用。在一態樣中，控制處理器108亦可利用OpenVG指令描繪2D圖形表面。

圖形處理器110可執行儲存於記憶體104中之指令。記憶體104能夠儲存用於應用程式(諸如，圖形或視訊應用程式)之應用程式指令118、API程式庫120、驅動程式122，及表面資訊124。可將應用程式指令118自記憶體104載入圖形處理系統102中以用於執行。舉例而言，控制處理器108、圖

形處理器 110 及顯示處理器 114 之一或多者可執行指令 118 之一或多者。

控制處理器 108、圖形處理器 110 及/或顯示處理器 114 亦可在應用程式指令 118 之執行期間載入及執行 API 程式庫 120 或驅動程式 122 內所含有的指令。指令 118 可參考或以其它方式調用 API 程式庫 120 或驅動程式 122 內之某些函數。因此，當圖形處理系統 102 執行指令 118 時，其亦可執行 API 程式庫 120 及/或驅動程式 122 內經識別之指令，如下文將更詳細描述的。驅動程式 122 可包括為控制處理器 108、圖形處理器 110 及顯示處理器 114 中之一或多者所特有的功能性。在一態樣中，可將應用程式指令 118、API 程式庫 120 及/或驅動程式 122 自儲存器件(諸如，非揮發性資料儲存媒體)載入記憶體 104 中。

圖形處理系統 102 亦包括表面緩衝器 112。圖形處理器 110、控制處理器 108 及顯示處理器 114 各自可操作性地耦接至表面緩衝器 112，以使得此等處理器中之每一者可自表面緩衝器 112 讀取資料或將資料寫入表面緩衝器 112 中。表面緩衝器 112 亦可操作性地耦接至圖框緩衝器 160。雖然展示為包括於圖 1 中之圖形處理系統 102 內，但表面緩衝器 112 及圖框緩衝器 160 在某些態樣中亦可直接包括於記憶體 104 內。

表面緩衝器 112 可為能夠儲存資料之任何永久或揮發性記憶體，諸如同步動態隨機存取記憶體(SDRAM)、嵌入式動態隨機存取記憶體(eDRAM)，或靜態隨機存取記憶體

(SRAM)。當圖形處理器110描繪圖形表面(諸如，螢幕上表面或螢幕外表面)時，圖形處理器110可將該描繪資料儲存於表面緩衝器112中。經描繪之每一圖形表面可藉由其尺寸及形狀來界定。尺寸及形狀可不受所使用之顯示器件106之實際實體尺寸限制，因為顯示處理器114可將描繪後縮放及旋轉函數應用於經描繪之表面。

表面緩衝器112可包括一或多個經描繪之圖形表面116A-116N(共同地，116)，及一或多個表面等級117A-117N(共同地，117)。116中的每一經描繪之表面可含有經描繪之表面資料，其包括尺寸資料、形狀資料、像素色彩資料及可能在表面描繪期間產生的其他描繪資料。116中的每一經描繪之表面亦可具有與經描繪之表面116相關聯之表面等級117。每一表面等級117界定116中的對應的經描繪之表面重疊或下伏至所得的圖形圖框上之等級。雖然圖1中將表面緩衝器112展示為單一表面緩衝器，但表面緩衝器112可包含各自儲存一或多個經描繪之表面116A-116N之一或多個表面緩衝器。

經描繪之表面(諸如，表面116A)可為螢幕上表面(諸如，窗口表面)或螢幕外表面(諸如，pbuffer表面或pixmap表面)。窗口表面及pixmap表面可聯繫至本地開視窗系統內之對應窗口及pixmap。Pixmap可用於螢幕外描繪至可經由本地API存取之緩衝器中。在一態樣中，用戶端應用程式可藉由調用與平台介面層相關聯之函數(諸如，EGL API之實例化)而產生初始表面。在產生初始表面之後，用戶端應用程

式可使描繪背景(亦即,狀態機)與每一初始表面相關聯。可藉由跨平台API(諸如,OpenGL ES)之實例化產生描繪背景。接著,用戶端應用程式可將資料描繪至初始表面中以產生經描繪之表面。用戶端應用程式可藉由導致可程式處理器(諸如,控制處理器108或圖形處理器110)產生經描繪之表面而將資料描繪至初始表面中。

經描繪之表面116可源自器件100內之若干不同來源。舉例而言,圖形處理器110及控制處理器108可各自產生一或多個經描繪之表面,且接著將經描繪之表面儲存於表面緩衝器112中。圖形處理器110可回應於自控制處理器108所接收之指令而藉由使用加速3D圖形描繪管道來產生經描繪之表面。控制處理器108可藉由使用通用處理管道(其不被加速用於圖形描繪)產生經描繪之表面。控制處理器108亦可檢索儲存於記憶體104之各種部分內的經描繪之表面,諸如儲存於記憶體104之表面資訊124內的經描繪或預先描繪之表面。因此,經描繪之表面116A-116N中之每一者可源自器件100內之相同或不同來源。

在一態樣中,由圖形處理器110產生的經描繪之表面可為螢幕上表面,且由控制處理器108產生或檢索的經描繪之表面可為螢幕外表面。在另一態樣中,圖形處理器110可產生螢幕上表面以及螢幕外表面。由圖形處理器110產生的螢幕外表面可在圖形處理器110未經利用時(亦即,具有相對大量的可用資源)之指定時間產生,且接著儲存於表面緩衝器112中。接著,在圖形處理器110可能經過度利用時(亦即,

具有相對小量的可用資源)之時間，顯示處理器114可將預先產生的圖形表面重疊至圖形圖框上。可藉由預先描繪器件100內之某些圖形表面而改良圖形處理器110之平均處理量，此可導致圖形處理系統102之總功率節省。

顯示處理器114能夠自表面緩衝器112中檢索經描繪之表面116，將經描繪之圖形表面重疊至圖形圖框上，且驅動顯示器件106顯示所得的圖形圖框。每一圖形表面116重疊之等級可藉由界定用於該圖形表面之對應表面等級117來確定。表面等級117A-117N可藉由使用者程式(諸如，藉由應用程式指令118)來界定，且經儲存作為與經描繪之表面相關聯之參數。表面等級可儲存於表面緩衝器112中或記憶體104之表面資訊124區塊中。

表面等級117A-117N可各自經界定為任何數字，其中數字愈高，在顯示圖形圖框上經顯示表面的層級愈高。舉例而言，具有較高表面等級之表面可能看來似乎較接近顯示器之觀察者。亦即，具有較高表面等級之表面中所含有的物件可能看來似乎在具有較低表面等級之表面中所含有的物件之前面。

在一態樣中，顯示處理器114可根據一或多個複合或摻合模式組合圖形表面，諸如：(1)覆寫，(2) α 摻合，(3)不具有 α 摻合之色彩鍵控，及(4)具有 α 摻合之色彩鍵控。根據兩個表面之部分重疊的覆寫複合模式，可顯示具有較高表面等級之表面的重疊部分而不顯示具有較低表面等級之任何表面的重疊部分。作為簡單實例，用於桌上型電腦上之背

景影像將具有比桌上之圖示低的表面等級。在某些狀況下，顯示處理器114可使用正交投影或其他透視投影組合經描繪之圖形表面。 α 摻合複合模式可根據(例如)全表面恆定 α 摻合演算法或根據全表面每像素 α 摻合演算法執行 α 摻合。

根據具有 α 摻合複合模式之色彩鍵控，當兩個表面重疊時，顯示處理器114確定哪一表面具有較高表面等級及哪一表面具有較低表面等級。顯示處理器114接著可檢查較高表面之重疊部分內之每像素以確定哪些像素匹配基色(例如，洋紅色)。對於匹配基色之任何像素，將選擇來自較低表面之對應像素(亦即，具有相同顯示位置之像素)作為輸出像素(亦即，經顯示之像素)。對於不匹配基色之任何像素，將根據 α 摻合演算法將較高表面之像素以及來自較低表面之對應像素摻合在一起以產生輸出像素。

當選定的複合模式為不具有 α 摻合之色彩鍵控時，顯示處理器114可檢查較高表面之重疊部分內之每像素以確定哪些像素匹配基色。對於匹配基色之任何像素，將選擇來自較低表面之對應像素(亦即，具有相同顯示位置之像素)作為輸出像素。對於不匹配基色之任何像素，選擇來自較高表面之像素作為輸出像素。

在任何狀況下，顯示處理器114根據選定的複合模式組合該等層以產生可載入圖框緩衝器160中之所得的圖形圖框。顯示處理器114亦可對經描繪之圖形表面或圖框執行其他描繪後函數(包括縮放及旋轉)。在一或多個EGL擴展中，縮放及旋轉函數可為指定的。

在某些態樣中，控制處理器108可為RISC處理器，諸如嵌入由San Diego, CA之Qualcomm有限公司設計之行動台數據機中的ARM₁₁處理器。在某些態樣中，顯示處理器114可為亦嵌入由Qualcomm有限公司設計之行動台數據機中的行動顯示處理器(MDP)。處理器108、110及114中之任一者能夠存取緩衝器空間112內的經描繪之表面116A-116N。在一態樣中，每一處理器108、110及114可能夠提供描繪能力且將用於圖形表面之經描繪之輸出資料寫入表面緩衝器112中。

記憶體104亦包括表面資訊124，其儲存與圖形處理系統102內產生的經描繪之圖形表面有關的資訊。舉例而言，表面資訊124可包括表面概況。表面概況可包括經描繪之表面資料、表面等級資料及為每一表面所特有的其他參數。表面資訊124亦可包括與複合表面、重疊堆疊及層有關之資訊。重疊堆疊可儲存與應如何將結合至重疊堆疊之表面組合成所得的圖形圖框有關的資訊。舉例而言，重疊堆疊可儲存待關於窗口表面重疊及下伏之一序列層。每一層可具有結合至該層之一或多個表面。表面資訊124可經載入圖形處理系統102之表面緩衝器112或其他緩衝器(未圖示)中以供可程式處理器108、110、114使用。亦可往回提供表面緩衝器112內經更新之資訊以用於儲存於記憶體104之表面資訊124內。

圖2為說明用於儲存於器件100內的經描繪之表面的實例表面概況200之方塊圖。表面概況200可藉由可程式處理器

108、110、114之一者以及藉由應用程式指令118中所含有的使用者程式或藉由API程式庫120中所含有的API函數來產生。一旦產生表面概況200，就可將表面概況200儲存於記憶體104之表面資訊區塊124、表面緩衝器112中或圖形處理系統102內之其他緩衝器(未圖示)內。表面概況200可包括經描繪之表面資料202、啟用旗標204、表面等級資料206及複合表面資訊208。經描繪之表面資料202可包括尺寸資料、形狀資料、像素色彩資料及/或可在表面描繪期間產生的其他描繪資料。

啟用旗標204指示對應於表面概況之表面是被啟用還是停用。在一態樣中，若設定用於表面之啟用旗標204，則啟用表面，且顯示處理器114將表面重疊至所得的圖形圖框上。否則，若未設定啟用旗標204，則停用表面，且顯示處理器114不將表面重疊至所得的圖形圖框上。在另一態樣中，若設定用於表面之啟用旗標204，則僅在亦設定用於表面結合至之重疊堆疊之重疊堆疊啟用旗標(圖3A-304)時，顯示處理器114才可將表面重疊至所得的圖形圖框上。否則，若既未設定重疊堆疊啟用旗標亦未設定表面啟用旗標204，則顯示處理器不將表面重疊至所得的圖形圖框上。在另一態樣中，若設定用於表面之啟用旗標204，則僅在設定用於表面結合至之重疊堆疊及層之重疊堆疊啟用旗標(圖3B-324)與層啟用旗標(圖3C-344)時，顯示處理器114可將表面重疊至所得的圖形圖框上。否則，若未設定啟用旗標中之任一者，則顯示處理器114不將表面重疊至所得的圖形圖

框上。

表面等級資料206確定由顯示處理器114將每一圖形表面116重疊至所得的圖形圖框上的等級。表面等級資料206可藉由使用者程式(諸如，藉由應用程式指令118)來界定。複合表面資訊208可儲存識別與特定表面概況相關聯之複合表面或重疊堆疊的資訊。

在一態樣中，可程式處理器108、110、114可將儲存於記憶體104中之表面概況之各種部分上載至表面緩衝器112中。舉例而言，控制處理器108可自表面概況200上載經描繪之表面資料202及表面等級資料206，且將經上載之資料儲存於表面緩衝器112中。

圖3A為說明根據一態樣之可與複合表面相關聯之實例重疊堆疊300的方塊圖。重疊堆疊300可藉由可程式處理器108、110、114之一者以及藉由應用程式指令118中所含有的使用者程式或藉由API程式庫120中所含有的API函數來產生。一旦產生重疊堆疊，就可將重疊堆疊儲存於記憶體104之表面資訊區塊124、表面緩衝器112中或圖形處理系統102內之其他緩衝器(未圖示)內。

重疊堆疊300包括窗口表面302、啟用旗標304、經描繪之表面資料306A-306N(共同地，306)及表面等級資料308A-308N(共同地，308)。窗口表面302可為與重疊堆疊中之基礎層(例如，層零)相關聯之螢幕上描繪表面。窗口表面302之尺寸及形狀可與儲存於圖框緩衝器160中且隨後顯示於顯示器件106上之所得的圖形圖框之尺寸及形狀相同。經

描繪之表面資料306及表面等級資料308可對應於儲存於記憶體104之表面資訊區塊124中及/或圖形處理系統102之表面緩衝器112內的經描繪之表面資料及表面等級資料。在某些狀況下，重疊堆疊300可包括用於經描繪之表面的全部表面概況200，而在其他狀況下，重疊堆疊300可含有指向含有用於重疊堆疊300之處理之表面資訊的其他資料結構之位址指標。

在一態樣中，可給窗口表面302指派一為零之表面等級(界定為基礎表面等級)。具有正表面等級之表面可重疊窗口表面302，且在本文中被稱作重疊表面。具有負表面等級之表面下伏於窗口表面302且在本文中被稱作下伏表面。重疊表面可具有正表面等級，其中表面等級愈正，該表面看來似乎愈接近顯示器件106之觀察者。舉例而言，具有較高層等級之層中所含有的物件可能看來似乎在具有較低層等級之層中所含有的物件之前面。同樣地，下伏表面可具有負表面等級，其中表面等級愈負，該表面看來似乎距顯示器件106之觀察者愈遠。舉例而言，具有較低層等級之層中所含有的物件可能看來似乎在具有較高層等級之層中所含有的物件之後面或背面。在某些狀況下，可能需要重疊堆疊300具有至少一重疊表面或一下伏表面。使用者應用程式可查詢重疊堆疊300以確定支援多少層及當前多少個表面結合至每一層。重疊堆疊300亦可具有與堆疊相關聯之複合表面。複合表面可為唯讀的(自使用者程式之觀點看)且在必要時由其他API或顯示處理器114使用作為複合緩衝器。或

者，當在不使用專用複合緩衝器或表面之情況下將資料發送至顯示器件106時，重疊堆疊可為複合的"在空中(on-the-fly)"。

啟用旗標304確定重疊堆疊300是被啟用還是停用。若設定用於重疊堆疊300之啟用旗標304，則啟用重疊堆疊，且顯示處理器114可將重疊堆疊300內的所有經啟用之表面組合或處理成所得的圖形圖框。否則，若未設定啟用旗標304，則停用重疊堆疊304，且當產生所得的圖形圖框時，顯示處理器114可不使用重疊堆疊或處理任何相關聯之表面元素。根據一態樣，當停用重疊堆疊300時，仍可啟用窗口表面302且可停用所有其他重疊及下伏表面。因此，根據此態樣，可不停用窗口表面302。在其他態樣中，可停用全部重疊堆疊(包括窗口表面302)。

當組合該等表面時，顯示處理器114可參考重疊堆疊300來確定重疊、下伏或另外組合經描繪之表面306的次序。在一實例中，顯示處理器114可藉由尋找重疊堆疊300中具有最低表面等級之表面而識別看來似乎距顯示器之觀察者最遠的第一表面。接著，顯示處理器114可識別重疊堆疊300中具有第二最低表面等級(其大於第一表面之表面等級)之第二表面。顯示處理器114接著可根據複合模式組合第一表面與第二表面。顯示處理器114可繼續自後至前地重疊表面，以具有最高表面等級且看來似乎最接近觀察者之表面結束。以此方式，顯示處理器114可遍曆重疊堆疊300以循序地組合經描繪之表面306且產生所得的圖形圖框。

在某些狀況下，顯示處理器114可檢查每一表面以查看是否已設定用於表面之啟用旗標204。若設定了啟用旗標(亦即，啟用表面)，則顯示處理器114可組合或處理該表面與重疊堆疊300中之其他經啟用之表面。若未設定啟用旗標(亦即，停用表面)，則顯示處理器114可不組合或處理該表面與重疊堆疊300中之其他表面。儲存於重疊堆疊300內的表面之啟用旗標可藉由在可程式處理器108、110、114之一者上執行之使用者程式或API指令來設定或重設。以此方式，使用者程式可選擇性地啟用及停用用於任何特定圖形圖框之表面。

