

【公報種別】特許法第 17 条の 2 の規定による補正の掲載

【部門区分】第 6 部門第 3 区分

【発行日】平成30年5月31日(2018.5.31)

【公開番号】特開2015-69638(P2015-69638A)

【公開日】平成27年4月13日(2015.4.13)

【年通号数】公開・登録公報2015-024

【出願番号】特願2014-89899(P2014-89899)

【国際特許分類】

G 0 6 F 8/70 (2018.01)

G 0 6 T 11/00 (2006.01)

【F I】

G 0 6 F 9/06 6 2 0 K

G 0 6 T 11/00 1 0 0 A

【誤訳訂正書】

【提出日】平成30年4月11日(2018.4.11)

【誤訳訂正 1】

【訂正対象書類名】明細書

【訂正対象項目名】全文

【訂正方法】変更

【訂正の内容】

【発明の詳細な説明】

【発明の名称】グラフィカル画像データをキャプチャするソフトウェア開発キット

【関連出願の相互参照】

【0 0 0 1】

本出願は、2013年2月13日に提出された米国特許出願第 1 3 / 7 6 5 , 7 0 3 号、発明の名称「System and Method for Sharing Gameplay Experiences」に関連し、該出願の開示は、参照することによりその全体が組み込まれる。

【技術分野】

【0 0 0 2】

本開示の態様は、一般にグラフィカル画像データのキャプチャに関し、さらに具体的には、グラフィカル画像データをキャプチャするソフトウェア開発キットに関する。さらに、本発明は、前記ソフトウェア開発キットを用いる方法に関する。さらにまた、本開示の態様は、持続的（非一時的）機械可読データ・ストレージ媒体に格納されたソフトウェア製品に関し、該ソフトウェア製品は、前記方法を実施するため、デバイスのコンピューティング・ハードウェアで実行が可能である。

【背景技術】

【0 0 0 3】

昨今、携帯デバイス、すなわち携帯電子デバイスをサポートする様々なオペレーティング・システム（OS：operating system）およびグラフィックス装置が開発されてきた。通常、携帯デバイス用グラフィックス装置は、一つ以上のグラフィックス・エンジンおよび/またはオープン・グラフィックス・ライブラリ（OpenGL：Open Graphics Library）など、一つ以上のグラフィックス・ライブラリを含む。これらのグラフィックス・エンジンおよび/またはライブラリは、携帯デバイスのグラフィカル表示機構を介してユーザに提示するグラフィカル画像を生成するために、携帯デバイスのOSで実行されている様々なアプリケーションによって呼び出すことが可能である。

【0 0 0 4】

図 1 は、携帯デバイス用の既知の従来型グラフィックス装置の概略図である。図 1 は従

来技術を表している。従来型グラフィックス装置は、例えば、携帯デバイスのOSで実行されているソフトウェア・アプリケーション102が、ユーザにグラフィカル画像データを提示するために使用することができる。

【0005】

図1を参照すると、この従来型グラフィックス装置は、グラフィックス・エンジン104と、グラフィックス・ライブラリ106と、当業者には「フレームバッファ」としても知られる、一つ以上の一時バッファ108と、ファイナル・バッファ110と、グラフィックス・ドライバ112と、グラフィックス・ハードウェア114と、ディスプレイ116を含む。図1に示されるように、グラフィックス・エンジン104と、グラフィックス・ライブラリ106と、一つ以上の一時バッファ108と、ファイナル・バッファ110とは、OS空間に駐在することができる。グラフィックス・ドライバ112は、カーネル空間中のソフトウェア機能とすることが可能である。グラフィックス・ドライバ112は、グラフィックス・ハードウェア114を介してディスプレイ116とインターフェース接続される。グラフィックス・ハードウェア114は、例えば、集積グラフィックス・チップセット、または専用グラフィックス処理ユニット(GPU: graphics processing unit)とすることが可能である。ディスプレイ116は、例えば、携帯デバイスのタッチ・スクリーン・ディスプレイなどのピクセル・ディスプレイとすることが可能である。

【0006】

ある例において、ソフトウェア・アプリケーション102は、ゲームプレイの間に、グラフィカル画像データの生成および提示が必要となるゲーム・アプリケーションまたは類似のアプリケーションである。さらに、この例において、ソフトウェア・アプリケーション102は、グラフィカル画像データを生成するために、グラフィックス・エンジン104に対し一つ以上の関数コールを行う。

【0007】

関数コールを受信するのに応じ、グラフィックス・エンジン104は、ソフトウェア・アプリケーションの状態をどのようにグラフィックスで提示するかを計算し、一つ以上の対応する関数コールをグラフィックス・ライブラリ106に送信し、次いで、該ライブラリが、一つ以上の一時バッファ108中でグラフィカル画像データを生成または修正する。一つ以上の一時バッファ中に、アプリケーション中のグラフィックスの一つのフレームを提示するための全グラフィカル画像データが完全に生成されたならば、グラフィックス・ライブラリ106は、これら一時バッファを一緒に合成してファイナル・バッファ110を生成し、これを一つの新規バッファまたは一時バッファの一つとすることが可能である。グラフィックスの一つのフレームを表すために一つだけの一時バッファが生成される状況では、その一時バッファをファイナル・バッファ110として用いることが可能である。

【0008】

続いて、ファイナル・バッファ110中のグラフィカル画像データは、グラフィックス・ドライバ112に提供される。次いで、グラフィックス・ドライバ112は、グラフィックス・ハードウェア114および/またはディスプレイ116を制御し、画像をユーザに提示する。秒あたり複数の画像フレームがユーザに提示されると、ユーザはこれらの画像フレームと一緒に組み合わせて知覚し、ディスプレイ116に動画効果が生じる。

【0009】

グラフィックス・エンジン104は、ソフトウェア・アプリケーション102とグラフィックス装置との間のインターフェースとして機能する。なお、グラフィックス・エンジン104は、通常はソフトウェア・アプリケーション102の一部である。例えば、グラフィックス・エンジン104は、単に、グラフィックス・ライブラリ106を呼び出す、ソフトウェア・アプリケーション102のルーティンである。あるいは、グラフィックス・エンジン104は、グラフィカル画像データを生成または修正する、および/またはグラフィカル画像データにデジタル画像処理を実施する役割を果たす、ソフトウェア・アプリケーション102のフル機能を備えたサブプログラムである。

【 0 0 1 0 】

グラフィックス・エンジン 1 0 4 は、ソフトウェア・アプリケーション 1 0 2 の作成者によって生成されるか、第三者からライセンス供与される、オペレーティング・システム・ベンダーにより提供されることが可能である。多くの場合、ソフトウェア・アプリケーション 1 0 2 および / またはグラフィックス・エンジン 1 0 4 は、既知のグラフィックス装置によって生成されたグラフィカル画像データをキャプチャするインターフェースを提供できない。

【 0 0 1 1 】

こうしたグラフィカル画像データのキャプチャを可能にする従来技術は、グラフィカル画像データをキャプチャする機能を包含するために、ソフトウェア・アプリケーション 1 0 2 および / またはグラフィックス・エンジン 1 0 4 の修正を必要とする。このため、ソフトウェア・アプリケーション 1 0 2 の一部として追加のソフトウェア・モジュールを含める必要がある。さらに、ソフトウェア・アプリケーション 1 0 2 の実行の間にグラフィカル画像データをキャプチャする際に、その追加のソフトウェア・モジュールを呼び出すため、グラフィックス・エンジン 1 0 4 も修正する必要がある。

【 0 0 1 2 】

しかしながら、この従来技術には幾つかの不利な点がある。第一に、ソフトウェア・アプリケーション 1 0 2 および / またはグラフィックス・エンジン 1 0 4 を大幅に修正する必要がある。第二に、こうした大幅な修正は困難で、時間がかかり面倒である。第三に、グラフィックス・エンジン 1 0 4 が第三者からライセンス供与されている場合、グラフィックス・エンジン 1 0 4 のソース・コードは大抵非開示であり、したがって、ソフトウェア・アプリケーション 1 0 2 の作成者による修正ができない。さらに、従来技術は、潜在的にグラフィカル画像を生成する際にパフォーマンス低下を被りやすい。

【 0 0 1 3 】

グラフィックスのキャプチャを可能にする別の従来技術は、グラフィックスキャプチャ機能を有する第三者のソフトウェア・アプリケーションの使用が必要である。しかしながら、こうした第三者のソフトウェア・アプリケーションは、Apple iOS および Google Android などのオペレーティング・システムに強制される、携帯デバイス中のソフトウェア・アプリケーションのサンドボックス (sandbox) 化に起因して、携帯デバイスには適さないことが多い。用語「サンドボックス」化とは、あるソフトウェア・アプリケーションが別のソフトウェア・アプリケーションのデータを読み取るのを防止するソフトウェア・セキュリティの実践をいう。

【 0 0 1 4 】

また、一部のオペレーティング・システムには、それら自体がグラフィカル画像データキャプチャ機能を標準機能として有するものがある。但し、こうしたグラフィカル画像データキャプチャ機能は、通常、携帯デバイス用に設計されたオペレーティング・システムには備わっていない。

【 0 0 1 5 】

したがって、グラフィカル画像データキャプチャを可能にする能力のある、携帯デバイス用グラフィックス装置が必要とされている。

【 摘要 】

【 0 0 1 6 】

本開示は、携帯デバイス用グラフィックス装置の提供を図る。

【 0 0 1 7 】

また、本開示は、グラフィカル画像データをキャプチャするソフトウェア開発キット (SDK : Software Development Kit) の提供も図る。

【 0 0 1 8 】

本開示は、例えば、スクリーンショット、スクリーンショットのシリーズ、および / またはビデオを含め、前記グラフィカル画像データをキャプチャする SDK を使用方法の提供も図る。

【 0 0 1 9 】

一つの態様において、本発明の実施形態は、グラフィカル画像データをキャプチャする SDKを提供する。このSDKはデバイス中で用いよう構成される。このデバイスは、グラフィカル画像データを生成する一つ以上のソフトウェア・アプリケーションを実行するよう動作可能なコンピューティング・ハードウェアを含み、該画像データは、デバイスのグラフィカル表示機構を介して見ることができる。

【 0 0 2 0 】

このデバイスの例には、以下に限らないが、移動端末、携帯電話、スマート・フォン、モバイル・インターネット・デバイス(MID: Mobile Internet Device)、ファブレット・コンピュータ、タブレット・コンピュータ、ウルトラモバイル・パーソナル・コンピュータ(UMPC: Ultra-Mobile Personal Computer)、携帯情報端末(PDA: Personal Digital Assistant)、ウェブ・パッド、ハンドヘルド・パーソナル・コンピュータ(PC: Personal Computer)、ラップトップ・コンピュータ、双方向エンターテイメント・コンピュータ、ゲーム端末、テレビジョン(TV: Television)セット、およびセットトップ・ボックス(STB: Set-Top Box)が含まれる。

【 0 0 2 1 】

SDKは、デバイスのコンピューティング・ハードウェアで、ソフトウェア・アプリケーションと同時に実行されるように構成される。

【 0 0 2 2 】

このソフトウェア・アプリケーションは、グラフィカル画像データを生成するために、グラフィックス・エンジンおよび/またはグラフィックス・ライブラリに対し、一つ以上の呼び出しを行うよう動作可能である。

【 0 0 2 3 】

SDKは、グラフィックス・エンジンおよび/またはグラフィックス・ライブラリをラップするように動作可能である利点があって、グラフィックス・エンジンおよび/またはグラフィックス・ライブラリに対する実質上全ての関数コールをインターセプトできて、それらの関数コールがSDKを通過できる。

【 0 0 2 4 】

さらにSDKは、これら関数コールがグラフィックス・エンジンおよび/またはグラフィックス・ライブラリによって受信される前に、インターセプトした関数コールを修正するよう動作できる。これらのインターセプトされた関数コールの修正によって、SDKは、グラフィックス・エンジンおよび/またはグラフィックス・ライブラリによって生成されたグラフィカル画像データにアクセス、読み取り、コピー、および/または修正できるようにされてもよい。またはSDKは、グラフィックス・エンジンおよび/またはグラフィックス・ライブラリの動作を制御するために、任意の方法で関数コールを修正してもよい。

【 0 0 2 5 】

さらにSDKは、グラフィックス・エンジンおよび/またはグラフィックス・ライブラリによって生成されたグラフィカル画像データをコピーし、次いでそのコピーしたデータを、デバイスの一時メモリまたは非一時的ストレージ中の別の位置に格納するよう動作可能であるか、あるいは、グラフィカル画像データを、例えば、該グラフィカル画像データからビデオ映像の一フレームを生成するよう動作可能なビデオ符号化ソフトウェアまたはハードウェアに送信し、こうしたビデオをデバイスのメモリまたは非一時的ストレージに格納するよう動作可能であってもよい。

【 0 0 2 6 】

さらにSDKは、グラフィックス・エンジンおよび/またはグラフィックス・ライブラリによって生成されたグラフィカル画像データを、デバイスから空間的に遠隔のデータ・ストアに、アップロードまたはストリームするよう動作可能であってもよい。

【 0 0 2 7 】

さらにまた、SDKは、ソフトウェア・アプリケーションの中に統合されてその一部部分となるように構成されてもよい。

【 0 0 2 8 】

別の態様において、本開示の実施形態は、グラフィカル画像データをキャプチャする SDKを使用する方法を提供する。

【 0 0 2 9 】

さらに別の態様において、本開示の実施形態は、持続的（非一時的）機械可読データ・ストレージ媒体に格納されたソフトウェア製品を提供し、このソフトウェア製品は、前記方法を実施するためにデバイスのコンピューティング・ハードウェアで実行が可能である。該ソフトウェア製品は、例えば、ソフトウェア・アプリケーション・ストアから、デバイスにダウンロードできてよい。

【 0 0 3 0 】

本開示の実施形態は、従来技術の前記問題を実質的に除去しまたは前記問題に少なくとも部分的に対処し、ユーザが、対応するアプリケーション・バイナリ・インターフェース（ABI：Application Binary Interface）が生成されたときにSDKすなわち「リンクング・ライブラリ」をソフトウェア・アプリケーションに含めることだけを除き、ソフトウェア・アプリケーションおよび/またはグラフィックス・エンジンおよび/またはグラフィックス・ライブラリの修正を必要とせずに、例えばゲームプレイ中に生成されたグラフィカル画像データをユーザがキャプチャすることができ、グラフィカル画像データのキャプチャ機能を可能にする。

【 0 0 3 1 】

別の態様において、グラフィックス・ライブラリへの関数コールのインターセプトおよび修正を可能にするSDKおよび/またはコア技術は、オペレーティング・システムのベンダーによりオペレーティング・システム（OS）の一部として含められ、これにより、そのオペレーティング・システムをサポートする全てのアプリケーションが、それらのアプリケーションにSDKを含めずに、グラフィカル画像データのキャプチャができてよい。

【 0 0 3 2 】

本開示のさらなる態様、利点、特徴、および目的は、例示的实施形態の図面および詳細な説明を、添付の特許請求の範囲と併せ解釈することによって明らかになるう。

【 0 0 3 3 】

当然ながら、本開示の特徴は、添付の特許請求の範囲に定義された本開示の範囲から逸脱することなく、様々な組み合わせに容易に含められる。

【 図面の簡単な説明 】

【 0 0 3 4 】

前述の摘要、および以下の例示的实施形態の詳細な説明は、添付の図面と併せて読めばよりよく理解されよう。本開示を説明するために、本開示の例示的構成が図面に示されている。しかしながら、本開示は、本明細書に開示する特定の方法及び手段に限定されない。さらに、当業者は、これら図面が一定の縮尺でないことを理解していよう。可能な場合は常に、同一エレメントは同一番号で示されている。

【 0 0 3 5 】

単なる例示として、以下の図面を参照しながら本開示の実施形態を以下に説明する。

【 図 1 】 携帯デバイスすなわち携帯電子デバイス用の従来型グラフィックス装置の概略図である（従来技術）。

【 図 2 】 本開示のある実施形態による、グラフィカル画像データをキャプチャするソフトウェア開発キット（SDK）を用いるのに適したデバイスの様々なコンポーネントの概略図である。

【 図 3 】 本開示の第一実施形態による、上記デバイスに適したグラフィックス装置の概略図である。

【図4】本開示の第二実施形態による、上記デバイスに適したグラフィックス装置の概略図である。

【図5】本開示の第三実施形態による、上記デバイスに適したグラフィックス装置の概略図である。

【図6】本開示のある実施形態による、グラフィカル画像データをキャプチャする SDKを使用する方法のステップの図である。

【0036】

添付の図面において、下線付きの番号は、その下線付き番号が付与された事項、または下線付き番号が隣接する事項を表すように用いられている。下線のない番号は、その下線なし番号をリンクするラインによって識別される事項に関する。ある番号に下線がなく、関連する矢印が付与されている場合、その下線なし番号は、矢印が指す事項全般を識別するために用いられる。

【実施形態の詳細な説明】

【0037】

以下の詳細な説明は、本開示の実施形態をおよびそれらが実施できる方法を示している。本開示の遂行のベストモードが開示されているが、当業者は、本開示を遂行または実践する他の実施形態も可能であることを認識するであろう。

【0038】

本開示の諸実施形態は、グラフィカル画像データをキャプチャするソフトウェア開発キット(SDK)を提供する。このSDKは、例えば携帯デバイスなどのデバイス中で用いられるよう構成されている。該デバイスは、デバイスのグラフィカル表示機構を介して見ることができるグラフィカル画像データを生成する、一つ以上のアプリケーションを実行するよう動作可能なコンピューティング・ハードウェアを含む。

【0039】

このデバイスの例には、以下に限らないが、移動端末、携帯電話、スマート・フォン、モバイル・インターネット・デバイス(MID)、ファブレット・コンピュータ、タブレット・コンピュータ、ウルトラモバイル・パーソナル・コンピュータ(UMPC)、携帯情報端末(PDA)、ウェブ・パッド、ハンドヘルド・パーソナル・コンピュータ(PC)、ラップトップ・コンピュータ、双方向エンターテイメント・コンピュータ、ゲーム端末、テレビジョン(TV)セット、およびセットトップ・ボックス(STB)が含まれる。

【0040】

SDKは、該デバイスのコンピューティング・ハードウェアで、一つ以上のソフトウェア・アプリケーションと同時に実行されるように構成される。

【0041】

これら一つ以上のソフトウェア・アプリケーションは、グラフィカル画像データを生成するために、デバイスのグラフィックス・エンジンおよび/またはグラフィックス・ライブラリに対し、一つ以上の関数コールを行うよう動作可能である。

【0042】

SDKは、グラフィックス・エンジンおよび/またはグラフィックス・ライブラリをラップするように動作可能である利点があって、SDKがグラフィックス・エンジンおよび/またはグラフィックス・ライブラリに対する実質上全ての関数コールをインターセプトできる。こうした関数コールがSDKを通過できてよい。

【0043】

さらに、SDKは、これら関数コールがグラフィックス・エンジンおよび/またはグラフィックス・ライブラリによって受信される前に、インターセプトした関数コールを修正するよう動作可能である。実施形態よっては、SDKは、グラフィックス・エンジンおよび/またはグラフィックス・ライブラリによって生成されたグラフィカル画像データにアクセス、読み取り、コピー、および/または修正できるようにするために、これら関数コールを修正してもよい。あるいはSDKは、グラフィックス・エンジンおよび/またはグ

グラフィックス・ライブラリが、グラフィカル画像データにアクセス、生成、または修正する仕方を変更するために、これら関数コールを修正してもよい。画像データの例には、デバイスを介して表示可能な任意のグラフィックスが含まれ、この画像データは、デバイスまたは類似物におけるビデオ、ビデオフレーム、静止画像、グラフィックスの一部、の一つ以上を含む。