圖3B為說明根據另一態樣之可與複合表面相關聯之實例重疊堆疊320的方塊圖。類似於重疊堆疊300，重疊堆疊320亦可藉由可程式處理器108、110、114之一者以及藉由應用程式指令118中所含有的使用者程式或藉由API程式庫120中所含有的API函數來產生。一旦產生重疊堆疊，就可將重疊堆疊儲存於記憶體104之表面資訊區塊124、表面緩衝器112中或圖形處理系統102內之其他緩衝器(未圖示)內。

重疊堆疊320包括窗口表面322、啟用旗標324、經描繪之重疊層326A-326N(共同地，326)及下伏層328A-328N(共同地，328)。窗口表面322可為與重疊堆疊中之基礎層(例如，層0)相關聯之螢幕上描繪表面。窗口表面322之尺寸及形狀可與儲存於圖框緩衝器160中且隨後顯示於顯示器件106上之所得的圖形圖框之尺寸及形狀相同。

可給窗口表面322指派一為零之層等級(基礎層)。具有正

層等級之層可重疊窗口表面322，且在本文中被稱作重疊層。具有負層等級之層下伏於窗口表面322且在本文中被稱作下伏層。重疊層可具有正層等級，其中層等級愈正，該層看來似乎愈接近顯示器件106之觀察者。同樣地，下伏層可具有負層等級，其中層等級愈負，該層看來似乎距顯示器件106之觀察者愈遠。每一層可具有藉由執行可程式處理器108、110、114之一者之使用者程式或API指令而結合至該層的多個表面。在某些狀況下，可將層零(亦即，基礎層)限制至單一經描繪之表面，亦即，窗口表面322。在額外狀況下，重疊堆疊320可具有至少一重疊層或一下伏層。使用者應用程式可查詢重疊堆疊320以確定支援多少層及多少個表面可結合至每一層。重疊堆疊320可具有與重疊堆疊相關聯之複合表面。複合表面可為唯讀的(自使用者程式之觀點看)且在必要時由其他API或顯示處理器114使用作為複合緩衝器。或者，當在不使用專用複合緩衝器之情況下將資料發送至顯示器件106時，重疊堆疊可為複合的"在空中"。

啟用旗標324確定重疊堆疊320是被啟用還是停用。若設定了用於重疊堆疊320之啟用旗標324，則啟用重疊堆疊320，且顯示處理器114可將重疊堆疊320內所有經啟用之層組合或處理成所得的圖形圖框。否則，若未設定啟用旗標324，則停用重疊堆疊320，且當產生所得的圖形圖框時，顯示處理器114可不使用重疊堆疊320或處理任何相關聯之表面元素。根據一態樣，當停用重疊堆疊320時，仍可啟用窗口表面322，且可停用重疊層326及下伏層328。因此，根

據此態樣，可不停用窗口表面322。在其他態樣中，可停用全部重疊堆疊(包括窗口表面322)。

當組合該等層時，顯示處理器114可參考重疊堆疊320以確定重疊、下伏或另外組合該等層的次序。在一實例中，顯示處理器114可藉由尋找重疊堆疊320中具有最低表面等級之表面而識別看來似乎距顯示器之觀察者最遠的第一層。接著，顯示處理器114可識別結合至第一層的所有經描繪之表面。顯示處理器114接著可使用複合演算法(可能根據選定的複合模式應用多種像素摻合及鍵控操作之一或多者)組合第一層之經描繪之表面。在某些狀況下，顯示處理器114可檢查以查看是否啟用第一層中之每一表面，且接著僅組合第一層內經啟用之表面。顯示處理器114可使用複合表面臨時儲存第一層之組合表面。接著，顯示處理器114識別重疊堆疊320中具有第二最低層等級之第二表面。第二最低層等級可為仍大於第一層之層等級之最低層等級。顯示處理器114接著可組合結合至第二層之經描繪之表面與先前產生的複合表面。若結合至同一層之兩個表面重疊，則顯示處理器114可根據該等表面結合至層之次序而組合該等表面，如下文進一步詳細描述的。顯示處理器114可繼續自後至前地組合層，以具有最高層等級且看來似乎最接近觀察者之層結束。以此方式，顯示處理器114可遍曆重疊堆疊320以循序地組合層且產生所得的圖形圖框。

在某些狀況下，顯示處理器114可檢查每一層以查看是否已設定用於每一層之啟用旗標(圖3C-344)。若設定啟用旗標

(亦即，啟用層)，則顯示處理器114可組合或處理該層之經啟用之表面與重疊堆疊320中其他經啟用之層的經啟用之表面。若未設定啟用旗標(亦即，停用層)，則顯示處理器114可不組合或處理結合至該層之任何表面與重疊堆疊320中其他層之經啟用之表面。可藉由在可程式處理器108、110、114之一者上執行之使用者程式或API指令來設定或重設重疊堆疊320內之啟用旗標。以此方式，使用者程式可選擇性地啟用及停用用於任何特定圖形圖框之全部層及/或個別表面及/或完整的重疊堆疊。

圖3C為說明可用於重疊堆疊320中之實例層340的方塊圖。層340可為重疊層、下伏層，或基礎層。層340包括層等級資料342、啟用旗標344及經描繪之表面資料346A-346N(共同地，346)。層等級資料342指示層340駐留於重疊堆疊中之層級。在個別結合表面具有表面等級之狀況下，表面等級將與表面所結合至之層的層等級相同。

啟用旗標344確定層340是被啟用還是停用。若設定用於層340之啟用旗標344，則啟用層340，且顯示處理器114可將結合至層340之所有經啟用之表面組合或處理成所得的圖形圖框。否則，若未設定啟用旗標344，則停用層340，且當產生所得的圖形圖框時，顯示處理器114可不使用或處理層340內任何經啟用之表面。在某些狀況下，重疊堆疊320中之窗口表面322可為基礎層之一部分。根據一態樣，可不停用基礎層。在其他態樣中，可停用基礎層。

經描繪之表面資料346可對應於儲存於記憶體104之表面

資訊區塊124中或圖形處理系統102之表面緩衝器112內的經描繪之表面資料。在某些狀況下，層340可包括用於經描繪之表面的全部表面概況200，而在其他狀況下，層340可含有指向含有用於重疊堆疊200之處理之表面資訊的其他資料結構之位址指標。在任何狀況下，在可程式處理器108、110、114之一者上執行之使用者程式可使表面相關聯(亦即，結合)至重疊堆疊320內之特定層。當表面結合至層時，層將含有經描繪之表面資料346或指向用於彼特定表面的適當描繪之表面資料的資訊。

圖4A為描繪重疊堆疊400之實例及重疊層、下伏層與基礎層之間的關係的概念圖。重疊堆疊400的結構可類似於圖3B中所展示之重疊堆疊320。如圖4A中所展示的，"層3"看來似乎最接近顯示器之觀察者且"層-3"看來似乎距顯示器之觀察者最遠。層402具有為零之層等級且經界定為重疊堆疊400之基礎層。基礎層402可含有窗口表面，其可由圖形處理器110來描繪且經儲存作為表面緩衝器112內之經描繪之表面。因為正層(亦即，"層1"、"層2"及"層3")重疊或看來似乎在基礎層402之前面，所以將該等層界定為重疊層404。因為負層(亦即，"層-1"、"層-2"及"層-3")下伏或看來似乎在基礎層402之後面，所以將該等層界定為下伏層。換言之，此等層被基礎層402阻塞。根據一態樣，如圖4A中所展示的，層等級愈正，表面看來似乎愈接近顯示器件106(圖1)之觀察者。同樣地，層等級愈負，表面看來似乎距顯示器件106之觀察者愈遠。換言之，具有較高表面等級之表面

中所含有的物件可能看來似乎在具有較低表面等級之表面中所含有的物件之前面，且具有較低表面等級之表面中所含有的物件可能看來似乎在具有較高表面等級之表面中所含有的物件之背面或後面。

每一層可具有結合至該層之一或多個表面。在一態樣中，基礎層402必須具有結合至該層之螢幕上窗口表面。對於每一連續圖形圖框，螢幕上表面可由圖形處理器110來描繪。此外，根據此態樣，僅螢幕外表面(亦即，pbuffer、pixmap)可結合至重疊層404與下伏層406。螢幕外表面可藉由可程式處理器108、110或114中之任一者來描繪以及自記憶體104中(諸如，自表面資訊124)檢索。

圖4B更詳細地說明實例層410。層410可為圖4A中所說明之包括基礎層402、重疊層404或下伏層406中之任一者的重疊堆疊400內之層。層410包括表面412、414、416、418。在某些態樣中，表面412、414、416、418均可為螢幕外表面。可回應於API指令或使用者程式指令而藉由可程式處理器108、110、114之一者將表面412、414、416、418中之每一者結合至層410。一般而言，層內之每一表面經指派相同表面等級，該表面等級可對應於層等級。舉例而言，若層410為重疊堆疊400中之"層3"，則可為表面412、414、416、418中之每一者指派一為三之表面等級。在結合至同一層之兩個表面彼此重疊之狀況下(諸如，層410中之表面414及416)，可以該等表面經結合之次序描繪該等表面。舉例而言，若表面416係在表面414之後被結合，則表面416可能看

來似乎比表面414更接近觀察者。否則，若表面416係在表面414之前被結合，則表面414可能看來似乎較接近觀察者。在其他狀況下，不同描繪次序可用於重疊表面，諸如以與表面結合至層之次序相反的次序進行描繪。

顯示處理器114可組合重疊堆疊400中之層與表面以產生可發送至圖框緩衝器160或顯示器106之所得的圖形圖框。顯示處理器114可根據一或多個複合模式組合圖形表面，諸如：(1)覆寫，(2) α 摻合，(3)不具有 α 摻合之色彩鍵控，及(4)具有 α 摻合之色彩鍵控。

圖5A為根據一態樣說明圖1中所展示之API程式庫120之其他細節的方塊圖。如先前參看圖1所描述的，在圖形處理器110、控制處理器108及/或顯示處理器114之應用程式執行期間，API程式庫120可儲存於記憶體104中且藉由應用程式指令118來鏈接或參考。圖5B為根據一態樣說明圖1中所展示之驅動程式122之其他細節的方塊圖。在圖形處理器110、控制處理器108及/或顯示處理器114之應用程式執行期間，驅動程式122可儲存於記憶體104中且藉由應用程式指令118及/或API程式庫120來鏈接或參考。

在圖5A之實例中，API程式庫120包括OpenGL ES描繪API 502、OpenVG描繪API 504、EGL API 506，及下伏本地平台描繪API 508。圖5B中所展示之驅動程式122包括OpenGL ES描繪驅動程式522、OpenVG描繪驅動程式524、EGL驅動程式526，及下伏本地平台描繪驅動程式528。OpenGL ES描繪API 502為由應用程式指令118在圖形處理系統102之

應用程式執行期間調用以提供 OpenGL ES 支援的描繪函數 (諸如，2D 與 3D 描繪函數) 的 API。OpenGL ES 描繪驅動程式 522 由應用程式指令 118 及 / 或 OpenGL ES 描繪 API 502 在應用程式執行期間調用，以用於圖形處理系統 102 內之 OpenGL ES 描繪函數之低階驅動程式支援。

OpenVG 描繪 API 504 為由應用程式指令 118 在應用程式執行期間調用以提供 OpenVG 支援的描繪函數 (諸如，2D 向量圖形描繪函數) 的 API。OpenVG 描繪驅動程式 524 由應用程式指令 118 及 / 或 OpenVG 描繪 API 504 在應用程式執行期間調用，以用於圖形處理系統 102 內之 OpenVG 描繪函數之低階驅動程式支援。

EGL API 506 (圖 5A) 及 EGL 驅動程式 526 (圖 5B) 提供對圖形處理系統 102 中之 EGL 函數的支援。可將 EGL 擴展併入 EGL API 506 及 EGL 驅動程式 526 內。術語 EGL 擴展可指代擴展標準 EGL 規格或對標準 EGL 規格添加功能性之一或多個 EGL API 及 / 或 EGL 驅動程式的組合。可藉由修改現存的 EGL API 及 / 或現存的 EGL 驅動程式或藉由產生新的 EGL API 及 / 或新的 EGL 驅動程式來產生 EGL 擴展。在圖 5A-5B 之實例中，提供表面重疊 EGL 擴展，亦支援對 EGL 標準函數 *eglSwapBuffers* 之修改。因此，對於 EGL 表面重疊擴展，表面重疊 API 510 被包括於 EGL API 506 內且表面重疊驅動程式 530 被包括於 EGL 驅動程式 526 內。對於經修改之 EGL 函數 *eglSwapBuffers*，標準交換緩衝器 API 512 被包括於 EGL API 506 內且經修改之交換緩衝器驅動程式 532 被包括於 EGL 驅

動程式526內。

EGL表面重疊擴展提供用於將多個圖形表面(諸如，2D表面、3D表面及/或視訊表面)重疊至單一圖形圖框中之表面重疊堆疊。該等圖形表面可各自具有堆疊內之相關聯之表面等級。藉此根據堆疊內之表面之重疊次序達成表面之重疊。在一態樣中，EGL表面重疊擴展可提供用於重疊堆疊之產生及維護之函數。舉例而言，EGL表面重疊擴展可提供用於允許使用者程式或其他API產生重疊堆疊且將表面結合至重疊堆疊內之各種層的函數。EGL表面重疊堆疊亦可允許使用者程式或API函數選擇性地啟用或停用重疊堆疊內之表面或全部層，以及選擇性地啟用或停用重疊堆疊本身。最後，EGL表面重疊API可提供將表面結合組態以及表面等級資訊傳回使用者程式或用戶端API之函數。以此方式，EGL表面重疊擴展提供了含有螢幕上表面與螢幕外表面之重疊堆疊之產生及管理。

經修改之交換緩衝器驅動程式532可對重疊堆疊中之表面執行複雜計算且設立當組合該等表面時由顯示處理器114使用之各種資料結構。為了準備此資料，經修改之交換緩衝器驅動程式532可遍曆重疊堆疊，自具有最低等級之層開始且以直至含有窗口表面之基礎層之次序進行。在每一層內，驅動程式532可藉由以每一表面結合至層之次序處理每一表面而進行。接著，驅動程式532可處理含有窗口表面(且在某些狀況下，其他表面)之基礎層(亦即，層0)。最後，驅動程式532將繼續處理重疊層，自層等級1開始且以直至

最高層(其看來似乎最接近顯示器之觀察者)的次序進行。以此方式，經修改之EGL交換緩衝器驅動程式532系統地處理每一表面以便準備用於顯示處理器114在描繪圖形圖框時使用之資料。

如圖5A中所展示的，API程式庫120亦包括下伏本地平台描繪API 508。API 508為由由器件100實施之下伏本地平台在應用程式指令118之執行期間提供的彼等API。EGL API 506提供下伏本地平台描繪API 508與OpenGL ES描繪API 502和OpenVG描繪API 504之間的平台介面層。如圖5B中所展示的，驅動程式122包括下伏本地平台描繪驅動程式528。驅動程式528為由由器件100實施之下伏本地平台在應用程式指令118及/或API程式庫120之執行期間提供的彼等驅動程式。EGL驅動程式526提供介於下伏本地平台描繪驅動程式528與OpenGL ES描繪驅動程式522和OpenVG描繪驅動程式524之間的平台介面層。

圖6為說明根據本揭示案之另一態樣之器件600的方塊圖，器件600可用於將一組經描繪之圖形表面重疊或組合至圖形圖框上。在此態樣中，圖6中所展示之器件600為圖1中所展示之器件100的實例實例化。器件600包括圖形處理系統602、記憶體604及顯示器件606。類似於圖1中所展示之記憶體104，圖6之記憶體604包括用於應用程式指令618、API程式庫620、驅動程式622及表面資訊624之儲存空間。類似於圖1中所展示之圖形處理系統102，圖2之圖形處理系統602包括控制處理器608、圖形處理器610、顯示處理

器 614、表面緩衝器 612，及圖框緩衝器 660。雖然展示為包括於圖 6 中之圖形處理系統 602 內，但表面緩衝器 612 及圖框緩衝器 660 之一者或兩者在一態樣中亦可直接包括於記憶體 604 內。

圖形處理器 610 包括圖元處理單元 662 及像素處理單元 664。圖元處理單元 662 執行關於圖形處理系統 602 內之圖元之操作。圖元藉由頂點來界定且可包括點、線段、三角形、矩形。該等操作可包括圖元變換操作、圖元照明操作及圖元截割操作。像素處理單元 664 對圖形處理系統 602 內之個別像素或片段執行像素操作。舉例而言，像素處理單元 664 可執行 OpenGL ES API 中所指定的像素操作。該等操作可包括像素所有權測試、剪裁測試、多樣本片段操作、 α 測試、模板測試、深度緩衝器測試、摻合、抖動及邏輯運算。

顯示處理器 614 藉由根據重疊堆疊及選定的複合演算法重疊及下伏表面而組合圖形處理系統 602 內之兩個或兩個以上表面。複合演算法可係基於選定的複合模式，諸如：(1) 覆寫，(2) α 摻合，(3) 不具有 α 摻合之色彩鍵控，及 (4) 具有 α 摻合之色彩鍵控。顯示處理器 614 包括覆寫區塊 668、摻合單元 670、色彩鍵控區塊 672，及組合的色彩鍵控 α 摻合區塊 674。

覆寫區塊 668 執行用於顯示處理器 614 之覆寫操作。在兩個或兩個以上經描繪之圖形表面重疊之情況下，覆寫區塊 668 可選擇經描繪之圖形表面中之具有最高表面等級的一者，且格式化圖形圖框以使得對於經描繪之圖形表面之重

疊部分顯示經描繪之圖形表面之選定者。

摻合單元670執行用於顯示處理器614之摻合操作。摻合操作可包括(例如)全表面恆定 α 摻合操作及全表面每像素 α 摻合操作。

色彩鍵控區塊672執行不具有 α 摻合之色彩鍵控操作。舉例而言，色彩鍵控區塊672可檢查兩個重疊表面之較高表面之重疊部分內的每一像素以確定哪些像素匹配基色(例如，洋紅色)。對於匹配基色之任何像素，色彩鍵控區塊672可選擇來自較低表面之對應像素(亦即，具有相同顯示位置之像素)作為輸出像素(亦即，經顯示之像素)。對於不匹配基色之任何像素，色彩鍵控區塊672可選擇來自較高表面之像素作為輸出像素。

組合的色彩鍵控 α 摻合區塊674執行色彩鍵控操作以及 α 摻合操作。舉例而言，區塊674可檢查兩個重疊表面之較高表面之重疊部分內的每一像素以確定哪些像素匹配基色。對於匹配基色之任何像素，區塊674可選擇來自較低表面之對應像素(亦即，具有相同顯示位置之像素)作為輸出像素。對於不匹配基色之任何像素，區塊674可摻合較高表面之像素與來自較低表面之對應像素以產生輸出像素。

雖然圖6中將顯示處理器614展示為具有四個例示性操作區塊，但在其他實例中，顯示處理器614可具有執行各種像素摻合及鍵控演算法的或多或少之操作區塊。在某些狀況下，顯示處理器614能夠執行之獨特像素操作的總數目可小於圖形處理器610能夠執行之獨特像素操作的總數目。顯示

處理器 114 亦可對經描繪之圖形表面或圖框執行其他描繪後函數(包括縮放及旋轉)。

藉由將圖形處理劃分成執行複雜描繪之圖形處理器 610 及組合螢幕上表面與螢幕外表面之顯示處理器 614，且藉由並列操作處理器 610 及 614，可減少圖形處理器 610 之時鐘率。此外，因為功率消耗隨著圖形處理器 610 之時鐘率非線性地增加，所以可在圖形處理系統 602 中達成顯著功率節省。

在某些態樣中，顯示處理器 614 可包括比圖形處理器 610 中之圖形管道簡單的圖形管道。舉例而言，顯示處理器 614 中之圖形管道可不執行任何圖元操作。作為另一實例，由摻合單元 670 提供之像素操作之總數目可小於由像素處理單元 664 提供之像素操作之總數目。藉由將減少量之操作包括於顯示處理器 614 內，可簡化圖形管道並使其為流線型的以提供顯著功率節省且增加圖形處理系統 602 中之處理量。

在一態樣中，圖形應用程式可含有在連續圖框之間相對靜態之許多物件。在視訊遊戲應用程式中，靜態物件之實例可包括十字絲、計分盒、計時器、速度計及展示於視訊遊戲顯示器上之其他固定或不變元件。應注意，靜態物件在連續圖形圖框之間可具有某些運動或改變，但此等運動或改變之性質可能常常不需要逐圖框地重新描繪全部物件。在圖形處理系統 602 中，靜態物件可經指派至螢幕外表面且藉由使用比可用於圖形處理器 610 中之複雜圖形管道簡單之圖形管道來描繪。舉例而言，控制處理器 608 可能夠

使用比圖形處理器610少之功率描繪簡單2D表面。此等表面可接著由控制處理器608來描繪且被直接發送至顯示處理器614以用於與可能更複雜之其他表面組合。在此實例中，用於描繪靜態2D表面之組合圖形管道(亦即，控制處理器608及顯示處理器614)可消耗比圖形處理器610之複雜圖形管道少之功率。另一方面，圖形處理器610可能夠比控制處理器608更有效率地描繪複雜3D圖形。因此，藉由依待描繪之物件之特徵而選擇性地選擇不同圖形管道，器件600可用於更有效率地描繪圖形表面。

作為另一實例，控制處理器608可檢索或指導顯示處理器614檢索儲存於記憶體604或表面緩衝器612中的預先描繪之表面。預先描繪之表面可藉由使用者應用程式來提供或當較未大量利用資源時，可由圖形處理系統602來產生。控制處理器608可給預先描繪之表面中之每一者指派一表面等級且指導顯示處理器614根據選定的表面等級組合預先描繪之表面與其他經描繪之表面。以此方式，預先描繪之表面完全繞過圖形處理器610中之複雜圖形管道，此為器件600提供了顯著功率節省。

作為又一實例，某些物件可為相對靜態的，但不是完全靜態的。舉例而言，賽車遊戲可具有帶有指示車之當前速度之針的速度計刻度盤。速度計刻度盤可為完全靜態的，因為刻度盤在連續圖框中不會改變或移動，且針可為相對靜態的，因為針隨著車速改變而逐圖框地輕微地移動。不是在每一圖框使用圖形處理器610中之複雜圖形管道描繪

針，而是可在不同的預先描繪之表面上提供針之若干不同的實例化。舉例而言，每一預先描繪之表面可描述不同位置中之針以指示車之不同速度。可將所有此等表面結合至重疊堆疊。接著，當車速改變時，控制處理器608可啟用針之指示車之當前速度的適當實例化且停用針之其他實例化。因此，不需要在每一圖框由控制處理器608或圖形處理器610來描繪可能在連續圖框之間改變位置之針。藉由選擇性地啟用及停用預先描繪之表面，器件600能夠更有效率地描繪圖形圖框。

圖7為說明根據本揭示案之另一態樣之器件700的方塊圖，器件700可用於將一組經描繪之圖形表面重疊或組合至圖形圖框上。圖7描繪除包括兩個圖形處理器以及兩個表面緩衝器之外結構類似於圖1中所展示之器件100的器件700。器件700包括圖形處理系統702、記憶體704及顯示器件706。類似於圖1中所展示之記憶體104，圖7之記憶體704包括用於應用程式指令718、API程式庫720、驅動程式722及表面資訊724之儲存空間。類似於圖1中所展示之圖形處理系統102，圖7之圖形處理系統702包括控制處理器708、顯示處理器714，及圖框緩衝器760。雖然展示為包括於圖7中之圖形處理系統702內，但表面緩衝器712、表面緩衝器713及圖框緩衝器760中之任一者在某些態樣中亦可直接包括於記憶體704內。

圖形處理系統702亦包括3D圖形處理器710、2D圖形處理器711、3D表面緩衝器712，及2D表面緩衝器713。如圖7中

所展示的，圖形處理器710、711中之每一者操作性地耦接至控制處理器708及顯示處理器714。此外，表面緩衝器712、713中之每一者操作性地耦接至控制處理器708、顯示處理器714及圖框緩衝器760。3D圖形處理器710操作性地耦接至3D表面緩衝器712以在圖形處理系統702內形成3D圖形處理管道。類似地，2D圖形處理器711操作性地耦接至2D表面緩衝器713以在圖形處理系統702內形成2D圖形處理管道。類似於圖1中之表面緩衝器112，3D表面緩衝器712及2D表面緩衝器713可各自包含一或多個表面緩衝器，且該一或多個表面緩衝器中之每一者可儲存一或多個經描繪之表面。

在一態樣中，3D圖形處理器710可包括有效地實施3D描繪演算法之加速3D圖形描繪管道。2D圖形處理器711可包括有效地實施2D描繪演算法之加速2D圖形描繪管道。舉例而言，3D圖形處理器710可根據OpenGL ES API命令有效地描繪表面，且2D圖形處理器711可根據OpenVG API命令有效地描繪表面。

在圖形處理系統702中，控制處理器708可使用3D描繪管道(亦即，710、712)描繪3D表面，且亦可使用2D描繪管道(亦即，711、713)描繪2D表面。舉例而言，3D圖形處理器710可描繪第一組經描繪之圖形表面且將該第一組經描繪之圖形表面儲存於3D表面緩衝器712中。同樣地，2D圖形處理器711可描繪第二組經描繪之圖形表面且將該第二組經描繪之圖形表面儲存於2D表面緩衝器713中。顯示處理器714

可自表面緩衝器712中檢索3D經描繪之圖形表面及自表面緩衝器713中檢索2D經描繪之圖形表面且根據選定用於每一表面之表面等級或根據重疊堆疊而重疊2D表面與3D表面。在此實例中，描繪管道中之每一者具有用於儲存經描繪之2D及3D表面之專用表面緩衝器。由於圖形處理系統702內之專用表面緩衝器712、713，使得可不需要使處理器710與711彼此同步。換言之，根據一態樣，3D圖形處理器710與2D圖形處理器711可獨立於彼此操作，而不必協調表面緩衝器寫入操作之時序。因為不需要使處理器同步，所以圖形處理系統702內之處理量得以改良。因此，圖形處理系統702提供了藉由使用單獨的3D圖形加速管道及單獨的2D圖形加速管道而有效地描繪表面。

圖8為用於重疊或組合經描繪之圖形表面之方法800的流程圖。出於說明之目的，隨後描述關於圖1中之器件100描述方法800之效能。然而，可使用圖1、圖6或圖7中所展示之器件中之任一者來執行方法800。此外，在某些狀況下，該描述可指定特定可程式處理器執行特定操作。然而，應注意，可程式處理器108、110、114之一或多者可執行關於方法800所描述之動作之任一者。

如圖8中所展示的，顯示處理器114可檢索複數個經描繪之圖形表面(802)。經描繪之圖形表面可藉由可程式處理器108、110或114之一者來產生或描繪。在一態樣中，可程式處理器108、110或114之一者可將經描繪之圖形表面儲存於一或多個表面緩衝器112內或記憶體104內，且顯示處理器

114可自該一或多個表面緩衝器112中或自記憶體104中檢
索經描繪之圖形表面。在某些態樣中，圖形處理器110可至
少部分藉由使用加速3D圖形管道來描繪表面。此外，控制
處理器108可至少部分藉由使用通用處理管道來描繪一或
多個圖形表面。在一態樣中，圖形處理系統102可預先描繪
一或多個圖形表面且將預先描繪之圖形表面儲存於記憶體
104或表面緩衝器112中。在某些狀況下，處理器108或110
可將經描繪之表面直接發送至顯示處理器114而不將經描
繪之表面儲存於表面緩衝器112中。

選擇用於經描繪之圖形表面中之每一者的表面等級
(804)。舉例而言，控制處理器108或圖形處理器110可選擇
用於經描繪之表面之表面等級且將選定的表面等級117儲
存於表面緩衝器112中。在另一態樣中，應用程式指令118
或API程式庫120中之API函數可選擇表面等級且將選定的
表面等級117儲存於表面緩衝器112中。在某些態樣中，可
藉由將經描繪之表面結合至重疊堆疊而選擇選定的表面等
級。重疊堆疊可含有複數個層，每一層具有一獨特層等級。
可藉由選擇重疊堆疊中用於每一經描繪之表面之特定層且
將每一經描繪之表面結合至選定用於該表面之層而選擇表
面等級。當兩個或兩個以上表面結合至同一層時，可進一
步藉由確定用於該兩個或兩個以上表面之結合次序來選擇
表面等級。在某些狀況下，可將選定的表面等級直接發送
至顯示處理器114而不將表面等級儲存於表面緩衝器112
中。在一態樣中，可在將經描繪之圖形表面中之任一者輸

出至顯示器之前選擇用於經描繪之圖形表面中之每一者的表面等級。在另一態樣中，可在描繪特定的經描繪之圖形表面之前選擇用於該特定表面之表面等級。

顯示處理器 114 根據選定的表面等級將經描繪之圖形表面重疊至圖形圖框上 (806)。重疊經描繪之圖形表面可包括組合經描繪之表面與一或多個其他經描繪之圖形表面。在一態樣中，顯示處理器 114 可根據一或多個複合或摻合模式組合圖形表面。該等複合模式之實例包括 (1) 覆寫、(2) α 摻合、(3) 不具有 α 摻合之色彩鍵控及 (4) 具有 α 摻合之色彩鍵控。當顯示處理器 114 根據覆寫複合模式組合經描繪之圖形表面且兩個或兩個以上經描繪之圖形表面重疊時，顯示處理器 114 可選擇經描繪之圖形表面中之具有最高表面等級的一者，且格式化圖形圖框以便對於經描繪之圖形表面之重疊部分顯示經描繪之圖形表面之選定者。顯示處理器 114 可根據界定複數個層之重疊堆疊而組合表面。每一層可具有獨特層等級且包括結合至該層之一或多個表面。顯示處理器 114 在某些狀況下接著可遍曆該重疊堆疊以確定組合該等表面之次序。當兩個或兩個以上表面結合至同一層時，顯示處理器可進一步藉由確定用於該兩個或兩個以上表面之結合次序而確定組合該等表面之次序。當根據重疊堆疊來重疊經描繪之圖形表面時，顯示處理器 114 可格式化圖形圖框以使得當圖形圖框顯示於顯示器上時，結合至重疊堆疊內具有第一層等級之層的經描繪之圖形表面看來似乎比結合至具有低於該第一層等級之層等級之層的經描繪

之圖形表面較接近顯示器之觀察者。在顯示處理器114重疊經描繪之圖形表面之後，顯示處理器114可將圖形圖框輸出至圖框緩衝器160或至顯示器106。

圖9為用於重疊或組合經描繪之圖形表面之方法900的流程圖。出於說明之目的，隨後描述關於圖1中之器件100而描述方法900之效能。然而，方法900可使用圖1、圖6或圖7中所展示之器件中之任一者來執行。此外，在某些狀況下，該描述可指定特定可程式處理器執行特定操作。然而，應注意，可程式處理器108、110、114之一或多者可執行關於方法900所描述之動作之任一者。此外，方法900僅僅為使用本揭示案中所描述之技術之方法的實例。因此，操作之排序可不同於圖9中所展示之次序。

圖形處理器110可描繪螢幕上表面(902)。螢幕上表面可為窗口表面且可使用加速3D圖形管道來描繪。可程式處理器108、110、114之一者可產生用於螢幕上表面之重疊堆疊(904)。該重疊堆疊可具有複數個層且儲存於記憶體104之表面資訊區塊124中、表面緩衝器112內或圖形處理系統102內之其他緩衝器(未圖示)內。可程式處理器108、110之一者可描繪一或多個螢幕外表面(906)。實例螢幕外表面可包括pbuffer表面及pixmap表面。在某些狀況下，可藉由使用通用處理管道來描繪此等表面。可程式處理器108、110、114之一者可選擇重疊堆疊內之用於每一螢幕外表面之層(908)。作為實例，窗口表面可結合至重疊堆疊內之基礎層(亦即，層零)，且選定之層可包含重疊或看來似乎在基礎層

之前面之重疊層，及下伏或看來似乎在基礎層之後面之下伏層。在一態樣中，選定之層亦可包含基礎層。

如圖9中進一步展示的，可程式處理器108、110、114之一者確定用於含有兩個或兩個以上重疊表面之每一層之表面結合次序(910)。該表面結合次序可係基於用於給定層之表面之所要的顯示次序。舉例而言，結合至特定層之表面可看來似乎在於稍後時間結合至同一層之任何表面的後面。可程式處理器108、110、114之一者可根據表面結合次序將螢幕外表面結合至重疊堆疊之個別選定層(912)。在一態樣中，除螢幕外表面之外，可程式處理器108、110、114之一者可將螢幕上表面結合至重疊堆疊內之層。可程式處理器108、110、114之一者接著可對於待顯示之每一圖形圖框選擇性地啟用或停用重疊堆疊內的個別表面或層(914)，且接著選擇用於結合至重疊堆疊之每一表面之複合或摻合模式(916)。複合或摻合模式可為簡單覆寫、具有恆定 α 摻合之色彩鍵控、不具有恆定 α 摻合之色彩鍵控、全表面恆定 α 摻合或全表面每像素 α 摻合之一者。最後，顯示處理器114根據重疊堆疊、表面結合次序及選定的摻合模式而組合或重疊該等表面(918)。在一態樣中，當對於圖形圖框啟用重疊堆疊內之層時，顯示處理器114可處理結合至該層之經描繪之圖形表面中的每一者以產生圖形圖框。同樣地，當對於圖形圖框停用重疊堆疊內之層時，顯示處理器114可不處理結合至該層之任何經描繪之圖形表面以產生圖形圖框。在另一態樣中，當對於圖形圖框啟用經描繪之圖形表面

時，顯示處理器114可處理經描繪之圖形表面以產生第一圖形圖框。相反地，當對於第一圖形圖框停用經描繪之圖形表面時，顯示處理器114可不處理經描繪之圖形表面以產生第一圖形圖框。在某些狀況下，可將與重疊堆疊相關聯之窗口表面視為主要窗口表面。根據一態樣，可不停用主要窗口表面。在其他態樣中，可停用主要窗口表面。