【 0 0 4 4 】

さらに S D K は、グラフィックス・エンジンおよび / またはグラフィックス・ライブラリによって生成されたグラフィカル画像データのコピーを作成し、そのコピーを、デバイスの一時メモリまたは非一時的ストレージ中の別の位置に格納するよう動作可能であるか、あるいは、グラフィカル画像データを、例えば、該グラフィカル画像データからビデオ映像の一フレームを生成するビデオ符号化ソフトウェアまたはハードウェアに送信し、該ビデオをデバイスのメモリまたは非一時的ストレージに格納するよう動作可能であってもよい。

【 0 0 4 5 】

さらに S D K は、グラフィックス・エンジンおよび / またはグラフィックス・ライブラリによって生成されたグラフィカル画像データを、デバイスから空間的に遠隔のデータ・ストアに、アップロードまたはストリームするよう動作可能であってもよい。

【 0 0 4 6 】

さらに S D K は、ソフトウェア・アプリケーションの中に統合されてその一体部分となるように構成されてもよい。

【 0 0 4 7 】

次いで、図面を特にそれらの参照番号によって参照すると、図 2 は、本開示の実施形態による、例えば、グラフィカル画像データをキャプチャする S D K を用いるのに適したデバイス 2 0 0 の様々なコンポーネントの概略図である。デバイス 2 0 0 は、以下に限らないが、データ・メモリ 2 0 2 と、プロセッサ 2 0 4 などのコンピューティング・ハードウェアと、入力 / 出力 (I / O : I n p u t / O u t p u t) デバイス 2 0 6 と、ネットワーク・インターフェース 2 0 8 と、データ・メモリ 2 0 2 、プロセッサ 2 0 4 、 I / O デバイス 2 0 6 、およびネットワーク・インターフェース 2 0 8 を含む様々なコンポーネントを作動可能に接続するシステム・バス 2 1 0 と、を含む。入力 / 出力 (I / O) デバイス 2 0 6 は非一時的データ・ストレージ機構を含んでもよい。

【 0 0 4 8 】

データ・メモリ 2 0 2 は、固定型メモリ、リムーバブル・メモリ、またはそれらの組み合わせを含む。固定型メモリは、例えば、ランダムアクセス・メモリ (R A M : R a n d o m - A c c e s s M e m o r y) 、読み取り専用メモリ (R O M : R e a d - O n l y M e m o r y) 、フラッシュ・メモリ、またはハード・ドライブを含む。リムーバブル・メモリは、例えば、フラッシュ・メモリ・カード、メモリ・スティック、またはスマート・カードを含んでもよい。

【 0 0 4 9 】

メモリ 2 0 2 は、プロセッサ 2 0 4 で実行されたとき、様々なソフトウェア・アプリケーションをその上で実行できるインターフェースを提供する、オペレーティング・システム (O S) 2 1 2 を格納する。さらに、O S 2 1 2 は、デバイス 2 0 0 のユーザに、その中から選択が可能な一つ以上の選択肢を提供するユーザ・インターフェース (図 2 に示さず) を含むことができる。

【 0 0 5 0 】

例えば、該ユーザ・インタフェースによって、ユーザが遠隔サーバからゲーム・アプリケーションをダウンロードし、そのゲーム・アプリケーションをプロセッサ 2 0 4 で実行してゲームをプレイできてもよい。但し、当然ながら、ゲーム・アプリケーション以外のソフトウェア・アプリケーションを代わりにまたは加えてダウンロードできてもよい。さらに、ユーザには、ゲームプレイの間に生成されたグラフィカル画像データをキャプチャする選択肢が提供される。グラフィカル画像データが取り込まれる仕方の詳細は、図 3 、

4、および5に関連させて説明されている。これにより取り込まれたグラフィカル画像データは、次いでデータ・メモリ202に格納されてもよい。前述したように、本明細書で説明する諸実施形態は、ゲーム・アプリケーションに限定されない。すなわち、キャプチャは、任意のソフトウェア・アプリケーションとともに使用できる。

【0051】

さらに、取り込まれたグラフィカル画像データは、ネットワーク・インターフェース208を介して他のデバイスを用いる他のユーザと共有されてもよい。ネットワーク・インターフェース208によって、デバイス200が例えば、通信ネットワーク（図2に示さず）を介して他のデバイスと通信できてよい。加えて、ネットワーク・インターフェース208で、デバイス200が取り込まれたグラフィカル画像データを、通信ネットワークを介して遠隔サーバにアップロードまたはストリームできてよい。

【0052】

この通信ネットワークは例えば、互いに相互接続されて単一の大きなネットワークとして機能する、個別ネットワークの集合体である。こうした個別のネットワークは、有線、無線、またはそれらの組み合わせであってもよい。個別のネットワークの例には、以下に限らないが、ローカル・エリア・ネットワーク（LAN：Local Area Network）、広域ネットワーク（WAN：Wide Area Network）、メトロポリタン・エリア・ネットワーク（MAN：Metropolitan Area Network）、無線LAN（WLAN：Wireless LAN）、無線WAN（WWAN：Wireless WAN）、無線MAN（WMAN：Wireless MAN）、第二世代（2G）通信ネットワーク、第三世代（3G）通信ネットワーク、第四世代（4G）通信ネットワーク、およびワールドワイド・インターオペラビリティ・フォー・マイクロウェーブ・アクセス（WiMAX：Worldwide Interoperability for Microwave Access）ネットワークが含まれる。

【0053】

デバイス200は、移動端末、携帯電話、スマート・フォン、MID、ファブレット・コンピュータ、タブレット・コンピュータ、UMPC、PDA、ウェブ・パッド、ハンドヘルドPC、ラップトップ・コンピュータ、双方向エンターテインメント・コンピュータ、ゲーム端末、TVセット、およびセットトップ・ボックス（STB）の少なくとも一つを使って実装されてもよい。

【0054】

図2は単なる例であって、添付の特許請求の範囲を不当に限定するものではない。当然ながら、デバイス200に対する具体的指定は一例として提供されたものであり、デバイス200を、特定の数、種類、または配置のモジュールおよび/またはコンポーネントのデバイス200に限定すると解釈されるべきではない。当業者は、本開示の実施形態の多くの変形、代替、および修正を認識されよう。

【0055】

図3は、本開示の第一実施形態による、デバイス200に適したグラフィックス装置の概略図である。説明のため、本グラフィックス装置は、デバイス200のプロセッサ204で実行されるソフトウェア・アプリケーション302によって使用される。

【0056】

本グラフィックス装置は、グラフィックス・エンジン304と、ラッパ306と、グラフィックス・ライブラリ308と、一つ以上の一時バッファ310と、ファイナル・バッファ312と、ビデオ・キャプチャ・コンポーネント314と、グラフィックス・ドライバ316と、グラフィックス・ハードウェア318と、ディスプレイ320を含む。ラッパ306とビデオ・キャプチャ・コンポーネント314とは、グラフィカル画像データをキャプチャするSDKの一部を形成する。グラフィックス・ドライバ316と、グラフィックス・ハードウェア318と、ディスプレイ320とは、デバイス200のグラフィカル表示機構の一部を形成する。

【 0 0 5 7 】

本開示のある実施形態によれば、SDKは、デバイス200のプロセッサ204で、ソフトウェア・アプリケーション302と同時に、またはその一部として実行するように構成される。

【 0 0 5 8 】

グラフィックス・エンジン304と、ラッパー306、グラフィックス・ライブラリ308、一時バッファ310、ファイナル・バッファ312、ビデオ・キャプチャ・コンポーネント314は、デバイス200のデータ・メモリ202に格納されてもよい。また、図3に示されるようにOS212に駐在したり、OS212で実行されたりしてもよい。

【 0 0 5 9 】

例えば、グラフィックス・エンジン304は、オブジェクト指向グラフィックス・レンダリング・エンジン(OGRE: Object-oriented Graphics Rendering Engine)、Irrlicht 3Dエンジン、Visual 3Dゲーム・エンジン、Truevision 3Dエンジン、Crystal Spaceエンジン、Unity 3Dゲーム・エンジン、Unreal Engine、Cocos 2Dエンジンなど、第三者からライセンス供与されたグラフィックス・エンジンでもよい。グラフィックス・ライブラリ308は、オープン・グラフィックス・ライブラリ(OpenGL: Open Graphics Library) API、Direct3D API、EAGLグラフィックス・ライブラリ(EAGL: EA Graphics Library) API、エンベデッド・システム・グラフィックス・ライブラリ(EGLE: Embedded-system Graphics Library) APIなど、クロスプラットフォームのグラフィックス・ライブラリ・アプリケーション・プログラミング・インタフェース(API: Application Programming Interface)を介して実装されてもよい。

【 0 0 6 0 】

さらに、グラフィックス・ドライバ316は、オペレーティング・システム(OS)のカーネル中のソフトウェア機能であってもよい。グラフィックス・ドライバ316は、システム供給のドライバ・コンポーネント、ベンダー供給のドライバ・コンポーネント、またはこれらの組み合わせであってもよい。グラフィックス・ドライバ316は、グラフィックス・ハードウェア318を介してディスプレイ320にインターフェース接続される。グラフィックス・ハードウェア318は例えば、デバイス200の集積グラフィックス・チップセット、または専用グラフィックス処理ユニット(GPU)であってもよい。ディスプレイ320は例えば、デバイス200のタッチ・スクリーン・ディスプレイなどのピクセル・ディスプレイであってもよい。

【 0 0 6 1 】

ある例示的实施形態において、ソフトウェア・アプリケーション302は、ゲームプレイの間、デバイス200のグラフィカル表示機構を介して見ることができるグラフィカル画像データの生成を必要とするゲーム・アプリケーションである。さらに、この例示的な実施形態において、デバイス200のユーザは、ゲームプレイの間に生成されたグラフィカル画像データをキャプチャすることを選ぶ。

【 0 0 6 2 】

グラフィカル画像データを生成するために、ソフトウェア・アプリケーション302は、グラフィックス・エンジン304に対し一つ以上の関数コールを行う。グラフィックス・エンジン304は、ソフトウェア・アプリケーション302とグラフィックス装置との間のインターフェースとしての機能を果たす。

【 0 0 6 3 】

グラフィックス・エンジン304は、通常、少なくとも部分的にソフトウェア・アプリケーション302中に実装される。あるいは、グラフィックス・エンジン304は、全体がソフトウェア・アプリケーション302とは別のルーティンとして実装されてもよい。グラフィックス・エンジン304は、ソフトウェア・アプリケーション302の作成者に

よって生成されるか、もしくは第三者からライセンス供与されてもよい。

【0064】

関数コールを受信するのに応じて、グラフィックス・エンジン304は、ソフトウェア・アプリケーションの状態をグラフィックスで表現する仕方を計算し、グラフィックス・ライブラリ308に対し、一つ以上の関数コールを行う。これは、ラッパー306を起動し、ラッパーは、グラフィックス・エンジン304が行うグラフィックス・ライブラリ308への関数コールをインターセプトするよう動作可能である。

【0065】

ラッパー306は、グラフィックス・ライブラリ308をラップするように動作できる利点があって、グラフィックス・ライブラリ308に対する実質上全ての関数コールがラッパー306を通過できる。なお、ラッパー306はアプリケーション302の一部である。

【0066】

S D Kが、グラフィックス・エンジン304および/またはグラフィックス・ライブラリ308によって生成されたグラフィカル画像データにアクセス、読み取り、コピー、および/または修正できるように、ラッパー306は、関数コールがグラフィックス・ライブラリ308によって受信される前にこれら関数コールを修正してもよい。あるいは、グラフィックス・ライブラリ308の動作を制御するため、別途に任意の方法で関数コールを修正してもよい。

【0067】

さらにS D Kは、一時バッファ310および/またはファイナル・バッファ312から、グラフィックス・エンジン304および/またはグラフィックス・ライブラリ308によって生成されたグラフィカル画像データのコピーを作成するよう動作可能であってもよい。特定の関数コールがラッパー306を通り過ぎるとき、ラッパー306は、ビデオ・キャプチャ・コンポーネント314が一時バッファ310および/またはファイナル・バッファ312からグラフィカル画像データをコピーできるような方法で、グラフィックス・ライブラリ308を制御するため、特定の関数コールがグラフィックス・ライブラリ308によって受信される前にその特定の関数コールを修正する。

【0068】

グラフィックス・エンジン304が単一の一時バッファ310のみを使用している例において、その一時バッファ310は、ファイナル・バッファ312としても使われ、関数コールに対しS D Kが行った、または行う修正によって、ビデオ・キャプチャ・コンポーネントが、当該バッファ中の、グラフィックス・ライブラリ308によって生成または修正されたデータを読み取れてもよい。グラフィックス・エンジン304が当該バッファ中の画像フレームの描画を完了したとき、グラフィックス・エンジン304は、完了した画像フレームをユーザに提示するため、グラフィックス・ライブラリ308を呼び出す。それに応じて、ラッパー306は、ビデオ・キャプチャ・コンポーネント314が、デバイス・メモリの他の場所にファイナル・バッファ312のコピーを格納したことを確かにし、その後、グラフィックス・ライブラリ308は、ユーザにファイナル・バッファ312を提示するために、グラフィックス・ドライバ316を呼び出す。別の実施形態において、ビデオ・キャプチャ・コンポーネント314は、画像フレームが完全に描画される前に一時バッファ310またはファイナル・バッファ312のコピーを格納し、すなわち、ユーザ・インタフェース・エレメントなど特定のグラフィカル・エレメントが当該対応バッファ中に生成される前に、グラフィカル画像データをキャプチャすることができる。

【0069】

グラフィックス・エンジン304が、複数のバッファ、すなわち一つ以上の一時バッファ310を使用する別の例において、グラフィックス・ライブラリ308は、グラフィカル画像データをそれら一つ以上の一時バッファ310に書き込む。グラフィックス・エンジン304が、一つ以上の一時バッファ310中の画像フレームの描画を完了したとき、グラフィックス・エンジン304は、ユーザに合成された画像フレームを提示するために

グラフィックス・ライブラリ 3 0 8 を呼び出す。それに応じて、ラッパー 3 0 6 は、グラフィックス・ライブラリ 3 0 8 が、グラフィカル画像データを一つ以上の一時バッファ 3 1 0 からファイナル・バッファ 3 1 2 の中に合成したことを確かにし、該データはビデオ・キャプチャ・コンポーネント 3 1 4 による読み取りが可能である。合成が終わったならば、ビデオ・キャプチャ・コンポーネント 3 1 4 が、ファイナル・バッファ 3 1 2 からの合成済みグラフィカル画像データのコピーを、デバイス・メモリ中の別の場所に格納することになり、その後、グラフィックス・ドライバ 3 1 6 がそのファイナル・バッファ 3 1 2 をユーザに提示することになる。別の実施形態において、ビデオ・キャプチャ・コンポーネント 3 1 4 は、フレームが完成する前に一時バッファ 3 1 0 またはファイナル・バッファ 3 1 2 のコピーを格納し、すなわち、ユーザ・インタフェース・エレメントなど特定のグラフィカル・エレメントが生成される前に、グラフィカル画像データをキャプチャするように動作可能である。

【 0 0 7 0 】

完成した画像フレームを受信するのに応じ、グラフィックス・ドライバ 3 1 6 は、グラフィックス・ハードウェア 3 1 8 および / またはディスプレイ 3 2 0 を制御し、完成した画像フレームをユーザに提示する。

【 0 0 7 1 】

一方、ビデオ・キャプチャ・コンポーネント 3 1 4 は、一時バッファ 3 1 0 またはファイナル・バッファ 3 1 2 のいずれかに書き込まれたグラフィカル画像データをキャプチャし、そのグラフィカル画像データをメモリ 2 0 2 中に、またはデバイス 2 0 0 中の別の内部ストレージの中にローカルに格納する、あるいは、例えば、ビデオ・エンコーダ・ソフトウェアまたはハードウェアを使ってグラフィカル画像データのフレーム群をビデオ・ファイルの中に圧縮するため、グラフィカル画像データをさらに処理するよう動作可能である。これにより、グラフィカル画像データは、例えば、ソフトウェア・アプリケーション 3 0 2 の動作のビデオ記録を提供するため等で、後の時点での画像フレーム群の再生に使われてもよい。

【 0 0 7 2 】

ビデオ・キャプチャ・コンポーネント 3 1 4 は、ファイナル・バッファ 3 1 2 のコンテンツを、デバイス 2 0 0 から空間的に遠隔のデータ・ストアにアップロードおよび / またはストリームするよう動作できる利点がある。こうしたコンテンツは例えば、ゲームプレイの経験を共有する目的のため、データ・ストアから一つ以上の他のユーザのデバイスにストリームされてもよいが、この目的に限らない。

【 0 0 7 3 】

さらに、SDK は、ラッパー 3 0 6 およびビデオ・キャプチャ・コンポーネント 3 1 4 を含め、ソフトウェア・アプリケーション 3 0 2 の中に統合されその一体部分となるように構成されてもよい。ラッパー 3 0 6 およびビデオ・キャプチャ・コンポーネント 3 1 4 は、例えば、ソフトウェア・アプリケーション 3 0 2 がオブジェクト・コード・ライブラリと一緒にリンクされている場合、ソフトウェア・アプリケーション 3 0 2 のアプリケーション・バイナリ・インターフェース (ABI) に含まれてもよい。こうしたリンク付けは、例えば、実行可能ファイルが生成される場合、またはソフトウェア・アプリケーション 3 0 2 がランタイムで使われる場合は常に、有利に行われる。結果として、ラッパー 3 0 6 およびビデオ・キャプチャ・コンポーネント 3 1 4 は、例えばソフトウェア・ディベロッパに実装されるような、透過的な仕方ではソフトウェア・アプリケーション 3 0 2 の一部となってもよい。

【 0 0 7 4 】

さらに、ソフトウェア・アプリケーション 3 0 2 は、SDK によって提供されるグラフィックスキャプチャ機能をユーザが「有効」または「無効」にできるよう動作可能であってもよい。

【 0 0 7 5 】

ソフトウェア・アプリケーション 3 0 2、グラフィックス・エンジン 3 0 4、および /

またはグラフィックス・ライブラリ 3 0 8 は、S D K をソフトウェアのアプリケーション・バイナリ・インターフェース (A B I) にリンクすることを除き、実質的に修正が不要であってもよい。

【 0 0 7 6 】

次いで、ソフトウェア・アプリケーション 3 0 2 が、グラフィカル画像データを生成するためにグラフィックス・エンジン 3 0 4 を呼び出すとき、グラフィックス・エンジン 3 0 4 は、グラフィックス・ライブラリ 3 0 8 を呼び出すことを意図して、代わりにラッパ- 3 0 6 を透過的に呼び出す。それに続いて、ラッパ- 3 0 6 は、ビデオ・キャプチャ・コンポーネント 3 1 4 が、生成されたグラフィカル画像データに対して、例えばこうしたデータを読み取り、コピーおよび修正するフルアクセスを有し、これにより、グラフィックス・エンジン 3 0 4 および / またはグラフィックス・ライブラリ 3 0 8 によって生成されたグラフィカル画像データのキャプチャを可能にするように、グラフィックス・ライブラリ 3 0 8 を設定するため、関数コールが通り過ぎる前にそれらを修正する。

【 0 0 7 7 】

上記に換えて、S D K は、O S 2 1 2 の中に統合されてその一体部分となるように構成されてもよい。これは、O S 2 1 2 で実行される一切のソフトウェア・アプリケーションに対しグラフィックスキャプチャ機能を有利に提供する。こうした例において O S 2 1 2 は、デバイス 2 0 0 のユーザが例えば、ソフトウェア・アプリケーション毎ベースでグラフィックスキャプチャ機能を制御できる機能を提供してもよい。