在一態樣中，提供EGL擴展以用於組合一組EGL表面以產生所得的圖形圖框。EGL擴展可提供與設立重疊堆疊及組合表面有關的至少七個新函數。以下展示用於七個新函數之實例函數宣告：

```
EGLSurface eglCreateCompositeSurfaceQUALCOMM( EGLDisplay dpy,  
                                           EGLSurface win,  
                                           const EGLint  
                                           *attrib_list );
```

```
EGLBoolean eglSurfaceOverlayEnableQUALCOMM ( EGLDisplay dpy,  
                                           EGLSurface surf,  
                                           EGLBoolean enable );
```

```
EGLBoolean eglSurfaceOverlayLayerEnableQUALCOMM ( EGLDisplay dpy,  
                                           EGLSurface comp_surf,  
                                           EGLint layer,  
                                           EGLBoolean enable );
```

```

EGLBoolean eglSurfaceOverlayBindQUALCOMM ( EGLDisplay dpy,
                                           EGLSurface comp_surf,
                                           EGLSurface surf,
                                           EGLint layer,
                                           EGLBoolean enable );

```

```

EGLBoolean eglGetSurfaceOverlayBindingQUALCOMM ( EGLDisplay dpy,
                                                  EGLSurface surf,
                                                  EGLSurface *comp_surf,
                                                  EGLint *layer);

```

```

EGLBoolean eglGetSurfaceOverlayQUALCOMM ( EGLDisplay dpy,
                                           EGLSurface surf,
                                           EGLBoolean *layer_enable,
                                           EGLBoolean *surf_enable );

```

```

EGLBoolean eglGetSurfaceOverlayCapsQUALCOMM ( EGLDisplay dpy,
                                              EGLSurface win,
                                              EGLCompositeSurfaceCaps
                                              *param);

```

可調用 *eglCreateCompositeSurfaceQUALCOMM* 函數來產生複合表面及 / 或重疊堆疊。可調用 *eglSurfaceOverlayEnableQUALCOMM* 函數來啟用或停用整個重疊堆疊或與重疊堆疊相關聯之個別表面。可調用 *eglSurfaceOverlayLayerEnableQUALCOMM* 函數來啟用或

停用重疊堆疊內之特定層。可調用 *eglSurfaceOverlayBindQUALCOMM* 函數來將表面結合或附著於重疊堆疊內之特定層。可調用 *eglGetSurfaceOverlayBindingQUALCOMM* 函數來確定複合表面(亦即，重疊堆疊)及重疊堆疊內特定表面所結合至之層。可調用 *eglGetSurfaceOverlayQUALCOMM* 函數來確定是否啟用特定表面以及是否啟用彼表面所結合至之層。可調用 *eglGetSurfaceOverlayCapsQUALCOMM* 函數來接收用於特定驅動程式及硬體環境中之複合表面的實施限制。

在一態樣中，EGL擴展可提供額外資料類型結構。一該結構提供用於複合表面(亦即，重疊堆疊)之實施限制。複合表面可儲存重疊堆疊或以其它方式與重疊堆疊相關聯。以下展示實例資料結構：

```
typedef struct
{
    EGLint      max_overlay;

    EGLint      max_underlay;

    EGLint      max_surface_per_layer;

    EGLint      max_total_surfaces;

    EGLBoolean  pBuffer_support;

    EGLBoolean  pixmap_support;

} EGLCompositeSurfaceCaps;
```

四個 *EGLint* 項分別提供：允許用於複合表面之重疊層之最大數目；允許用於複合表面之下伏表面之最大數目；允

許結合至或附著於重疊堆疊內之每一層之表面的最大數目；及允許結合至重疊堆疊內之所有層之表面的最大總數目。第一 *EGLBoolean* 項提供與複合表面是否支援 pbuffer 表面有關之資訊，且第二 *EGLBoolean* 項提供與複合表面是否支援 pixmap 表面有關之資訊。

EGL EGLSurface 資料結構可含有用於經描繪之表面的三個額外類型項 *EGLCompSurf*、*EGLBoolean* 及 *EGLCompositeSurfaceCaps*。*EGLCompSurf* 項提供至相關聯之複合表面之位址的指標，*EGLBoolean* 項確定是否啟用經描繪之表面，且 *EGLCompositeSurfaceCaps* 項提供關於相關聯之複合表面之實施限制的資訊。

EGLCompSurf 資料結構可含有與複合表面(亦即，重疊堆疊)有關之資訊。以下展示實例 *EGLCompSurf* 資料結構：

```
typedef struct
{
    EGLBoolean    OverlayEnable

    EGLSurface    WindowSurf

    EGLCompLayer  *Overlays[MAX_OVLY]

    EGLCompLayer  *Underlays[MAX_UNDLY]

} EGLCompSurf;
```

EGLCompSurf 資料結構可含有用於複合表面之至少四項：一 *EGLBoolean* 類型項；一 *EGLSurface* 類型項；及兩個至 *EGLCompLayer* 之指標類型陣列項。*EGLBoolean* 項提供用於重疊堆疊之啟用旗標，且 *EGLSurface* 項提供至相關聯之

窗口表面之指標。兩個 *EGLCompLayer* 陣列項提供至在重疊堆疊內之特定 *EGLCompLayer* 項之指標陣列。第一 *EGLCompLayer* 陣列項可提供至重疊堆疊內之重疊層之位址指標陣列，且第二 *EGLCompLayer* 陣列項可提供至下伏堆疊內之下伏層之位址指標陣列。

EGLCompLayer 資料結構可含有與重疊堆疊內之個別層有關之資訊。以下展示實例 *EGLCompLayer* 資料結構：

typedef struct

```

{
    EGLint      Level

    EGLint      surf_count

    EGLBoolean   OverlayEnable

    EGLSurface   Surfaces[MAX_LAYER_SURF]

} EGLCompLayer;
```

EGLCompLayer 資料結構可含有用於複合表面之至少四項：兩個 *EGLint* 類型項；一 *EGLBoolean* 類型項；及一 *EGLSurface* 類型陣列項。第一 *EGLint* 項提供層之等級，且第二 *EGLint* 項提供結合或附著於層之表面之總數目。*EGLSurface* 陣列項提供至結合或附著於層之表面之指標陣列。

可調用函數 *eglCreateCompositeSurfaceQUALCOMM* 來產生特定複合表面。使用者程式或 API 可將若干參數傳遞至該函數，若干參數包括至 *EGLDisplay* (亦即，繪製圖形之抽象顯示器) 及將用作用於重疊堆疊之窗口表面的 *EGLSurface*

類型之窗口表面的指標。此外，使用者程式或API可傳遞界定所得的複合表面之所要屬性之*EGLint*陣列資料結構。該函數可傳回包括重疊堆疊之*EGLSurface*類型之複合表面。

可調用 *eglSurfaceOverlayEnableQUALCOMM* 函數來啟用或停用重疊堆疊或與複合表面相關聯之個別表面。使用者程式或API可傳遞至適當 *EGLDisplay* 之指標以及至 *EGLSurface* (其為含有重疊堆疊之複合表面或重疊堆疊內之個別表面) 之指標。使用者程式或API亦傳遞指示是啟用還是停用表面之 *EGLBoolean* 參數。該函數可傳回指示函數成功了還是發生了錯誤之 *EGLBoolean* 參數。

可調用 *eglSurfaceOverlayLayerEnableQUALCOMM* 函數來啟用或停用重疊堆疊內之特定層。使用者程式或API可傳遞至適當 *EGLDisplay* 之指標以及至 *EGLSurface* (其為含有重疊堆疊之複合表面) 之指標。使用者程式或API亦傳遞指示啟用還是停用重疊堆疊內之所要層的 *EGLint* 參數。最後，使用者程式或API亦傳遞指示啟用還是停用重疊堆疊內所含有之層的 *EGLBoolean* 參數。該函數可傳回指示函數成功了還是發生了錯誤之 *EGLBoolean* 參數。

可調用 *eglSurfaceOverlayBindQUALCOMM* 函數來將表面結合或附著於重疊堆疊內之特定層。使用者程式或API可傳遞至適當 *EGLDisplay* 之指標以及至 *EGLSurface* (其為含有重疊堆疊之複合表面) 之指標。此外，使用者程式或API可傳遞至將結合至重疊堆疊之 *EGLSurface* 之位址指標及指示表面應結合至重疊堆疊內之哪個層之 *EGLint* 類型值。最

後，使用者程式或API亦傳遞指示是啟用還是停用個別表面之*EGLBoolean*參數。該函數可傳回指示函數成功了還是發生了錯誤之*EGLBoolean*參數。

可調用 *eglGetSurfaceOverlayBindingQUALCOMM* 函數來確定重疊堆疊內特定表面所結合至之層。使用者程式或API可傳遞至適當 *EGLDisplay* 之指標以及至 *EGLSurface* (找尋其層資訊) 之指標。使用者程式或API亦傳遞至 *EGLSurface* 指標之指標，其中傳回含有重疊堆疊之複合表面。最後，使用者程式或API傳遞 *EGLint* 指標，該函數使用該 *EGLint* 指標傳回表面所結合至之層等級。該函數可傳回指示函數成功了還是發生了錯誤之 *EGLBoolean* 參數。

可調用 *eglGetSurfaceOverlayQUALCOMM* 函數來確定是否啟用特定表面以及是否啟用彼表面所結合至之層。使用者程式或API可傳遞至適當 *EGLDisplay* 之指標以及至 *EGLSurface* (其為資訊被找尋之表面) 之指標。使用者程式或API亦可傳遞兩個 *EGLBoolean* 指標，該函數使用該兩個 *EGLBoolean* 指標傳回關於表面之資訊。第一 *EGLBoolean* 參數指示是否啟用表面所結合至之層，且第二 *EGLBoolean* 參數指示是否啟用特定表面。該函數可傳回指示函數成功了還是發生了錯誤之 *EGLBoolean* 參數。

可調用 *eglGetSurfaceOverlayCapsQUALCOMM* 函數來接收用於與特定本地窗相關聯之複合表面或重疊堆疊之實施限制。使用者程式或API可傳遞至適當 *EGLDisplay* 之指標以及至 *EGLSurface* (其為實施限制資訊被找尋之窗口表面) 之

指標。此外，使用者程式或 API 傳遞至 *EGLCompositeSurfaceCaps* 類型之資料之指標，該函數使用該指標傳回允許用於特定窗口表面之複合表面之實施限制。該函數可傳回指示函數成功了還是發生了錯誤之 *EGLBoolean* 參數。

為了提供支援組合一組 EGL 表面以產生所得的圖形圖框之 EGL 擴展之實施的使用實例，下文提供樣本碼。該樣本碼實施以下之情況：其中目標器件具有視訊圖形陣列 (VGA) 液晶顯示器 (LCD) 螢幕，但應用程式將 3D 內容描繪至四分之一視訊圖形陣列 (QVGA) 窗口表面以改良效能且減少功率消耗。一旦組合該等層，程式碼就接著將解析度按比例放大至全 VGA 螢幕尺寸。

應用程式為賽車遊戲，其在下伏層中具有部分包廂。該遊戲亦具有重疊層，該重疊層含有圓形類比轉速計(在左下角中)及數位速度計與排擋指示器(兩者均位於右下角中)。樣本碼利用上文所列之函數及資料結構之許多者。在樣本碼中，設立表面、重疊及下伏尺寸設定以使得該等表面、重疊及下伏之尺寸小於窗口表面之尺寸。有意地進行此，以便避免所得的圖形圖框之過度平均深度複雜性。在執行該樣本之前，應進行包括產生 EGL 顯示之 EGL 初始化。

```
3D_window = eglCreateWindowSurface( dpy, config, window, NULL );
```

```
// Now resize it to QVGA (src & dst rects)
```

```
// This will save buffer memory and 3D render time
```

```

EGLSurfaceScaleRect src_rect = {0, 0, 320, 240};

EGLSurfaceScaleRect dst_rect = {0, 0, 320, 240};

eglSetSurfaceScaleQUALCOMM ( dpy, 3D_window, &src_rect, &dst_rect );

eglSurfaceScaleEnableQUALCOMM ( dpy, 3D_window, EGL_TRUE );


// Setup OpenGL ES rendering to only the QVGA region used

glViewport( 0, 0, 320, 240 );

glScissor( 0, 0, 320, 240 );


// Create a pbuffer for the skybox underlay

// Assume 3D content will always occlude lower half of QVGA window

// So we only need to update the upper half of the underlay

EGLint attribs[] = {EGL_WIDTH, 320, EGL_HEIGHT, 120, EGL_NONE};

skybox = eglCreatePbufferSurface( dpy, config, attribs);


// Position the skybox in the QVGA src rect of the composite surface

// 1 to 1 scaling (i.e. blit)

src_rect.height = 120;

dst_rect.y = 120;

dst_rect.height = 120;

eglSetSurfaceScaleQUALCOMM ( dpy, skybox, &src_rect, &dst_rect );

eglSurfaceScaleEnableQUALCOMM ( dpy, skybox, EGL_TRUE );


// Render the initial skybox

// Can use a single textured tri-strip or drawtexture for this

// Can disable depth/lighting etc for max performance

```

// Create a pbuffer for the tach dial overlay

attribs[1] = 40;

attribs[3] = 30;

dial = eglCreatePbufferSurface(dpy, config, attribs);

// Position the dial in the QVGA src rect of the composite surface

// 1 to 1 scaling (i.e. blit)

src_rect.width = 40;

src_rect.height = 30;

dst_rect.y = 0;

dst_rect.width = 40;

dst_rect.height = 30;

eglSetSurfaceScaleQUALCOMM (dpy, dial, &src_rect, &dst_rect);

eglSurfaceScaleEnableQUALCOMM (dpy, dial, EGL_TRUE);

// Can use eglSurfaceTransparency here to make the tach dial

// partially translucent

// Render the tachometer dial - this will remain static

// Create a pbuffer for the tach needle overlay

needle = eglCreatePbufferSurface(dpy, config, attribs);

// Position the needle in the QVGA src rect of the composite surface

// 1 to 1 scaling (i.e. blit)

// this directly overlays the dial... but since the dial is bound

```

// first, it will be combined first

eglSetSurfaceScaleQUALCOMM ( dpy, needle, &src_rect, &dst_rect );

eglSurfaceScaleEnableQUALCOMM ( dpy, needle, EGL_TRUE );

// Setup the color_key for the needle so the dial will show through

// use Magenta (0xFF00FF on 8-bit/comp devices)

// This assumes the setup code created a 565 window surface

eglSetSurfaceColorKeyQUALCOMM ( dpy, needle, (0xFF >> 3), 0,

                                (0xFF >> 3) );

eglSurfaceColorKeyEnableQUALCOMM ( dpy, needle, EGL_TRUE );

// Render the initial needle

// this will get updated once per frame or so

// don't update if the RPM change is small

// Be sure to set magenta as the clear color

// Create a pbuffer for the digital speedometer & gear overlay

attribs[1] = 80;

attribs[3] = 20;

speedo = eglCreatePbufferSurface( dpy, config, attribs);

// Position the speedometer in the QVGA src rect of the composite surface

// 1 to 1 scaling (i.e. blit)

src_rect.width = 80;

src_rect.height = 20;

dst_rect.x = 240;

```

```

dst_rect.width = 80;

dst_rect.height = 20;

eglSetSurfaceScaleQUALCOMM ( dpy, speedo, &src_rect, &dst_rect );

eglSurfaceScaleEnableQUALCOMM ( dpy, speedo, EGL_TRUE );

// Setup the color_key for the speedo so the 3D surface
// will show through

// use Magenta (0xFF00FF on 8-bit/comp devices)

// This assumes the setup code created a 565 window surface

eglSetSurfaceColorKeyQUALCOMM ( dpy, speedo, (0xFF >> 3), 0,
                                     (0xFF >> 3) );

eglSurfaceColorKeyEnableQUALCOMM ( dpy, speedo, EGL_TRUE );

// Render the initial speedometer and gear indicators

// this will get updated once per frame or so

// don't update if the speedometer change is small

// Create the "composite" surface

comp = eglCreateCompositeSurfaceQUALCOMM ( dpy, 3D_window, 0);

// Setup the "composite" surface to scale from QVGA to VGA

src_rect.width = 320;

src_rect.height = 240;

dst_rect.x = 0;

dst_rect.width = 640;

dst_rect.height = 480;