【 0 0 7 8 】

こうした実装によって、携帯デバイスで実行される全ソフトウェア・アプリケーションに対してシステム全体のグラフィックスキャプチャ機能を提供する O S 2 1 2 を使用した携帯デバイスを、ハードウェア製造者が製造できるという利点がある。さらに、オープンソース・オペレーティング・システムの場合、この実装で、このシステム全体のグラフィックスキャプチャ機能を含めるために、そうしたオープンソース・オペレーティング・システムを個々のディベロッパが修正できるという利点がある。

【 0 0 7 9 】

ラッパ- 3 0 6 の実行は、ラッパ- 3 0 6 自体がグラフィカル画像データを生成するのではないので、測定可能なほどのパフォーマンス低下につながることはないという利点がある。ビデオ・キャプチャ・コンポーネント 3 1 4 によるグラフィカル画像データのキャプチャ、およびこうしたグラフィカル画像データのビデオ・ファイルへの格納または圧縮がある程度パフォーマンス低下をもたらす可能性はあるが、これは、ソフトウェア・アプリケーション 3 0 2 の実行に起因するパフォーマンス低下と比較すると実質的に無視できる。

【 0 0 8 0 】

図 3 は単なる例であって、添付の特許請求の範囲を不当に限定するものではない。グラフィックス装置に対する具体的指定は、該グラフィックス装置を、特定の数、種類、配置のモジュール、ルーティン、ソフトウェア製品、コンポーネントおよび / または A P I のグラフィックス装置に限定すると解釈されるべきではない。当業者は、本開示の実施形態の多くの変形、代替、および修正を認識しよう。

【 0 0 8 1 】

例えば、ゲームプレイの間に生成されたグラフィカル画像データとともに、ビデオおよび / またはデジタル・オーディオ・データをキャプチャするために、グラフィックス装置と類似の仕方、デバイス 2 0 0 にビデオ装置が実装されてもよい。グラフィカル画像データとともにデジタル・オーディオ・データから成るビデオ・ファイルは、次いで、遠隔のサーバに有利にアップロードもしくはストリームされ、および / または他のユーザと共有される。

【 0 0 8 2 】

図 4 は、本開示の第二実施形態による、デバイス 2 0 0 に適したグラフィックス装置の概略図であり、別途に述べられた個所を除き、図 4 および図 5 は図 3 に記載の機能の準用

である。説明上、グラフィックス装置は、デバイス 200 のプロセッサ 204 で実行されるソフトウェア・アプリケーション 402 によって使用される。

【0083】

このグラフィックス装置は、ラッパ 404 と、グラフィックス・エンジン 406 と、グラフィックス・ライブラリ 408 と、一つ以上の一時バッファ 410 と、ファイナル・バッファ 412 と、ビデオ・キャプチャ・コンポーネント 414 と、グラフィックス・ドライバ 416 と、グラフィックス・ハードウェア 418 と、ディスプレイ 420 とを含む。ラッパ 404 と、ファイナル・バッファ 412 と、ビデオ・キャプチャ・コンポーネント 414 とは、グラフィカル画像データを キャプチャする SDK の一部を形成する。グラフィックス・ドライバ 416 と、グラフィックス・ハードウェア 418 と、ディスプレイ 420 とは、デバイス 200 のグラフィカル表示機構の一部を形成する。

【0084】

本開示のある実施形態によれば、SDK は、デバイス 200 のプロセッサ 204 で、ソフトウェア・アプリケーション 402 と同時に、またはその一部として実行するよう構成される。

【0085】

ラッパ 404 と、グラフィックス・エンジン 406、グラフィックス・ライブラリ 408、一時バッファ 410、ファイナル・バッファ 412、ビデオ・キャプチャ・コンポーネント 414 は、デバイス 200 のデータ・メモリ 202 に格納されてもよい。また、図 4 に示されるように OS 212 に駐在したり、OS 212 で実行されたりしてもよい。

【0086】

例えば、グラフィックス・エンジン 406 は、オブジェクト指向グラフィックス・レンダリング・エンジン (OGRE)、Irrlicht 3D エンジン、Visual3D ゲーム・エンジン、Truevision 3D エンジン、Crystal Space エンジン、Unity 3D ゲーム・エンジン、Unreal Engine、Cocos2D エンジンなど、第三者からライセンス供与されたグラフィックス・エンジンでもよい。グラフィックス・ライブラリ 408 は、OpenGL API、Direct3D API、EAGL API、EGL API など、クロスプラットフォームのグラフィックス・ライブラリ API を介して実装されてもよい。

【0087】

さらに、グラフィックス・ドライバ 416 は、オペレーティング・システム (OS) のカーネル中のソフトウェア機能であってもよい。グラフィックス・ドライバ 416 は、システム供給のドライバ・コンポーネント、ベンダー供給のドライバ・コンポーネント、またはこれらの組み合わせであってもよい。グラフィックス・ドライバ 416 は、グラフィックス・ハードウェア 418 を介してディスプレイ 420 にインターフェース接続される。グラフィックス・ハードウェア 418 は例えば、デバイス 200 の集積グラフィックス・チップセット、または専用グラフィックス処理ユニット (GPU) であってもよい。ディスプレイ 420 は例えば、デバイス 200 のタッチ・スクリーン・ディスプレイなどのピクセル・ディスプレイであってもよい。

【0088】

ある例示的实施形態において、ソフトウェア・アプリケーション 402 は、ゲームプレイの間、デバイス 200 のグラフィカル表示機構を介して見ることが出来るグラフィカル画像データの生成を必要とするゲーム・アプリケーションである。さらに、デバイス 200 のユーザは、この例示的な実施形態において、ゲームプレイの間に生成されたグラフィカル画像データを キャプチャする ことを選ぶ。

【0089】

グラフィカル画像データを生成するために、ソフトウェア・アプリケーション 402 は、グラフィックス・エンジン 406 に対し一つ以上の関数コールを行う。グラフィックス・エンジン 406 は、ソフトウェア・アプリケーション 402 とグラフィックス装置との間のインターフェースとしての機能を果たす。

【 0 0 9 0 】

グラフィックス・エンジン 4 0 6 は、通常、ソフトウェア・アプリケーション 4 0 2 の部分として実装される。あるいは、グラフィックス・エンジン 4 0 6 は、全体がソフトウェア・アプリケーション 4 0 2 とは別のルーティンとして実装される。グラフィックス・エンジン 4 0 6 は、ソフトウェア・アプリケーション 4 0 2 の作成者によって生成されるか、もしくは第三者からライセンス供与されてもよい。

【 0 0 9 1 】

ソフトウェア・アプリケーション 4 0 2 が、グラフィックス・エンジン 4 0 6 に対し関数コールを行うと、ラッパー 4 0 4 が起動される。ラッパー 4 0 4 は、ソフトウェア・アプリケーション 4 0 2 が行うグラフィックス・エンジン 4 0 6 への関数コールをインターセプトするよう動作可能である。

【 0 0 9 2 】

ラッパー 4 0 4 は、グラフィックス・エンジン 4 0 6 をラップするように動作可能で、グラフィックス・エンジン 4 0 6 へのおよび / またはグラフィックス・エンジン 4 0 6 からの実質上全ての関数コールがラッパー 4 0 4 を通過できてよい。なお、ラッパー 4 0 4 はアプリケーション 4 0 2 の部分である。

【 0 0 9 3 】

関数コール受信するのに応じ、グラフィックス・エンジン 4 0 6 は、ソフトウェア・アプリケーションの状態をグラフィックスで提示する仕方を計算し、グラフィックス・ライブラリ 4 0 8 に対し一つ以上の関数コールを行う。それに応じ、ラッパー 4 0 4 は、グラフィックス・エンジン 4 0 6 がグラフィックス・ライブラリ 4 0 8 に行う関数コールをインターセプトするよう動作可能である。

【 0 0 9 4 】

グラフィカル画像データをキャプチャするために、ラッパー 4 0 4 は、関数コールがグラフィックス・エンジン 4 0 6 および / またはグラフィックス・ライブラリ 4 0 8 によって受信される前にこれらの関数コールを修正するように動作しうる。それによって S D K が、グラフィックス・エンジン 4 0 6 および / またはグラフィックス・ライブラリ 4 0 8 によって生成されたグラフィカル画像データにアクセス、読み取り、コピー、および / または修正できるようにする。あるいはラッパー 4 0 4 は、グラフィックス・エンジン 4 0 6 および / またはグラフィックス・ライブラリ 4 0 8 の動作を制御するため、別途に任意の方法で関数コールを修正してもよい。

【 0 0 9 5 】

さらに S D K は、一時バッファ 4 1 0 および / またはファイナル・バッファ 4 1 2 から、グラフィックス・エンジン 4 0 6 および / またはグラフィックス・ライブラリ 4 0 8 によって生成されたグラフィカル画像データのコピーを作成するよう動作可能であってもよい。特定の関数コールがラッパー 4 0 4 を通り過ぎるとき、ラッパー 4 0 4 は、ビデオ・キャプチャ・コンポーネント 4 1 4 が一時バッファ 4 1 0 および / またはファイナル・バッファ 4 1 2 からグラフィカル画像データをコピーできるような方法で、グラフィックス・エンジン 4 0 6 および / またはグラフィックス・ライブラリ 4 0 8 を制御するため、特定の関数コールがグラフィックス・エンジン 4 0 6 および / またはグラフィックス・ライブラリ 4 0 8 によって受信される前にその特定の関数コールを修正する。

【 0 0 9 6 】

グラフィックス・エンジン 4 0 6 が単一の一時バッファ 4 1 0 のみを使用している例において、その一時バッファは、ファイナル・バッファ 4 1 2 としても使われ、関数コールに対し S D K が行った、または行う修正によって、ビデオ・キャプチャ・コンポーネントが、当該バッファ中の、グラフィックス・ライブラリ 4 0 8 によって生成または修正されたデータを読み取れてもよい。それに応じて、ラッパー 4 0 4 は、ビデオ・キャプチャ・コンポーネント 4 1 4 が、デバイス・メモリの他の場所にファイナル・バッファ 4 1 2 のコピーを格納したことを確かにし、その後、グラフィックス・ライブラリ 4 0 8 は、ユーザにファイナル・バッファ 4 1 2 を提示するために、グラフィックス・ドライバ 4 1 6 を