```

```

eglSetSurfaceScaleQUALCOMM ( dpy, comp, &src_rect, &dst_rect );

eglSurfaceScaleEnableQUALCOMM ( dpy, comp, EGL_TRUE );


// Bind the pbuffers to the overlay and underlay layers

eglSurfaceOverlayBindQUALCOMM ( dpy, comp, skybox, -1, EGL_TRUE);

eglSurfaceOverlayBindQUALCOMM ( dpy, comp, dial, 1, EGL_TRUE);

eglSurfaceOverlayBindQUALCOMM ( dpy, comp, needle, 1, EGL_TRUE);

eglSurfaceOverlayBindQUALCOMM ( dpy, comp, speedo, 1, EGL_TRUE);


// Enable the layer combining...

eglSurfaceOverlayLayerEnableQUALCOMM ( dpy, comp, -1, EGL_TRUE );

eglSurfaceOverlayLayerEnableQUALCOMM ( dpy, comp, 1, EGL_TRUE );

eglSurfaceOverlayEnableQUALCOMM ( dpy, comp, EGL_TRUE );


// Any additional EGL setup ...


// Setup OpenGL ES rendering for the main 3D window surface


// Draw calls here


// Swap to the display, the enabled layers will be combined

// as they are copied to the associated native window

eglSwapBuffers( dpy, 3D_window, 3D_window, ctx );

```

在上文之樣本碼中，產生EGL窗口表面，其最初具有640像素之寬度及480像素之高度。在此實例中，窗口表面之尺寸匹配目標顯示器VGA顯示器之尺寸。接著，將窗口表面重新尺寸設定至QVGA(320x240)顯示器之尺

寸以便節省緩衝器記憶體以及3D描繪時間。藉由設立源矩形(亦即, `src_rect`)及目的矩形(亦即, `dst_rect`)而進行重新尺寸設定。源矩形指定或選擇將重新縮放成所得表面之EGL窗口表面之一部分。目的矩形指定藉由源矩形指定的EGL窗口表面之該部分將重新縮放至之最終尺寸。因為表面為窗口表面且`src_rect`小於初始窗口尺寸, 所以使與窗口表面相關聯之緩衝器收縮以匹配新表面尺寸, 因此節省顯著記憶體空間及描繪頻寬。在設立此等變數之後, 調用 `eglSetSurfaceScaleQUALCOMM` 函數及 `eglSurfaceScaleEnableQUALCOMM` 函數以根據源矩形及目的矩形而重新尺寸設定窗口表面。

接著, 產生、重新尺寸設定及定位用於上文所描述之各種重疊及下伏層中之每一者的若干pbuffer表面。首先, 產生pbuffer表面以描繪包廂下伏表面。包廂下伏具有120像素之高度(該高度為QVGA複合表面區域之高度之一半), 且可藉由調用 `eglSetSurfaceScaleQUALCOMM` 及 `eglSurfaceScaleEnableQUALCOMM` 函數而定位於複合表面區域之上半部中。因為包廂僅在複合表面區域之上半部中可見, 所以藉由將pbuffer表面限制至複合表面區域之上半部而避免外來描繪。因此, 硬體處理量得以改良。在產生包廂下伏之後, 產生描繪轉速計刻度盤及針之兩個pbuffer重疊表面且接著藉由調用 `eglSetSurfaceScaleQUALCOMM` 函數及 `eglSurfaceScaleEnableQUALCOMM` 函數而定位該兩個pbuffer重疊表面。接著藉由調用 `eglSetSurfaceColorKeyQUALCOMM`

函數及 *eglSurfaceColorKeyEnableQUALCOMM* 函數而設立用於針重疊表面之色彩鍵控。當將經色彩鍵控之表面描繪至顯示器時，不將匹配指定的透明色彩(亦即，洋紅色)之任何像素複製至顯示器。此防止針重疊表面中所含有的背景資訊混淆轉速計刻度盤。接著，亦產生及定位描述數位速度計及排擋指示器之 pbuffer 表面重疊。亦對數位速度計及排擋指示器應用色彩鍵控。

接著，產生用於 *3D_window* 窗口表面之複合表面。設立該複合表面以將組合表面自 QVGA 尺寸縮放至目標顯示器(其為 VGA)之尺寸。接著，藉由對 *eglSurfaceOverlayBindQUALCOMM* 函數進行若干調用而將不同 pbuffer 表面結合或附著於複合表面。將包廂下伏層結合至具有 "-1" 等級之層且將對應於轉速計刻度盤、轉速計針、數位速度計及排擋指示器之其他重疊表面均結合至具有等級 "1" 之層。再一次，負層等級指示下伏層，且正層等級指示重疊層。接著，藉由調用 *eglSurfaceOverlayLayerEnableQUALCOMM* 函數而啟用重疊堆疊內之下伏層及重疊層(亦即，"-1" 及 "1")中之每一者。在啟用個別層之後，藉由調用 *eglSurfaceOverlayEnableQUALCOMM* 函數而啟用重疊堆疊本身。

在完成對 3D 窗口表面之 OpenGL ES 描繪之後，樣本碼調用經修改之 *eglSwapBuffers* 函數。將與重疊堆疊(亦即，*3D_window*)相關聯之窗口表面作為參數傳遞至該函數。在一態樣中，經修改之 *eglSwapBuffers* 函數可根據藉由樣本碼

提供之重疊堆疊、尺寸設定資訊、色彩鍵控資訊及結合資訊來組合該等表面與層。在組合該等表面與層之後，*eglSwapBuffers*函數可將所得的圖形圖框複製至相關聯之本地窗(亦即，*dpy*)中。

在另一態樣中，經修改之*eglSwapBuffers*函數可向顯示處理器114發送指令以便組合該等表面與層。回應於該等指令，顯示處理器114接著可執行各種表面組合函數，諸如，本揭示案中所描述之複合演算法(其可包括覆寫、具有恆定 α 摻合之色彩鍵控、不具有恆定 α 摻合之色彩鍵控、全表面恆定 α 摻合或全表面每像素 α 摻合。對於每一表面，*eglSwapBuffers*函數可執行複雜計算且設立由顯示處理器114在組合該等表面時使用之各種資料結構。為了準備此資料，*eglSwapBuffers*可遍曆重疊堆疊，自具有最低等級之層開始且以直至含有窗口表面之基礎層之次序進行。在每一層內，該函數可以每一表面結合至層之次序經由每一表面進行。接著，該函數可處理含有窗口表面(且在某些狀況下，其他表面)之基礎層(亦即，層0)。最後，該函數將繼續處理重疊層，自層等級1開始且以直至看來似乎最接近顯示器之觀察者之最高層之次序進行。以此方式，*eglSwapBuffers*系統地處理每一表面以便準備供顯示處理器114使用之資料。

上文所描述之裝置、方法及電腦程式產品可用於各種類型之器件中，諸如無線電話、蜂巢式電話、膝上型電腦、無線多媒體器件(例如，攜帶型視訊播放機或攜帶型視訊遊

戲器件)、無線通信個人電腦(PC)卡、個人數位助理(PDA)、外部或內部數據機或經由無線通道通信之任何器件。

該等器件可具有各種名稱，諸如存取終端機(AT)、存取單元、用戶單元、行動台、行動器件、行動單元、行動電話、手機、遠端台、遠端終端機、遠端單元、使用者器件、使用者設備、掌上型器件等等。

本揭示案中所描述之技術可在通用微處理器、數位信號處理器(DSP)、特殊應用積體電路(ASIC)、場可程式閘陣列(FPGA)或其他等效邏輯器件內實施。因此，如本文中所使用之術語"處理器"或"控制器"可指代上述結構之一或多者或其任何組合，以及適合於實施本文中所描述之技術之任何其他結構。此外，術語"處理器"或"控制器"亦可指代執行本文中所描述之技術之一或多個處理器或一或多個控制器。

本文中所描述之組件及技術可以硬體、軟體、韌體或其任何組合來實施。描述為模組或組件之任何特徵可一起實施於積體邏輯器件中或單獨作為離散但能共同操作之邏輯器件。在各種態樣中，該等組件可至少部分作為一或多個積體電路器件(其可共同地被稱作積體電路器件)來形成，諸如積體電路晶片或晶片組。該電路可提供於單一積體電路晶片器件中或多個能共同操作之積體電路晶片器件中，且可用於多種影像、顯示、音訊或其他多媒體應用及器件中之任一者中。在某些態樣中，(例如)該等組件可形成諸如無線通信器件手機之行動器件之一部分。

若以軟體來實施，則可至少部分藉由電腦可讀媒體來實現該等技術，該電腦可讀媒體包含當由一或多個處理器執行時執行上文所描述之方法之一或多者的指令。電腦可讀媒體可形成可包括封裝材料之電腦程式產品之一部分。電腦可讀媒體可包含諸如同步動態隨機存取記憶體(SDRAM)之隨機存取記憶體(RAM)、唯讀記憶體(ROM)、非揮發性隨機存取記憶體(NVRAM)、電可擦可程式唯讀記憶體(EEPROM)、嵌入式動態隨機存取記憶體(eDRAM)、靜態隨機存取記憶體(SRAM)、快閃記憶體、磁性或光學資料儲存媒體。

該等技術額外或替代性地可至少部分藉由電腦可讀通信媒體來實現，該電腦可讀通信媒體以指令或資料結構之形式載運或傳達程式碼且可由一或多個處理器存取、讀取及/或執行。可將任何連接適當地稱為電腦可讀媒體。舉例而言，若使用同軸電纜、光纜、雙絞線、數位用戶線(DSL)或無線技術(諸如，紅外線、無線電及微波)而自一網站、伺服器或其他遠端源傳輸軟體，則將同軸電纜、光纜、雙絞線、DSL或無線技術(諸如，紅外線、無線電及微波)包括於媒體之定義中。亦應將以上之組合包括於電腦可讀媒體之範疇內。所利用之任何軟體可由一或多個處理器來執行，諸如一或多個DSP、通用微處理器、ASIC、FPGA或其他等效積體或離散邏輯電路。

上文中已描述本揭示案之各種態樣。此等及其他態樣在以下申請專利範圍之範疇內。

【圖式簡單說明】

圖1為說明可用於將一組經描繪或預先描繪之圖形表面重疊至圖形圖框上之實例器件的方塊圖。

圖2為說明用於儲存於圖1之器件內的經描繪之表面的實例表面概況之方塊圖。

圖3A為說明可在圖1之器件內使用之實例重疊堆疊的方塊圖。

圖3B為說明可在圖1之器件內使用之另一實例重疊堆疊的方塊圖。

圖3C為說明可用於圖3B之重疊堆疊中之實例層的方塊圖。

圖4A為描繪重疊堆疊及在重疊層、下伏層與基礎層之間的關係的概念圖。

圖4B更詳細地說明用於圖4A之重疊堆疊中之實例層。

圖5A為說明圖1中所展示之API程式庫之其他細節的方塊圖。

圖5B為說明圖1中所展示之驅動程式之其他細節的方塊圖。

圖6為說明可用於將一組經描繪之圖形表面重疊或組合至圖形圖框上之另一實例器件的方塊圖。

圖7為說明具有2D圖形處理器與3D圖形處理器之可用於將一組經描繪之圖形表面重疊或組合至圖形圖框上的實例器件之方塊圖。

圖8為用於重疊或組合經描繪之圖形表面之方法的流程

圖。

圖9為用於重疊或組合經描繪之圖形表面之另一方法的
流程圖。

【主要元件符號說明】

100	器 件
102	圖 形 處 理 系 統
104	記 憶 體
106	顯 示 器 件
108	可 程 式 處 理 器 / 控 制 處 理 器
110	可 程 式 處 理 器 / 圖 形 處 理 器
112	表 面 緩 衝 器 / 緩 衝 器 空 間
114	可 程 式 處 理 器 / 顯 示 處 理 器
116	經 描 繪 之 圖 形 表 面
116A-116N	經 描 繪 之 圖 形 表 面
117	表 面 等 級
117A-117N	表 面 等 級
118	應 用 程 式 指 令
120	API程 式 庫
122	驅 動 程 式
124	表 面 資 訊
160	圖 框 緩 衝 器
200	表 面 概 況
202	經 描 繪 之 表 面 資 料
204	啟 用 旗 標

206	表面等級資料
208	複合表面資訊
300	重疊堆疊
302	窗口表面
304	啟用旗標
306	經描繪之表面資料
306A-306N	經描繪之表面資料
308	表面等級資料
308A-308N	表面等級資料
320	重疊堆疊
322	窗口表面
324	啟用旗標
326	經描繪之重疊層
326A-326N	經描繪之重疊層
328	下伏層
328A-328	N下伏層
340	實例層
342	層等級資料
344	啟用旗標
346	經描繪之表面資料
346A-346N	經描繪之表面資料
400	重疊堆疊
402	層
404	重疊層

406	下伏層
410	實例層
412	表面
414	表面
416	表面
418	表面
502	OpenGL ES描繪 API
504	OpenVG描繪 API
506	EGL API
508	下伏本地平台描繪 API
510	表面重疊 API
512	標準交換緩衝器 API
522	OpenGL ES描繪驅動程式
524	OpenVG描繪驅動程式
526	EGL驅動程式
528	下伏本地平台描繪驅動程式
530	表面重疊驅動程式
532	經修改之交換緩衝器驅動程式
600	器件
602	圖形處理系統
604	記憶體
606	顯示器件
608	控制處理器
610	圖形處理器

612	表面緩衝器
614	顯示處理器
618	應用程式指令
620	API程式庫
622	驅動程式
624	表面資訊
660	圖框緩衝器
662	圖元處理單元
664	像素處理單元
668	覆寫區塊
670	摻合單元
672	色彩鍵控區塊
674	組合的色彩鍵控 α 摻合區塊
700	器件
702	圖形處理系統
704	記憶體
706	顯示器件
708	控制處理器
710	3D圖形處理器
711	2D圖形處理器
712	3D表面緩衝器
713	2D表面緩衝器
714	顯示處理器
718	應用程式指令

720	API程式庫
722	驅動程式
724	表面資訊
760	圖框緩衝器

五、中文發明摘要：

整體而言，本發明揭示用於將一組經描繪之圖形表面重疊或組合至一單一圖形圖框上的各種技術。一實例器件包括一第一處理器，其在該器件將複數個經描繪之圖形表面中之任一者輸出至一顯示器之前選擇用於該等經描繪之圖形表面中之每一者的一表面等級。該器件進一步包括一第二處理器，其檢索該等經描繪之圖形表面，根據該等選定之表面等級中之每一者將該等經描繪之圖形表面重疊至一圖形圖框上，且將該圖形圖框輸出至該顯示器。

六、英文發明摘要：

In general, the present disclosure describes various techniques for overlaying or combining a set of rendered graphics surfaces onto a single graphics frame. One example device includes a first processor that selects a surface level for each of a plurality of rendered graphics surfaces prior to the device outputting any of the rendered graphics surfaces to a display. The device further includes a second processor that retrieves the rendered graphics surfaces, overlays the rendered graphics surfaces onto a graphics frame in accordance with each of the selected surface levels, and outputs the graphics frame to the display.

十、申請專利範圍：

1. 一種器件，其包含：

一第一處理器，其在該器件將複數個經描繪之圖形表面中之任一者輸出至一顯示器之前選擇用於該等經描繪之圖形表面中之每一者之一表面等級；及

一第二處理器，其檢索該等經描繪之圖形表面，根據該等選定之表面等級中之每一者將該等經描繪之圖形表面重疊至一圖形圖框上，且將該圖形圖框輸出至該顯示器。

2. 如請求項1之器件，其中：

該第一處理器及該第二處理器為單獨的處理器，且

該第一處理器及該第二處理器各自選自由以下各物組成之群：一圖形處理器、一顯示處理器及一控制處理器。

3. 如請求項1之器件，其進一步包含：

一或多個表面緩衝器；及

一第三處理器，其描繪該等經描繪之圖形表面，且將該等經描繪之圖形表面儲存於該一或多個表面緩衝器中，

其中該第二處理器自該一或多個表面緩衝器中檢索該等經描繪之圖形表面。

4. 如請求項3之器件，其中：

該第三處理器包含一包含一加速三維(3D)圖形管道之圖形處理器，且

該第一處理器包含一包含一通用處理管道之控制處理

器。

5. 如請求項3之器件，其中：

該等經描繪之圖形表面包含一第一組經描繪之圖形表面，

該第一處理器描繪一第二組經描繪之圖形表面，且在該器件將該等經描繪之圖形表面中之任一者輸出至該顯示器之前選擇用於該第二組經描繪之圖形表面中的該等經描繪之圖形表面中之每一者的一表面等級，且

該第二處理器根據該等選定之表面等級將該第二組經描繪之圖形表面重疊至該圖形圖框上。

6. 如請求項5之器件，其中：

該第一組經描繪之圖形表面中的該等經描繪之圖形表面之至少一者包含一螢幕上表面，且

該第二組經描繪之圖形表面中的該等經描繪之圖形表面中之每一者包含一螢幕外表面。

7. 如請求項5之器件，其中：

該第一組經描繪之圖形表面中的該等經描繪之圖形表面中之每一者包含一三維(3D)表面，且

該第二組經描繪之圖形表面中的該等經描繪之圖形表面之至少一者包含一選自由以下各物組成之群之表面：一二維(2D)表面及一視訊表面。

8. 如請求項3之器件，其中：

該第三處理器包含一在圖元上執行操作之圖元處理單元，及一執行像素操作之像素處理單元，且

該第二處理器執行該等像素操作之一有限集合。

9. 如請求項8之器件，其中由該第二處理器執行之獨特像素操作之一數目小於由該像素處理單元執行之獨特像素操作之一數目。

10. 如請求項1之器件，其中：

該第二處理器包含一覆寫區塊，且

當該等經描繪之圖形表面重疊時，該覆寫區塊選擇該等經描繪之圖形表面之具有一最高表面等級的一者，且格式化該圖形圖框以便對於該等經描繪之圖形表面之重疊部分顯示該等經描繪之圖形表面之該選定之一者。

11. 如請求項1之器件，其中：

該第二處理器至少部分藉由根據一或多個複合模式組合該等經描繪之圖形表面而將該等經描繪之圖形表面重疊至該圖形圖框上，其中該一或多個複合模式包括以下之至少一者：具有恆定 α 摻合之色彩鍵控、不具有恆定 α 摻合之色彩鍵控、全表面恆定 α 摻合或全表面每像素 α 摻合。

12. 如請求項1之器件，其中：

該第一處理器產生一具有複數個層之重疊堆疊，且將該等經描繪之圖形表面中之每一者結合至該重疊堆疊內之個別層，

該第二處理器根據該重疊堆疊將該等經描繪之圖形表面重疊至該圖形圖框上。

13. 如請求項12之器件，其中：

該重疊堆疊中之該等層中之每一者具有一獨特層等級，且

當將該圖形圖框顯示於該顯示器上時，結合至該重疊堆疊內之一具有一第一層等級之層的經描繪之圖形表面看來似乎比結合至具有小於該第一層等級之層等級之層的經描繪之圖形表面較接近該顯示器之一觀察者。

14. 如請求項12之器件，其中：

該第一處理器對於待顯示之每一圖形圖框選擇性地啟用及停用該重疊堆疊內的個別層，

當對於一第一圖形圖框啟用該重疊堆疊內之一第一層時，該第二處理器處理結合至該第一層之該等經描繪之圖形表面中的每一者以產生該第一圖形圖框，且

當對於該第一圖形圖框停用該第一層時，該第二處理器不處理結合至該第一層之任何經描繪之圖形表面以產生該第一圖形圖框。

15. 如請求項1之器件，其中：

該第一處理器對於待顯示之每一圖形圖框選擇性地啟用及停用個別經描繪之圖形表面；

當對於一第一圖形圖框啟用該複數個經描繪之圖形表面內的一第一經描繪之圖形表面時，該第二處理器處理該第一經描繪之圖形表面以產生該第一圖形圖框；

當對於該第一圖形圖框停用該第一經描繪之圖形表面時，該第二處理器不處理該第一經描繪之圖形表面以產生該第一圖形圖框。

16. 如請求項1之器件，其中該等經描繪之圖形表面包含一第一組經描繪之圖形表面，該器件進一步包含：

一第一表面緩衝器；

一第二表面緩衝器；

一第一圖形處理器，其產生該第一組經描繪之圖形表面，且將該第一組經描繪之圖形表面儲存於該第一表面緩衝器中；及

一第二圖形處理器，其產生一第二組經描繪之圖形表面，且將該第二組經描繪之圖形表面儲存於該第二表面緩衝器中，

其中該第一處理器在該器件將該等經描繪之圖形表面中之任一者輸出至該顯示器之前選擇用於該第二組經描繪之圖形表面中的該等經描繪之圖形表面中之每一者之一表面等級，且

其中該第二處理器自該第一表面緩衝器中檢索該第一組經描繪之圖形表面，自該第二表面緩衝器中檢索該第二組經描繪之圖形表面，且根據該等選定之表面等級中之每一者將該第一組經描繪之圖形表面及該第二組經描繪之圖形表面重疊至該圖形圖框上。

17. 如請求項16之器件，其中：

該第一圖形處理器包含一加速三維(3D)圖形管道；且

該第二圖形處理器包含一加速二維(2D)圖形管道。

18. 如請求項1之器件，其中該器件包含一無線通信器件手機。

19. 如請求項1之器件，其中該器件包含一或多個積體電路器件。

20. 一種方法，其包含：

檢索複數個經描繪之圖形表面；

在將該等經描繪之圖形表面中之任一者輸出至一顯示器之前選擇用於該等經描繪之圖形表面中之每一者之一表面等級；

根據該等選定之表面等級中之每一者將該等經描繪之圖形表面重疊至一圖形圖框上；及

將該圖形圖框輸出至該顯示器。

21. 如請求項20之方法，其進一步包含：

產生該等經描繪之圖形表面；及

將該等經描繪之圖形表面儲存於一或多個表面緩衝器中，

其中檢索該複數個經描繪之圖形表面包含自該一或多個表面緩衝器中檢索該複數個經描繪之圖形表面。

22. 如請求項20之方法，其中該等經描繪之圖形表面包含第一組經描繪之圖形表面，且其中該方法進一步包含：

用一第一處理器產生該第一組經描繪之圖形表面；

用一第二處理器產生第二組經描繪之圖形表面，其中該第一處理器及該第二處理器為單獨處理器；

選擇用於該第二組經描繪之圖形表面中之每一經描繪之圖形表面之一表面等級；及

根據該等選定之表面等級將該第二組經描繪之圖形表

面重疊至該圖形圖框上。

23. 如請求項22之方法，其中：

該第一組經描繪之圖形表面中的該等經描繪之圖形表面之至少一者包含一螢幕上表面，且

該第二組經描繪之圖形表面中的該等經描繪之圖形表面中之每一者包含一螢幕外表面。

24. 如請求項22之方法，其中：

該第一組經描繪之圖形表面中的該等經描繪之圖形表面中之每一者包含一三維(3D)表面，且

該第二組經描繪之圖形表面中的該等經描繪之圖形表面之至少一者包含一選自由以下各物組成之群之表面：
一二維(2D)表面及一視訊表面。

25. 如請求項20之方法，其中根據該等選定之表面等級中之每一者將該等經描繪之圖形表面重疊至該圖形圖框上包含：

當該等經描繪之圖形表面重疊時，選擇該等經描繪之圖形表面之具有一最高表面等級的一者；及

格式化該圖形圖框以便對於該等經描繪之圖形表面之重疊部分顯示該等經描繪之圖形表面之該選定之一者。

26. 如請求項20之方法，其中根據該等選定之表面等級中之每一者將該等經描繪之圖形表面重疊至該圖形圖框上包含：

根據一或多個複合模式組合該等經描繪之圖形表面，
其中該一或多個複合模式包括以下之至少一者：具有恆

定 α 摻合之色彩鍵控、不具有恆定 α 摻合之色彩鍵控、全表面恆定 α 摻合或全表面每像素 α 摻合。

27. 如請求項20之方法，其進一步包含：

產生一具有複數個層之重疊堆疊；

將該等經描繪之圖形表面中之每一者結合至該重疊堆疊內之個別層；

根據該重疊堆疊將該等經描繪之圖形表面重疊至該圖形圖框上。

28. 如請求項27之方法，其中該重疊堆疊中之該等層中之每一者具有一獨特層等級，且其中根據該重疊堆疊將該等經描繪之圖形表面重疊至該圖形圖框上包含：

格式化該圖形圖框以使得當將該圖形圖框顯示於該顯示器上時，結合至該重疊堆疊內之一具有一第一層等級之層的經描繪之圖形表面看來似乎比結合至具有小於該第一層等級之層等級之層的經描繪之圖形表面較接近該顯示器之一觀察者。

29. 如請求項27之方法，其進一步包含：

對於待顯示之每一圖形圖框，選擇性地啟用該重疊堆疊內的個別層；

當對於一第一圖形圖框啟用該重疊堆疊內之一第一層時，處理結合至該第一層之該等經描繪之圖形表面中的每一者以產生該第一圖形圖框；及

當對於該第一圖形圖框停用該第一層時，不處理結合至該第一層之任何經描繪之圖形表面以產生該第一圖形

圖框。

30. 如請求項27之方法，其進一步包含：

對於待顯示之每一圖形圖框，選擇性地啟用個別經描繪之圖形表面；

當對於一第一圖形圖框啟用一第一經描繪之圖形表面時，處理一第一經描繪之圖形表面以產生一第一圖形圖框；

當對於該第一圖形圖框停用該第一經描繪之圖形表面時，不處理該第一經描繪之圖形表面以產生該第一圖形圖框。

31. 如請求項20之方法，其中該等經描繪之圖形表面包含一第一組經描繪之圖形表面，且其中該方法進一步包含：

用一第一圖形處理器產生該第一組經描繪之圖形表面；

用一第二圖形處理器產生一第二組經描繪之圖形表面；

將該第一組經描繪之圖形表面儲存於一第一表面緩衝器中；

將該第二組經描繪之圖形表面儲存於一第二表面緩衝器中；

在將該等經描繪之圖形表面中之任一者輸出至該顯示器之前選擇用於該第二組經描繪之圖形表面內的每一經描繪之圖形表面之一表面等級；

自該第一表面緩衝器中檢索該第一組經描繪之圖形表

面；

自該第二表面緩衝器中檢索該第二組經描繪之圖形表面；及

根據該等選定之表面等級中之每一者將該第一組經描繪之圖形表面及該第二組經描繪之圖形表面重疊至該圖形圖框上。

32. 如請求項31之方法，其中：

該第一圖形處理器包含一加速三維(3D)圖形管道；且

該第二圖形處理器包含一加速二維(2D)圖形管道。

33. 一種器件，其包含：

用於檢索複數個經描繪之圖形表面之構件；

用於在將該等經描繪之圖形表面中之任一者輸出至一顯示器之前選擇用於該等經描繪之圖形表面中之每一者的一表面等級的構件；

用於根據該等選定之表面等級中之每一者將該等經描繪之圖形表面重疊至一圖形圖框上的構件；及

用於將該圖形圖框輸出至該顯示器之構件。

34. 如請求項33之器件，其進一步包含：

用於產生該等經描繪之圖形表面之構件；及

用於將該等經描繪之圖形表面儲存於一或多個表面緩衝器中之構件，

其中用於檢索該複數個經描繪之圖形表面之該構件包含用於自該一或多個表面緩衝器中檢索該複數個經描繪之圖形表面之構件。

35. 如請求項33之器件，其中該等經描繪之圖形表面包含一第一組經描繪之圖形表面，且其中該器件進一步包含：

用於用一第一處理器產生該第一組經描繪之圖形表面之構件；

用於用一第二處理器產生一第二組經描繪之圖形表面之構件，其中該第一處理器及該第二處理器為單獨處理器；

用於選擇用於該第二組經描繪之圖形表面中之每一經描繪之圖形表面的一表面等級的構件；及

用於根據該等選定之表面等級將該第二組經描繪之圖形表面重疊至該圖形圖框上的構件。

36. 如請求項35之器件，其中：

該第一組經描繪之圖形表面中的該等經描繪之圖形表面之至少一者包含一螢幕上表面，且

該第二組經描繪之圖形表面中的該等經描繪之圖形表面中之每一者包含一螢幕外表面。

37. 如請求項35之器件，其中：

該第一組經描繪之圖形表面中的該等經描繪之圖形表面中之每一者包含一三維(3D)表面，且

該第二組經描繪之圖形表面中的該等經描繪之圖形表面之至少一者包含一選自由以下各物組成之群之表面：一二維(2D)表面及一視訊表面。

38. 如請求項33之器件，其中用於根據該等選定之表面等級中之每一者將該等經描繪之圖形表面重疊至該圖形圖框

上的該構件包含：

用於當該等經描繪之圖形表面重疊時選擇該等經描繪之圖形表面之具有一最高表面等級的一者的構件；及

用於格式化該圖形圖框以便對於該等經描繪之圖形表面之重疊部分顯示該等經描繪之圖形表面之該選定之一者的構件。

39. 如請求項33之器件，其中用於根據該等選定之表面等級中之每一者將該等經描繪之圖形表面重疊至該圖形圖框上的該構件包含：

用於根據一或多個複合模式組合該等經描繪之圖形表面之構件，其中該一或多個複合模式包括以下之至少一者：具有恆定 α 摻合之色彩鍵控、不具有恆定 α 摻合之色彩鍵控、全表面恆定 α 摻合或全表面每像素 α 摻合。

40. 如請求項33之器件，其進一步包含：

用於產生一具有複數個層之重疊堆疊之構件；

用於將該等經描繪之圖形表面中之每一者結合至該重疊堆疊內之個別層的構件及

用於根據該重疊堆疊將該等經描繪之圖形表面重疊至該圖形圖框上之構件。

41. 如請求項40之器件，其中該重疊堆疊中之該等層中之每一者具有一獨特層等級，且其中用於根據該重疊堆疊將該等經描繪之圖形表面重疊至該圖形圖框上之該構件包含：

用於執行以下之構件：格式化該圖形圖框以使得當將

該圖形圖框顯示於該顯示器上時，結合至該重疊堆疊內之一具有一第一層等級之層的經描繪之圖形表面看來似乎比結合至具有小於該第一層等級之層等級之層的經描繪之圖形表面較接近該顯示器之一觀察者。

42. 如請求項40之器件，其進一步包含：

用於對於待顯示之每一圖形圖框選擇性地啟用該重疊堆疊內的個別層的構件；及

用於執行以下之構件：當對於一第一圖形圖框啟用該重疊堆疊內之一第一層時，處理結合至該第一層之該等經描繪之圖形表面中的每一者以產生該第一圖形圖框，且當對於該第一圖形圖框停用該第一層時，不處理結合至該第一層之任何經描繪之圖形表面以產生該第一圖形圖框。

43. 如請求項33之器件，其進一步包含：

用於對於待顯示之每一圖形圖框選擇性地啟用個別經描繪之圖形表面的構件；及

用於執行以下之構件：當對於一第一圖形圖框啟用一第一經描繪之圖形表面時，處理一第一經描繪之圖形表面以產生一第一圖形圖框，且當對於該第一圖形圖框停用該第一經描繪之圖形表面時，不處理該第一經描繪之圖形表面以產生該第一圖形圖框。

44. 如請求項33之器件，其中該等經描繪之圖形表面包含一第一組經描繪之圖形表面，且其中該器件進一步包含：

用於用一第一圖形處理器產生該第一組經描繪之圖形

表面之構件；

用於用一第二圖形處理器產生一第二組經描繪之圖形表面之構件；

用於將該第一組經描繪之圖形表面儲存於一第一表面緩衝器中之構件；

用於將該第二組經描繪之圖形表面儲存於一第二表面緩衝器中之構件；

用於在該器件將該等經描繪之圖形表面中之任一者輸出至該顯示器之前選擇用於該第二組經描繪之圖形表面內的每一經描繪之圖形表面之一表面等級的構件；

用於自該第一表面緩衝器中檢索該第一組經描繪之圖形表面之構件；

用於自該第二表面緩衝器中檢索該第二組經描繪之圖形表面之構件；及

用於根據該等選定之表面等級中之每一者將該第一組經描繪之圖形表面及該第二組經描繪之圖形表面重疊至該圖形圖框上的構件。

45. 如請求項44之器件，其中：

該第一圖形處理器包含一加速三維(3D)圖形管道；且

該第二圖形處理器包含一加速二維(2D)圖形管道。

46. 一種電腦可讀媒體，其包含造成一或多個處理器執行以下操作之指令：

檢索複數個經描繪之圖形表面；

在將該等經描繪之圖形表面中之任一者輸出至一顯示

器之前選擇用於該等經描繪之圖形表面中之每一者之一表面等級；

根據該等選定之表面等級中之每一者將該等經描繪之圖形表面重疊至一圖形圖框上；及

將該圖形圖框輸出至該顯示器。

47. 如請求項46之電腦可讀媒體，其進一步包含造成該一或多個處理器執行以下操作之指令：

產生該等經描繪之圖形表面；及

將該等經描繪之圖形表面儲存於一或多個表面緩衝器中，

其中造成該一或多個處理器檢索該複數個經描繪之圖形表面的該等指令包含造成該一或多個處理器自該一或多個表面緩衝器中檢索該複數個經描繪之圖形表面的指令。

48. 如請求項46之電腦可讀媒體，其中該等經描繪之圖形表面包含一第一組經描繪之圖形表面，且其中該電腦可讀媒體進一步包含造成該一或多個處理器執行以下操作之指令：

產生該第一組經描繪之圖形表面；

產生一第二組經描繪之圖形表面；

選擇用於該第二組經描繪之圖形表面中之每一經描繪之圖形表面之一表面等級；及

根據該等選定之表面等級將該第一組經描繪之圖形表面及該第二組經描繪之圖形表面重疊至該圖形圖框上。

49. 如請求項48之電腦可讀媒體，其中：

該第一組經描繪之圖形表面中的該等經描繪之圖形表面之至少一者包含一螢幕上表面，且

該第二組經描繪之圖形表面中的該等經描繪之圖形表面中之每一者包含一螢幕外表面。

50. 如請求項48之電腦可讀媒體，其中：

該第一組經描繪之圖形表面中的該等經描繪之圖形表面中之每一者包含一三維(3D)表面，且

該第二組經描繪之圖形表面中的該等經描繪之圖形表面之至少一者包含一選自由以下各物組成之群之表面：一二維(2D)表面及一視訊表面。

51. 如請求項46之電腦可讀媒體，其中造成該一或多個處理器根據該等選定之表面等級中之每一者將該等經描繪之圖形表面重疊至該圖形圖框上的該等指令包含造成該一或多個處理器執行以下操作之指令：