呼び出す。別の実施形態において、ビデオ・キャプチャ・コンポーネントは、画像フレームが完全に描画される前に一時バッファ 410 またはファイナル・バッファ 412 のコピーを格納し、すなわち、ユーザ・インタフェース・エレメントなど特定のグラフィカル・エレメントが当該対応バッファ中に生成される前に、グラフィカル画像データをキャプチャすることができる。

【0097】

グラフィックス・エンジン 406 が、複数のバッファ、すなわち複数の一時バッファ 410 を使用する別の例の状況において、グラフィックス・ライブラリ 408 は、グラフィカル画像データをそれら一時バッファ 410 に書き込む。グラフィックス・エンジン 406 が、一時バッファ 410 中の画像フレームの描画を完了したとき、グラフィックス・エンジン 406 は、合成された画像フレームをユーザに提示するためにグラフィックス・ライブラリ 408 を呼び出す。それに応じて、ラッパー 404 は、グラフィックス・ライブラリ 408 が、グラフィカル画像データを一つ以上の一時バッファ 410 からファイナル・バッファ 412 の中に合成したことを確かにし、このデータはビデオ・キャプチャ・コンポーネント 414 による読み取りが可能である。合成が終わったならば、ビデオ・キャプチャ・コンポーネント 414 が、ファイナル・バッファ 412 からの合成済みグラフィカル画像データのコピーを、デバイス・メモリ中の別の場所に格納することになり、その後、グラフィックス・ドライバ 416 がそのファイナル・バッファ 412 をユーザに提示することになる。別の実施形態において、ビデオ・キャプチャ・コンポーネント 414 は、フレームが完成する前に一時バッファ 410 またはファイナル・バッファ 412 のコピーを格納し、すなわち、ユーザ・インタフェース・エレメントなど特定のグラフィカル・エレメントが生成される前に、グラフィカル画像データをキャプチャするように動作可能である。

【0098】

完成した画像フレームを受信するのに応じ、グラフィックス・ドライバ 416 は、グラフィックス・ハードウェア 418 および / またはディスプレイ 420 を制御し、完成した画像フレームをユーザに提示する。

【0099】

一方、ビデオ・キャプチャ・コンポーネント 414 は、一時バッファ 410 またはファイナル・バッファ 412 のいずれかに書き込まれたグラフィカル画像データをキャプチャし、そのグラフィカル画像データをメモリ 202 中に、またはデバイス 200 中の別の内部ストレージの中にローカルに格納するよう動作可能であるか、あるいは、例えば、ビデオ・エンコーダ・ソフトウェアまたはハードウェアを使ってグラフィカル画像データのフレーム群をビデオ・ファイルの中に圧縮するため、グラフィカル画像データをさらに処理するよう動作可能である。これにより、グラフィカル画像データは、例えばソフトウェア・アプリケーション 402 の動作のビデオ記録を提供するため、例えば後の時点での画像フレーム群の再生に使われてもよい。

【0100】

ビデオ・キャプチャ・コンポーネント 414 は、ファイナル・バッファ 412 のコンテンツを、デバイス 200 から空間的に遠隔のデータ・ストアにアップロードおよび / またはストリームするよう動作できるという利点があってもよい。こうしたコンテンツは、例えば、ゲームプレイの経験を共有する目的のため、データ・ストアから一つ以上の他のユーザのデバイスにストリームされてもよいが、この目的に限らない。

【0101】

さらに SDK は、ラッパー 404 およびビデオ・キャプチャ・コンポーネント 414 を含め、ソフトウェア・アプリケーション 402 の中に統合されてその一体部分となるように構成されてもよい。ラッパー 404 およびビデオ・キャプチャ・コンポーネント 414 は、例えば、ソフトウェア・アプリケーション 402 がオブジェクト・コード・ライブラリと一緒にリンクされている場合、ソフトウェア・アプリケーション 402 のアプリケーション・バイナリ・インターフェース (ABI) に含まれてもよい。こうしたリンク付け

は、例えば、実行可能ファイルが生成される場合、またはソフトウェア・アプリケーション 402 がランタイムで使われる場合は常に、有利に行われる。結果として、ラッパー 404 およびビデオ・キャプチャ・コンポーネント 414 は、例えばソフトウェア・ディベロッパに実装されるような、透過的な仕方ではソフトウェア・アプリケーション 402 の一部となってもよい。

【0102】

さらに、ソフトウェア・アプリケーション 402 は、SDK によって提供されるグラフィックスキャプチャ機能をユーザが「有効」または「無効」にできるように動作可能であってもよい。

【0103】

ソフトウェア・アプリケーション 402、グラフィックス・エンジン 406、および/またはグラフィックス・ライブラリ 408 は、SDK をアプリケーション・バイナリ・インターフェース (ABI) にリンクすることを除き、修正が不要でもよい。

【0104】

次いで、ソフトウェア・アプリケーション 402 が、グラフィカル画像データを生成するためにグラフィックス・エンジン 406 を呼び出すとき、ソフトウェア・アプリケーション 402 は、代わりにラッパー 404 を透過的に呼び出す。それに続いて、ラッパー 404 は、ビデオ・キャプチャ・コンポーネント 414 が、生成されたグラフィカル画像データに対して、例えばこうしたデータを読み取り、コピーおよび修正するフルアクセスを有し、これにより、グラフィックス・エンジン 406 および/またはグラフィックス・ライブラリ 408 によって生成されたグラフィカル画像データのキャプチャを可能にするように、グラフィックス・エンジン 406 および/またはグラフィックス・ライブラリ 408 を設定するため、関数コールが通り過ぎる前にそれらを修正する。

【0105】

ラッパー 404 の実行は、ラッパー 404 自体がグラフィカル画像データを生成するのではないので、測定可能なほどのパフォーマンス低下につながることはないという利点がある。ビデオ・キャプチャ・コンポーネント 414 によるグラフィカル画像データのキャプチャ、およびこうしたグラフィカル画像データのビデオ・ファイルへの格納または圧縮がある程度パフォーマンス低下をもたらす可能性はあるが、これは、ソフトウェア・アプリケーション 402 の実行に起因するパフォーマンス低下と比較すると実質的に無視できる。

【0106】

図 4 は単なる例であって、添付の特許請求の範囲を不当に限定するものではない。当然ながら、グラフィックス装置に対する具体的指定は、該グラフィックス装置を、特定の数、種類、配置のモジュール、ルーティン、ソフトウェア製品、コンポーネントおよび/または API のグラフィックス装置に限定すると解釈されるべきではない。当業者は、本開示の実施形態の多くの変形、代替、および修正を認識していよう。

【0107】

例えば、ゲームプレイの間に生成されたグラフィカル画像データとともに、ビデオおよび/またはデジタル・オーディオ・データをキャプチャするために、グラフィックス装置と類似の仕方ではビデオ装置がデバイス 200 に実装されてもよい。グラフィカル画像データとともにデジタル・オーディオ・データから成るビデオ・ファイルは、次いで、遠隔のサーバに有利にアップロードされ、および/または他のユーザと共有される。

【0108】

図 5 は、本開示の第三実施形態による、デバイス 200 に適したグラフィックス装置の概略図である。説明のために、グラフィックス装置は、デバイス 200 のプロセッサ 204 で実行されるソフトウェア・アプリケーション 502 によって使用される。

【0109】

このグラフィックス装置は、グラフィックス・エンジン 504 と、ビデオキャプチャ API 506 と、グラフィックス・ライブラリ 508 と、一つ以上の一時バッファ 510 と

、ファイナル・バッファ 5 1 2 と、ビデオ・キャプチャ・コンポーネント 5 1 4 と、グラフィックス・ドライバ 5 1 6 と、グラフィックス・ハードウェア 5 1 8 と、ディスプレイ 5 2 0 とを含む。ビデオキャプチャ API 5 0 6 と、ビデオ・キャプチャ・コンポーネント 5 1 4 とは、グラフィカル画像データをキャプチャする SDK の一部を形成する。グラフィックス・ドライバ 5 1 6 と、グラフィックス・ハードウェア 5 1 8 と、ディスプレイ 5 2 0 とは、デバイス 2 0 0 のグラフィカル表示機構の一部を形成する。

【 0 1 1 0 】

本開示のある実施形態によれば、SDK は、デバイス 2 0 0 のプロセッサ 2 0 4 で、ソフトウェア・アプリケーション 5 0 2 と同時に、またはその一部として実行するよう構成される。

【 0 1 1 1 】

グラフィックス・エンジン 5 0 4 と、ビデオキャプチャ API 5 0 6、グラフィックス・ライブラリ 5 0 8、一時バッファ 5 1 0、ファイナル・バッファ 5 1 2、ビデオ・キャプチャ・コンポーネント 5 1 4 は、デバイス 2 0 0 のデータ・メモリ 2 0 2 中に格納されてもよく、図 5 に示されるように OS 2 1 2 に駐在したり、OS 2 1 2 で実行されたりしてもよい。

【 0 1 1 2 】

例えば、グラフィックス・エンジン 5 0 4 は、OGRE、Irrlicht 3D エンジン、Visual 3D ゲーム・エンジン、Truevision 3D エンジン、Crystal Space エンジン、Unity 3D ゲーム・エンジン、Unreal Engine、Cocos 2D エンジンなど、第三者からライセンス供与されたグラフィックス・エンジンでもよい。グラフィックス・ライブラリ 5 0 8 は、OpenGL API、Direct 4D API、EAGL API、EGL API など、クロスプラットフォームのグラフィックス・ライブラリ API を介して実装されてもよい。