當該等經描繪之圖形表面重疊時，選擇該等經描繪之圖形表面之具有一最高表面等級的一者；及

格式化該圖形圖框以便對於該等經描繪之圖形表面之重疊部分顯示該等經描繪之圖形表面之該選定之一者。

52. 如請求項46之電腦可讀媒體，其中造成該一或多個處理器根據該等選定之表面等級中之每一者將該等經描繪之圖形表面重疊至該圖形圖框上的該等指令包含造成該一或多個處理器執行以下操作之指令：

根據一或多個複合模式組合該等經描繪之圖形表面，

其中該一或多個複合模式包括以下之至少一者：具有恆定 α 摻合之色彩鍵控、不具有恆定 α 摻合之色彩鍵控、全表面恆定 α 摻合或全表面每像素 α 摻合。

53. 如請求項46之電腦可讀媒體，其進一步包含造成該一或多個處理器執行以下操作之指令：

產生一具有複數個層之重疊堆疊；

將該等經描繪之圖形表面中之每一者結合至該重疊堆疊內之個別層；及

根據該重疊堆疊將該等經描繪之圖形表面重疊至該圖形圖框上。

54. 如請求項53之電腦可讀媒體，其中該重疊堆疊中之該等層中之每一者具有一獨特層等級，且其中造成該一或多個處理器根據該重疊堆疊將該等經描繪之圖形表面重疊至該圖形圖框上之該等指令包含造成該一或多個處理器執行以下操作之指令：

格式化該圖形圖框以使得當將該圖形圖框顯示於該顯示器上時，結合至該重疊堆疊內之一具有一第一層等級之層的經描繪之圖形表面看來似乎比結合至具有小於該第一層等級之層等級之層的經描繪之圖形表面較接近該顯示器之一觀察者。

55. 如請求項53之電腦可讀媒體，其進一步包含造成該一或多個處理器執行以下操作之指令：

對於待顯示之每一圖形圖框，選擇性地啟用該重疊堆疊內的個別層；

當對於一第一圖形圖框啟用該重疊堆疊內之一第一層時，處理結合至該第一層之該等經描繪之圖形表面中的每一者以產生該第一圖形圖框；及

當對於該第一圖形圖框停用該第一層時，不處理結合至該第一層之任何經描繪之圖形表面以產生該第一圖形圖框。

56. 如請求項46之電腦可讀媒體，其進一步包含造成該一或多個處理器執行以下操作之指令：

對於待顯示之每一圖形圖框，選擇性地啟用個別經描繪之圖形表面；

當對於一第一圖形圖框啟用一第一經描繪之圖形表面時，處理一第一經描繪之圖形表面以產生一第一圖形圖框；及

當對於該第一圖形圖框停用該第一經描繪之圖形表面時，不處理該第一經描繪之圖形表面以產生該第一圖形圖框。

57. 如請求項46之電腦可讀媒體，其中該等經描繪之圖形表面包含一第一組經描繪之圖形表面，且其中該電腦可讀媒體進一步包含造成該一或多個處理器執行以下操作之指令：

用一第一圖形處理器產生該第一組經描繪之圖形表面；

用一第二圖形處理器產生一第二組經描繪之圖形表面；

將該第一組經描繪之圖形表面儲存於一第一表面緩衝器中；

將該第二組經描繪之圖形表面儲存於一第二表面緩衝器中；

在該器件將該等經描繪之圖形表面中之任一者輸出至該顯示器之前選擇用於該第二組經描繪之圖形表面內的每一經描繪之圖形表面之一表面等級；

自該第一表面緩衝器中檢索該第一組經描繪之圖形表面；

自該第二表面緩衝器中檢索該第二組經描繪之圖形表面；及

根據該等選定之表面等級中之每一者將該第一組經描繪之圖形表面及該第二組經描繪之圖形表面重疊至該圖形圖框上。

58. 如請求項57之電腦可讀媒體，其中：

該第一圖形處理器包含一加速三維(3D)圖形管道；且

該第二圖形處理器包含一加速二維(2D)圖形管道。

十一、圖式：

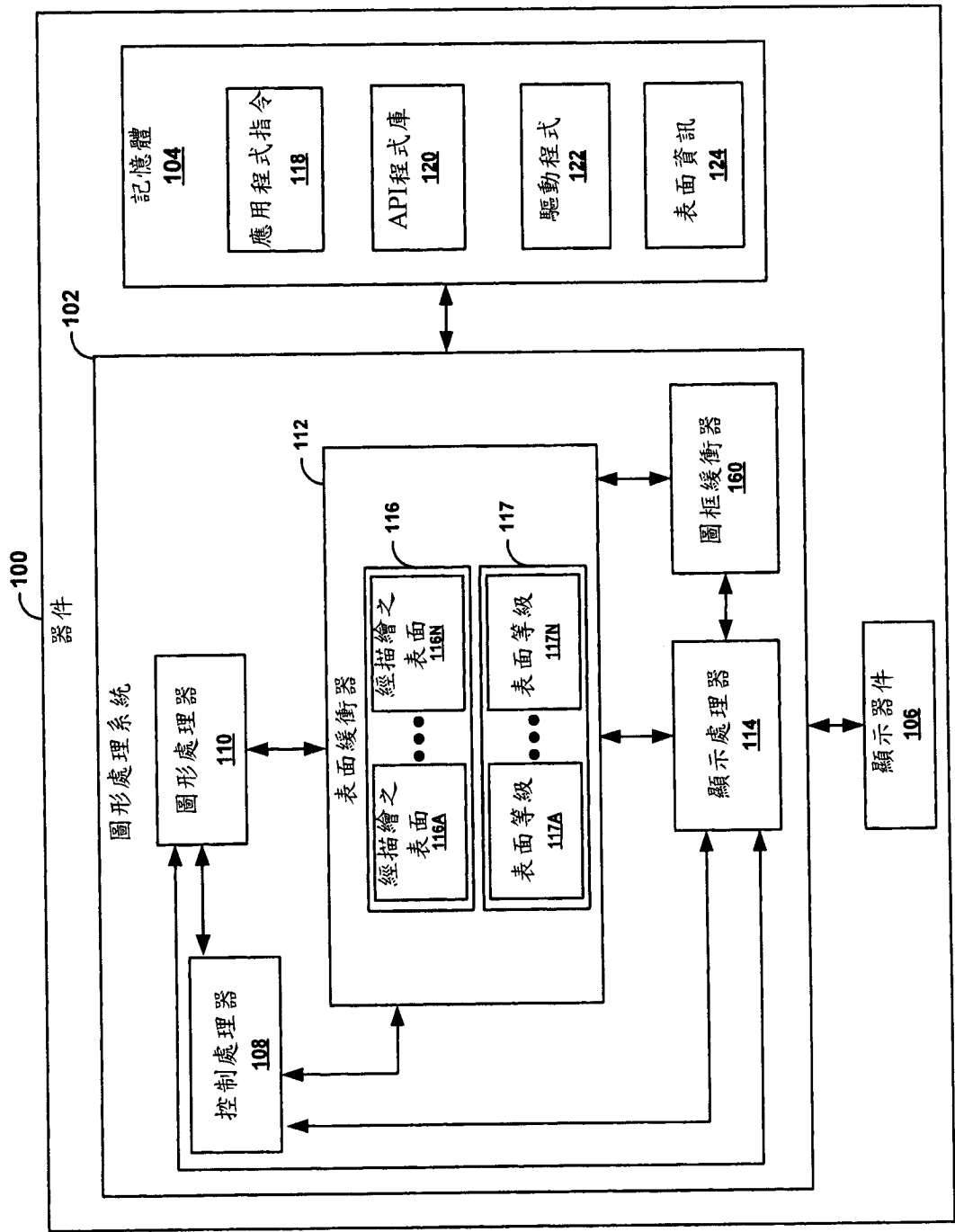


圖1

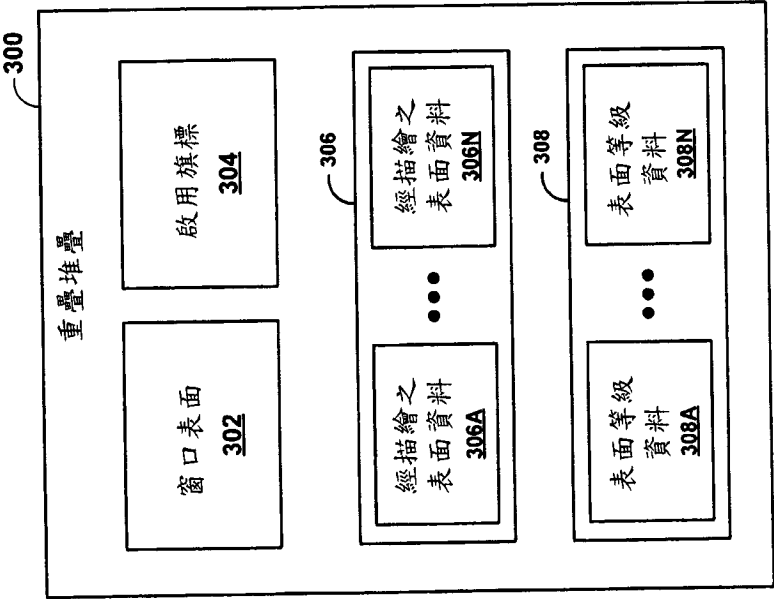


圖3A

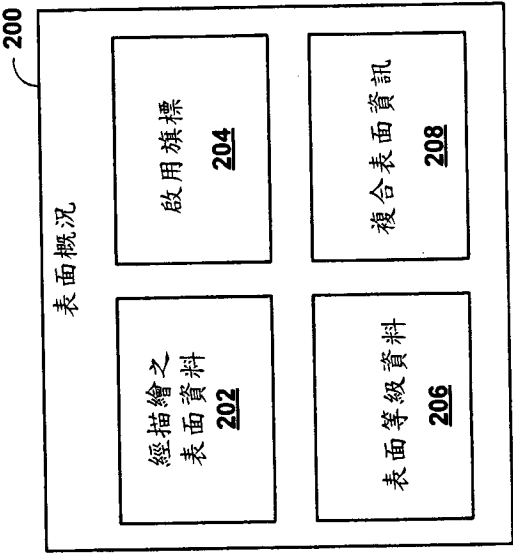


圖2

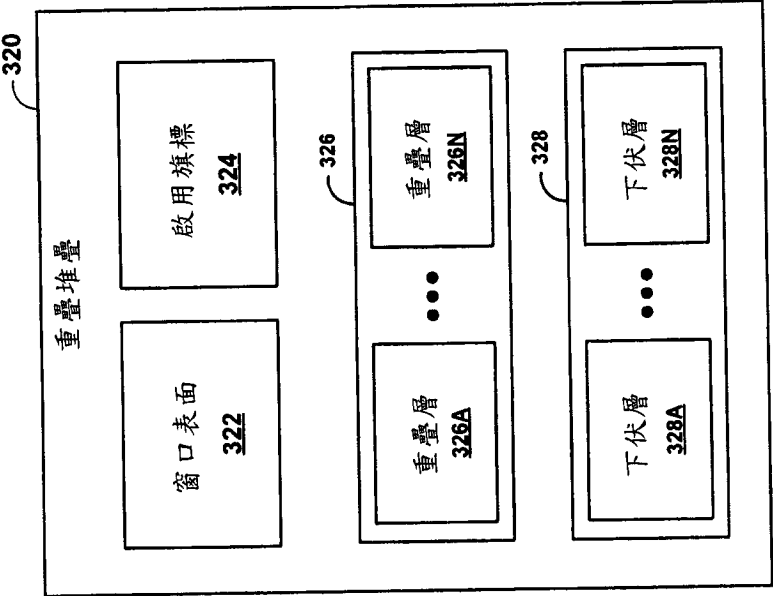


圖3B

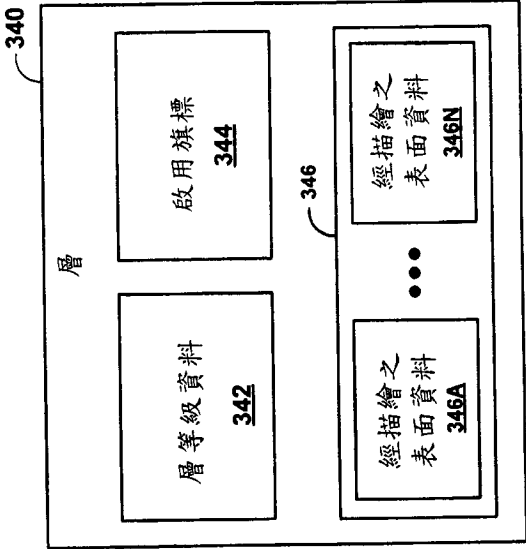


圖3C

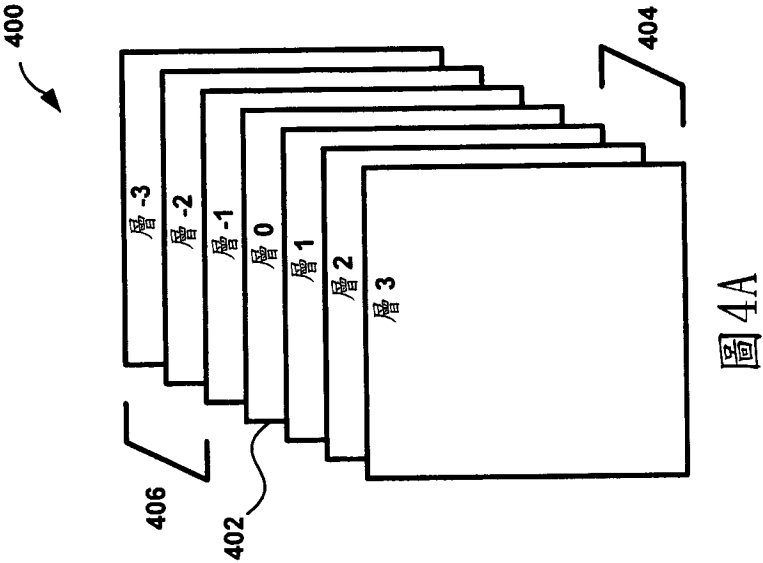


圖 4A

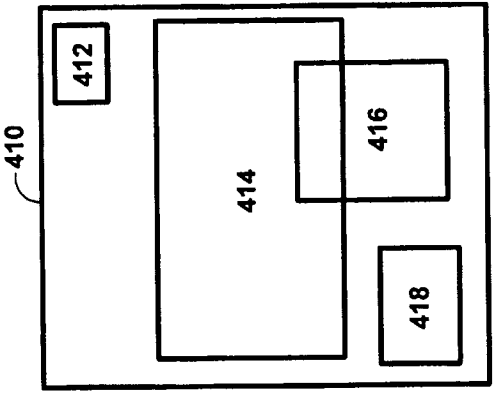


圖 4B

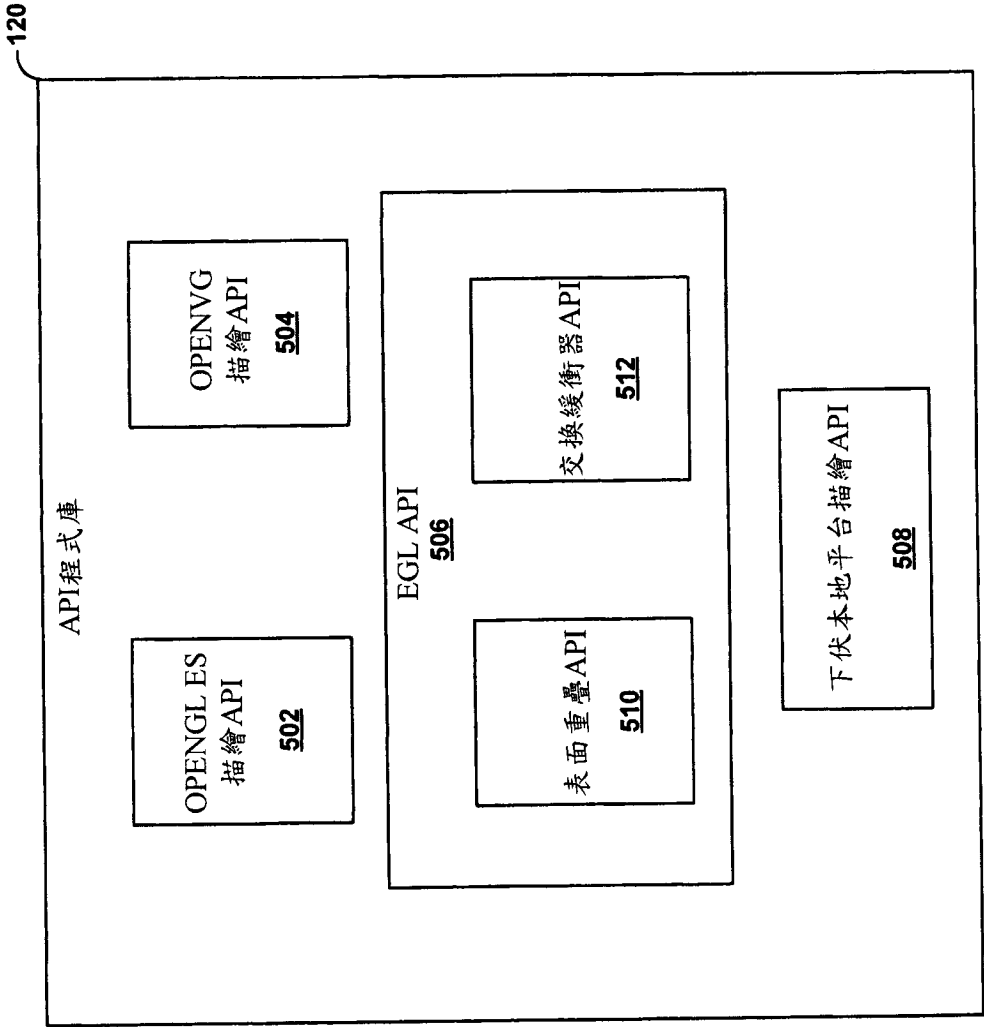


圖5A

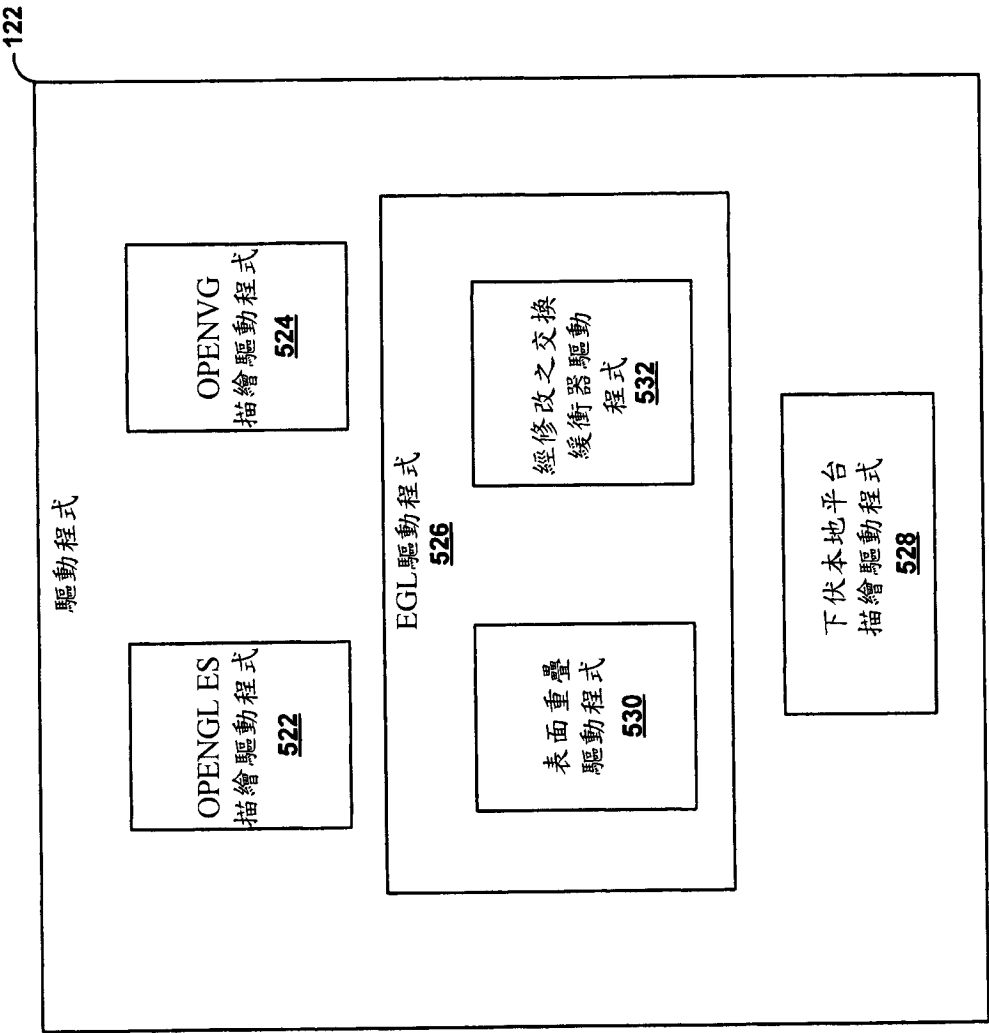


圖5B

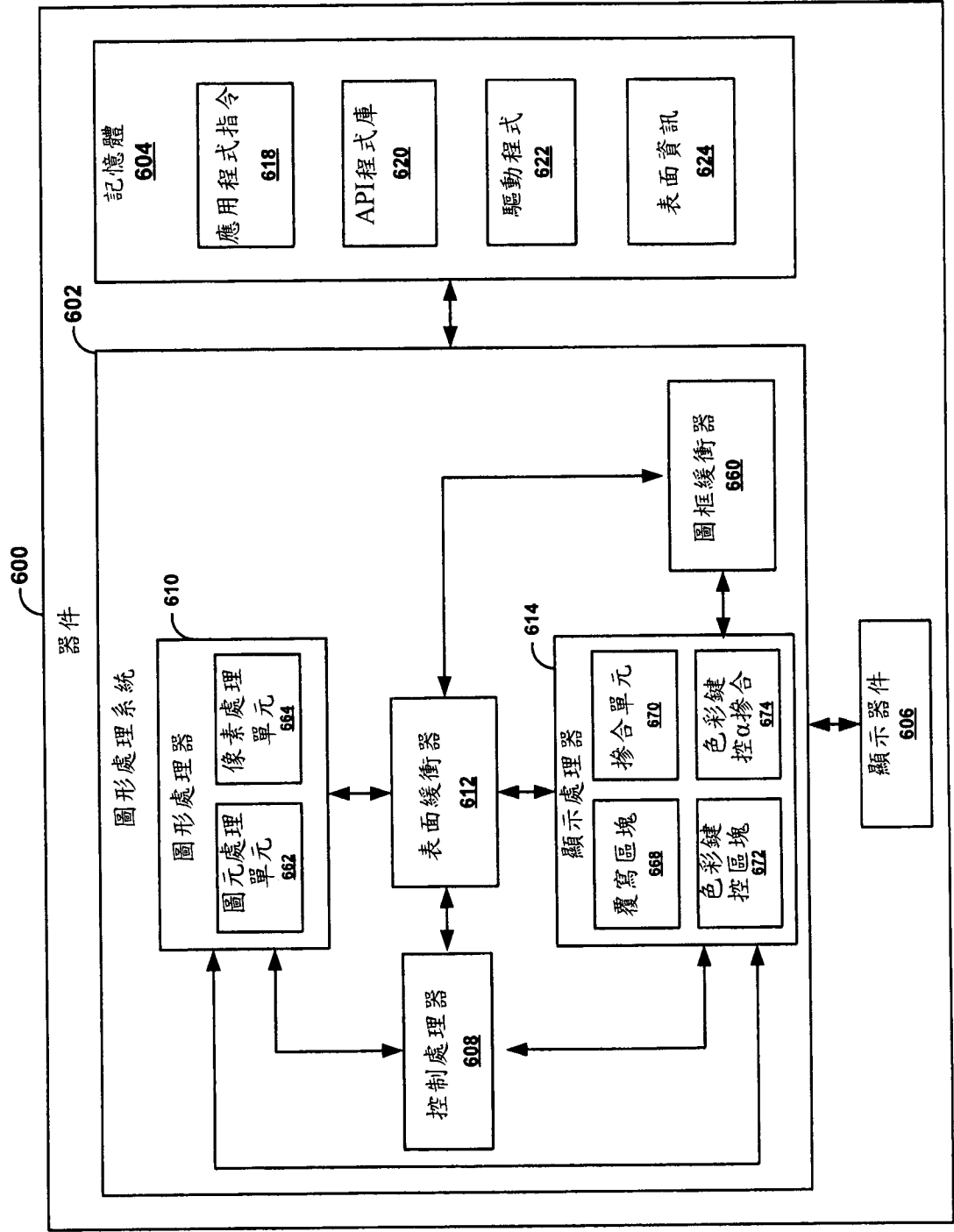


圖6

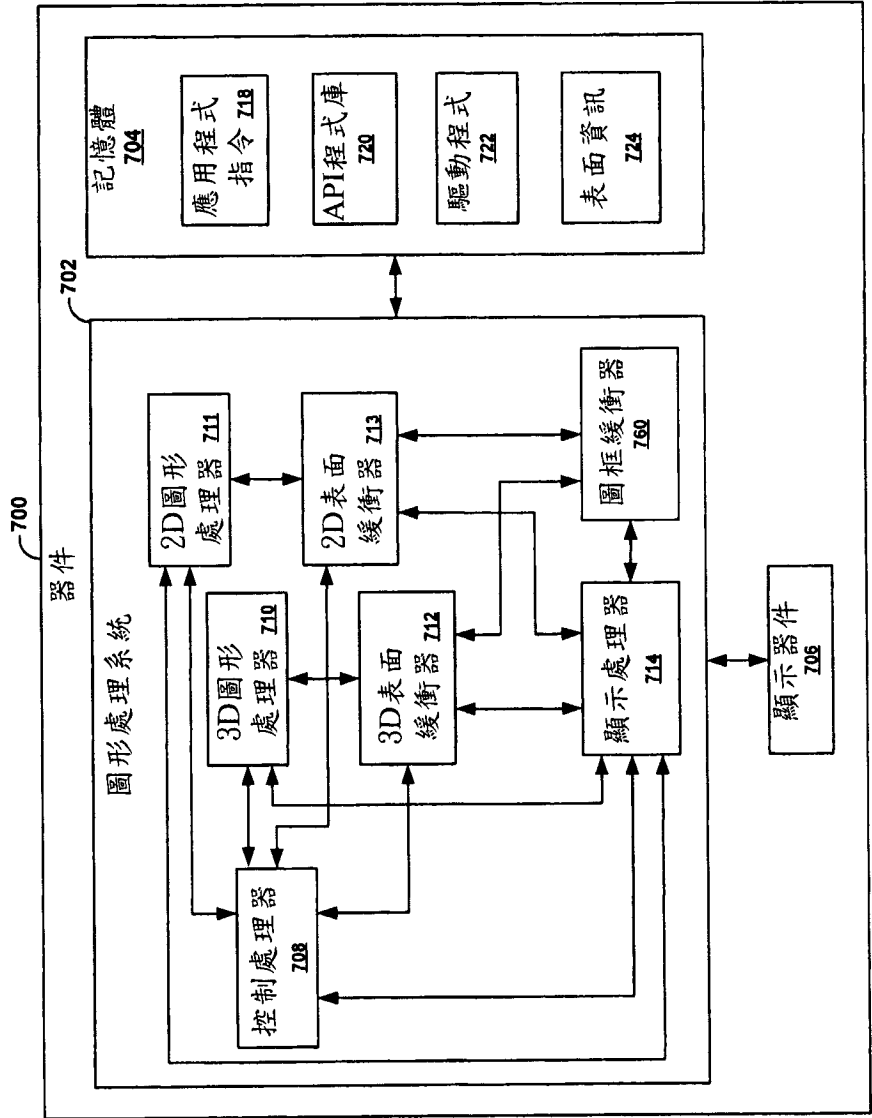


圖7

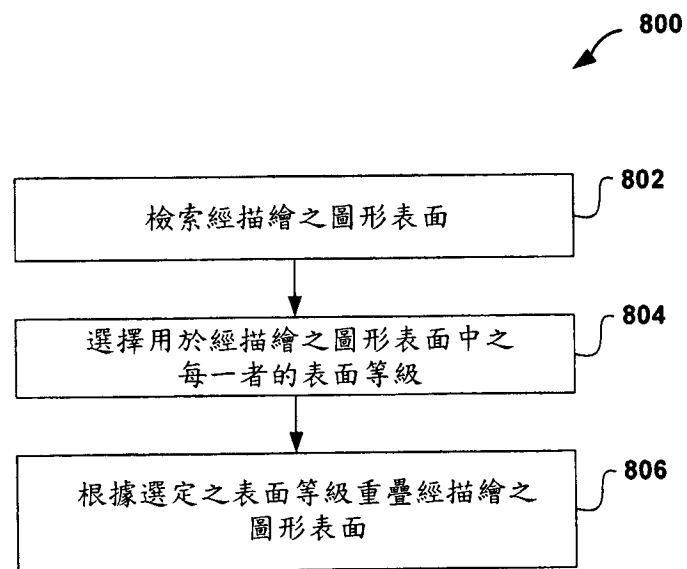


圖8

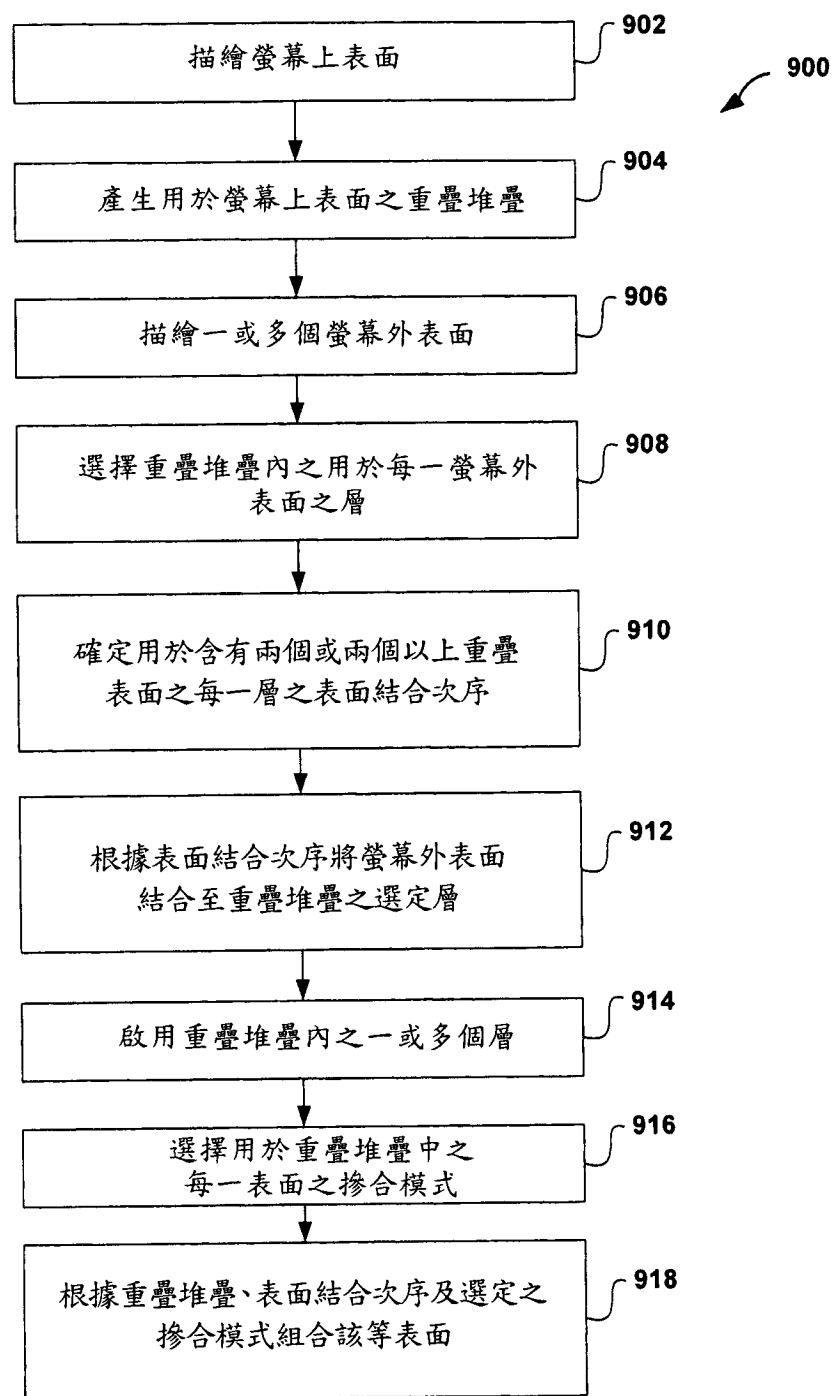


圖9

七、指定代表圖：

(一)本案指定代表圖為：第（ 1 ）圖。

(二)本代表圖之元件符號簡單說明：

100	器 件
102	圖 形 處 理 系 統
104	記 憶 體
106	顯 示 器 件
108	可 程 式 處 理 器 / 控 制 處 理 器
110	可 程 式 處 理 器 / 圖 形 處 理 器
112	表 面 緩 衝 器 / 緩 衝 器 空 間
114	可 程 式 處 理 器 / 顯 示 處 理 器
116	經 描 繪 之 圖 形 表 面
116A-116N	經 描 繪 之 圖 形 表 面
117	表 面 等 級
117A-117N	表 面 等 級
118	應 用 程 式 指 令
120	API程 式 庫
122	驅 動 程 式
124	表 面 資 訊
160	圖 框 緩 衝 器

八、本案若有化學式時，請揭示最能顯示發明特徵的化學式：

（ 無 ）