【 0 1 1 3 】

さらに、グラフィックス・ドライバ 5 1 6 は、デバイス 2 0 0 のカーネル中のソフトウェア機能であってもよい。グラフィックス・ドライバ 5 1 6 は、システム供給のドライバ・コンポーネント、ベンダー供給のドライバ・コンポーネント、またはこれらの組み合わせであってもよい。グラフィックス・ドライバ 5 1 6 は、グラフィックス・ハードウェア 5 1 8 を介してディスプレイ 5 2 0 にインターフェース接続される。グラフィックス・ハードウェア 5 1 8 は例えば、デバイス 2 0 0 の集積グラフィックス・チップセット、または専用グラフィックス処理ユニット (GPU) であってもよい。ディスプレイ 5 2 0 は例えば、デバイス 2 0 0 のタッチ・スクリーン・ディスプレイなどのピクセル・ディスプレイであってもよい。

【 0 1 1 4 】

ある例示的实施形態において、ソフトウェア・アプリケーション 5 0 2 は、ゲームプレイの間、デバイス 2 0 0 のグラフィカル表示機構を介して見ることができるグラフィカル画像データの生成を必要とするゲーム・アプリケーションである。さらに、デバイス 2 0 0 のユーザは、この例示的实施形態において、ゲームプレイの間に生成されたグラフィカル画像データをキャプチャすることを選ぶ。

【 0 1 1 5 】

グラフィカル画像データを生成するために、ソフトウェア・アプリケーション 5 0 2 は、グラフィックス・エンジン 5 0 4 に対し一つ以上の関数コールを行う。グラフィックス・エンジン 5 0 4 は、ソフトウェア・アプリケーション 5 0 2 とグラフィックス装置との間のインターフェースとしての機能を果たす。

【 0 1 1 6 】

グラフィックス・エンジン 5 0 4 は、少なくとも部分的にソフトウェア・アプリケーション 5 0 2 中に実装されてもよい。あるいは、グラフィックス・エンジン 5 0 4 は、全体がソフトウェア・アプリケーション 5 0 2 とは別のルーティンとして実装されてもよい。グラフィックス・エンジン 5 0 4 は、ソフトウェア・アプリケーション 5 0 2 の作成者に

よって生成されるか、もしくは第三者からライセンス供与されてもよい。

【0117】

ここで、当然ながら、ソフトウェア・アプリケーション502および/またはグラフィックス・エンジン504は、ビデオキャプチャAPI506と統合するために明示的に修正される。一つの例において、グラフィックス・エンジン504は、グラフィックス・エンジン504がオープンソース・グラフィックス・エンジンの場合、ソフトウェア・アプリケーション502の作成者によって修正されてもよい。別の例では、グラフィックス・エンジン504は、グラフィックス・エンジン504の作成者によって修正されてもよい。

【0118】

この実装において、グラフィックス・エンジン504は、ビデオキャプチャAPI506を呼び出すために明示的に修正され、該APIは、グラフィックスキャプチャ機能を提供する。結果として、グラフィックス・エンジン504は、ソフトウェア・アプリケーション502から関数コールを受信するのに応じて、ビデオキャプチャを可能にするためビデオキャプチャAPI506に関数コールを行い、グラフィックス・ライブラリ508宛てに意図した特定の関数コールを、ビデオキャプチャAPI506に明示的に送信する。

【0119】

グラフィカル画像データをキャプチャするために、ビデオキャプチャAPI506は、関数コールがビデオキャプチャAPI506からグラフィックス・ライブラリ508に送信される前にこれらの関数コールを修正してもよい。それによって、SDKが、グラフィックス・エンジン504および/またはグラフィックス・ライブラリ508によって生成されたグラフィカル画像データにアクセス、読み取り、コピー、および/または修正できるようにされてもよい。

【0120】

さらにSDKは、一時バッファ510および/またはファイナル・バッファ512から、グラフィックス・エンジン504および/またはグラフィックス・ライブラリ508によって生成されたグラフィカル画像データのコピーを作成するよう動作可能であってもよい。特定の関数コールがビデオキャプチャAPI506を通り過ぎるとき、ビデオキャプチャAPI506は、ビデオ・キャプチャ・コンポーネント514が一時バッファ510および/またはファイナル・バッファ512からグラフィカル画像データをコピーできるような方法で、グラフィックス・ライブラリ508を制御するため、特定の関数コールがグラフィックス・ライブラリ508に送信される前にその特定の関数コールを修正する。

【0121】

グラフィックス・エンジン504が単一のバッファのみを使用している例において、その一時バッファ510は、ファイナル・バッファ512としても使われ、関数コールに対しSDKが行った、または行う修正によって、ビデオ・キャプチャ・コンポーネントが、当該バッファ中のグラフィックス・ライブラリ508によって生成または修正されたデータを読み取れてもよい。グラフィックス・エンジン504が当該バッファ中の画像フレームの描画を完了したとき、該グラフィックス・エンジン504はビデオキャプチャAPI506を呼び出し、次いで、該APIが、完成した画像フレームをユーザに提示するため、グラフィックス・ライブラリ508を呼び出す。それに応じて、ビデオキャプチャAPI506は、ビデオ・キャプチャ・コンポーネント514が、デバイス・メモリの他の場所にファイナル・バッファ512のコピーを格納したことを確かにし、その後、グラフィックス・ライブラリ508は、ユーザにファイナル・バッファ512を提示するために、グラフィックス・ドライバ516を呼び出す。別の実施形態において、ビデオ・キャプチャ・コンポーネント514は、画像フレームが完全に描画される前に一時バッファ510またはファイナル・バッファ512のコピーを格納し、すなわち、ユーザ・インタフェース・エレメントなど特定のグラフィカル・エレメントが当該対応バッファ中に生成される前に、グラフィカル画像データをキャプチャすることができる。

【0122】

グラフィックス・エンジン 504 が、複数のバッファ、すなわち複数の一時バッファ 510 を使用する別の例において、グラフィックス・ライブラリ 508 は、グラフィカル画像データをそれら一時バッファ 510 に書き込む。グラフィックス・エンジン 504 が、一時バッファ 510 中の画像フレームの描画を完了したとき、グラフィックス・エンジン 504 はビデオキャプチャ API 506 を呼び出し、次いで、該 API が、完成した画像フレームをユーザに提示するため、グラフィックス・ライブラリ 508 を呼び出す。それに応じて、ビデオキャプチャ API 506 は、グラフィックス・ライブラリ 508 が、一時バッファ 510 からグラフィカル画像データをファイナル・バッファ 512 の中に合成したことを確かにし、このデータはビデオ・キャプチャ・コンポーネント 514 による読み取りが可能である。合成が終わったならば、ビデオ・キャプチャ・コンポーネント 514 が、ファイナル・バッファ 512 からの合成済みグラフィカル画像データのコピーを、デバイス・メモリ中の別の場所に格納することになり、その後、グラフィックス・ドライバ 516 がそのファイナル・バッファ 512 をユーザに提示することになる。別の実施形態において、ビデオ・キャプチャ・コンポーネント 514 は、フレームが完成する前に一時バッファ 510 またはファイナル・バッファ 512 のコピーを格納する、すなわち、ユーザ・インタフェース・エレメントなど特定のグラフィカル・エレメントが生成される前に、グラフィカル画像データを キャプチャする ように動作可能である。

【0123】

完成した画像フレームを受信するのに応じ、グラフィックス・ドライバ 516 は、グラフィックス・ハードウェア 518 および / またはディスプレイ 520 を制御し、完成した画像フレームをユーザに提示する。

【0124】

一方、ビデオ・キャプチャ・コンポーネント 514 は、一時バッファ 510 またはファイナル・バッファ 512 のいずれかに書き込まれたグラフィカル画像データを キャプチャし、そのグラフィカル画像データをメモリ 202 中に、またはデバイス 200 中の別の内部ストレージの中にローカルに格納するよう動作可能であるか、あるいは、例えば、ビデオ・エンコーダ・ソフトウェアまたはハードウェアを使ってグラフィカル画像データのフレーム群をビデオ・ファイルの中に圧縮するため、グラフィカル画像データをさらに処理するよう動作可能である。これにより、グラフィカル画像データは、例えばソフトウェア・アプリケーション 502 の動作のビデオ記録を提供するため、例えば後の時点での画像フレーム群の再生に使われてもよい。

【0125】

ビデオ・キャプチャ・コンポーネント 514 は、ファイナル・バッファ 512 のコンテンツを、デバイス 200 から空間的に遠隔のデータ・ストアにアップロードおよび / またはストリームするよう動作できるという利点があってもよい。こうしたコンテンツは例えば、ゲームプレイの経験を共有する目的のため、データ・ストアから一つ以上の他のユーザのデバイスにストリームされてもよいが、この目的に限らない。

【0126】

さらに SDK は、ビデオキャプチャ API 506 およびビデオ・キャプチャ・コンポーネント 514 を含め、ソフトウェア・アプリケーション 502 の中に統合されてその一部分となるように構成されてもよい。ビデオキャプチャ API 506 およびビデオ・キャプチャ・コンポーネント 514 は、例えば、ソフトウェア・アプリケーション 502 がオブジェクト・コード・ライブラリと一緒にリンクされている場合、ソフトウェア・アプリケーション 502 のアプリケーション・バイナリ・インターフェース (ABI) に含まれてもよい。こうしたリンク付けは、例えば、実行可能ファイルが生成される場合、またはソフトウェア・アプリケーション 502 がランタイムで使われる場合は常に、有利に行われる。その結果、ビデオキャプチャ API 506 およびビデオ・キャプチャ・コンポーネント 514 は、ソフトウェア・アプリケーション 502 の一部となってもよい。

【0127】

さらにソフトウェア・アプリケーション 502 は、SDK によって提供されるグラフィ

ックスキャプチャ機能をユーザが「有効」または「無効」にできるよう動作可能であってもよい。

【0128】

次いで、ソフトウェア・アプリケーション502が、グラフィカル画像データを生成するためにグラフィックス・エンジン504を呼び出すとき、グラフィックス・エンジン504は、ビデオキャプチャAPI506を明示的に呼び出す。それに続いて、ビデオキャプチャAPI506は、ビデオ・キャプチャ・コンポーネント514が、生成されたグラフィカル画像データに対して、例えばこうしたデータを読み取りコピーおよび修正するフルアクセスを有し、これにより、グラフィックス・エンジン504および/またはグラフィックス・ライブラリ508によって生成されたグラフィカル画像データのキャプチャを可能にするように、グラフィックス・ライブラリ508を設定するため、関数コールが通り過ぎる前にそれらを修正する。

【0129】

ビデオキャプチャAPI506の実行は、ビデオキャプチャAPI506自体がグラフィカル画像データを生成するのではないので、測定可能なほどのパフォーマンス低下につながることはないという利点がある。ビデオ・キャプチャ・コンポーネント514によるグラフィカル画像データのキャプチャ、およびそのビデオ・フォーマットへの格納または圧縮がある程度パフォーマンス低下をもたらす可能性はあるが、これは、ソフトウェア・アプリケーション502の実行に起因するパフォーマンス低下と比較すると実質的に無視できる。

【0130】

図5は単なる例であって、添付の特許請求の範囲を不当に限定するものではない。当然ながら、グラフィックス装置に対する具体的指定は、該グラフィックス装置を、特定の数、種類、配置のモジュール、ルーティン、ソフトウェア製品、コンポーネントおよび/またはAPIのグラフィックス装置に限定すると解釈されるべきではない。当業者は、本開示の実施形態の多くの変形、代替、および修正を認識していよう。

【0131】

例えば、ゲームプレイの間に生成されたグラフィカル画像データとともに、ビデオおよび/またはデジタル・オーディオ・データをキャプチャするために、グラフィックス装置と類似の仕方でビデオ装置がデバイス200に実装されてもよい。グラフィカル画像データとともにデジタル・オーディオ・データから成るビデオ・ファイルは、次いで、遠隔のサーバに有利にアップロードされ、および/または他のユーザと共有される。

【0132】

図6は、本開示のある実施形態による、グラフィカル画像データをキャプチャするSDKを使用する方法のステップの図である。本方法は、論理流れ図のステップの集合として示され、該流れ図は、ハードウェア、ソフトウェア、またはその組み合わせ中に実装可能なステップのシーケンスを表している。

【0133】

前述したように、SDKは、デバイス200などのデバイス中で用いられるように構成されている。このデバイスは、デバイスのグラフィカル表示機構を介して見ることができるグラフィカル画像データを生成する、一つ以上のソフトウェア・アプリケーションを実行するよう動作可能なコンピューティング・ハードウェアを含む。

【0134】

ステップ602で、SDKは、デバイスのコンピューティング・ハードウェアで、ソフトウェア・アプリケーションと同時に実行されるように構成される。ステップ602によれば、SDKは例えば、ソフトウェア・アプリケーションが、グラフィックス・エンジンおよび/またはグラフィックス・ライブラリに対し一つ以上の関数コールを行ったときに起動されてもよい。

【0135】

ステップ604で、SDKは、ソフトウェア・アプリケーションが行うグラフィックス

・エンジンおよび／またはグラフィックス・ライブラリへの関数コールをインターセプトするよう動作する。これに換えてまたはこれに加えて、図5を参照して説明した実施形態においては、関数コールはビデオキャプチャAPI 506に明示的に送信される。

【0136】

ステップ604によれば、SDKは、グラフィックス・エンジンおよび／またはグラフィックス・ライブラリに対する実質上全ての関数コールがSDKを通過するように、グラフィックス・エンジンおよび／またはグラフィックス・ライブラリをラップするように動作してもよい。

【0137】

次にステップ606で、前述のようにSDKは、関数コールがグラフィックス・エンジンおよび／またはグラフィックス・ライブラリによって受信される前に、これら関数コールを修正するよう動作する。これによって、SDKが、グラフィックス・エンジンおよび／またはグラフィックス・ライブラリによって生成されたグラフィカル画像データにアクセス、読み取り、コピー、および／または修正できる。あるいは、グラフィックス・エンジンおよび／またはグラフィックス・ライブラリの動作を制御するために、任意の方法で関数コールを修正できてよい。

【0138】

ステップ606によれば、SDKは、ビデオ・キャプチャ・コンポーネント314、414、514が、グラフィックス・エンジンおよび／またはグラフィックス・ライブラリによって生成されたグラフィカル画像データにアクセス、読み取り、コピー、および／または修正できるようにグラフィックス・エンジンおよび／またはグラフィックス・ライブラリを設定してもよい。あるいは、グラフィックス・エンジンおよび／またはグラフィックス・ライブラリの動作を制御するため、任意の方法で関数コールを修正できてよい。

【0139】

引き続き、ステップ608で、前述したように、SDKは、一時バッファ310、410、510中にまたはファイナル・バッファ312、412、512中に生成または修正されたグラフィカル画像データをキャプチャするよう動作する。ビデオ・キャプチャ・コンポーネントは、ビデオ・キャプチャ・コンポーネント314、ビデオ・キャプチャ・コンポーネント414、またはビデオ・キャプチャ・コンポーネント514の実装と同様な仕方で実装されてもよい。

【0140】

ステップ608で取り込まれたグラフィカル画像データは、デバイスの内部データ・メモリ中にローカルに格納することができる。

【0141】

さらに、本方法は、SDKが、ビデオ・キャプチャ・バッファのコンテンツを、デバイスから空間的に遠隔のデータ・ストアにアップロードおよび／またはストリームするよう動作するさらなるステップを含んでもよい。

【0142】

一方、ステップ610で、ファイナル・バッファ中の画像フレーム群が、デバイスのグラフィカル表示機構を介しユーザに提示される。

【0143】

ステップ608およびステップ610は例えば、同時に実施されてもよい。

【0144】

さらに、本方法は、SDKがソフトウェア・アプリケーションの中に統合されてその一体部分となるように構成される、任意ステップを含み、これが本開示の実施形態の初期設定であることが望ましい。

【0145】

ステップ602～610は、単なる例示であって、本特許請求の範囲から逸脱することなく、一つ以上のステップが追加された、一つ以上のステップが除去された、または一つ以上のステップが異なる順序で行われる、他の代替方法も提供することも可能である。

【 0 1 4 6 】

本開示の諸実施形態は、持続性（非一時的）機械可読データ・ストレージ媒体に格納されたソフトウェア製品を提供し、このソフトウェア製品は、図6に関連させて説明した方法を実施するため、デバイス200のコンピューティング・ハードウェアで実行が可能である。該ソフトウェア製品は、例えば、「App store」などの、ソフトウェア・アプリケーション・ストアからデバイス200にダウンロードが可能であってもよい。

【 0 1 4 7 】

本開示の諸実施形態は、以下に限らないが、ユーザが、ゲームプレイなどのソフトウェア・アプリケーションを用いる間に生成されるグラフィカル画像データを、ユーザの携帯デバイスのパフォーマンスを実質的に低下させることなくキャプチャすることができるようにすることと、単にSDKすなわち「リンク・ライブラリ」をソフトウェア・アプリケーションのABIに含めることを除きソフトウェア・アプリケーションおよび/またはグラフィックス・エンジンを修正するいかなる必要もなしに、グラフィカル画像データをキャプチャする機能を可能にすることと、を含めて様々な用途に可能に使用できる。こうしたSDKは、例えばユーザがグラフィカル画像データキャプチャを制御できるように、追加の機能を提供してもよく、こうした機能は、ユーザに供するためソフトウェア・アプリケーションの修正を必要としてもよい。さらに、前述のように、本開示の諸実施形態は、対応するオーディオ・コンテンツと組み合わせられた画像および/またはビデオ・コンテンツの組み合わせのキャプチャを可能にする。

【 0 1 4 8 】

前述した本開示の実施形態は、添付の特許請求の範囲によって定義された本開示の範囲から逸脱することなく修正が可能である。本開示の記載および特許請求の範囲に用いられた「包含する（including）」、「含む/備える（comprising）」、「組み込む（incorporating）」、「から成る（consisting of）」、「有する（have）」、「である（is）」などの表現は、非排他的な形で、すなわち、明示的に記載されていない事項、コンポーネント、またはエレメントの存在もあり得るとして、解釈されるものとする。また、単数形での記載は、複数にも関すると解釈すべきである。

【 0 1 4 9 】

最後に、単なる例示として、例示的实施形態をいくつか載せておく。これらは本願の優先権の基礎となる米国特許出願の明細書に最初に記載された独立請求項に記載されたものに一致している。

1. グラフィカル画像データをキャプチャするソフトウェア開発キット（SDK）であって、前記ソフトウェア開発キットは、デバイスのグラフィカル表示機構を介して見ることができるグラフィカル画像データを生成する、一つ以上のソフトウェア・アプリケーションを実行するよう動作可能なコンピューティング・ハードウェアを含む、前記デバイス中で用いられるよう構成され、

前記ソフトウェア開発キットは、前記一つ以上のソフトウェア・アプリケーションと同時に実行されるように構成され、

前記一つ以上のソフトウェア・アプリケーションは、前記グラフィカル画像データを生成するために、グラフィックス・エンジンおよび/またはグラフィックス・ライブラリに対し、一つ以上の関数コールを行うよう動作可能であり、

前記ソフトウェア開発キットは、前記一つ以上の関数コールが前記グラフィックス・エンジンおよび/または前記グラフィックス・ライブラリによって受信される前に、前記一つ以上の関数コールを修正するよう動作可能であって、前記ソフトウェア開発キットが、前記グラフィックス・エンジンおよび/または前記グラフィックス・ライブラリによって生成された前記グラフィカル画像データにアクセス、読み取り、コピー、および/または修正すること、ならびに/あるいは、前記グラフィックス・エンジンおよび/またはグラフィックス・ライブラリの動作を制御するために、別途に任意の方法で前記関数コールを修正することを可能にする、

ソフトウェア開発キット。

2. グラフィカル画像データをキャプチャするソフトウェア開発キット（SDK）を使用する方法であって、前記ソフトウェア開発キットは、デバイスのグラフィカル表示機構を介して見ることができるグラフィカル画像データを生成する、一つ以上のソフトウェア・アプリケーションを実行するよう動作可能なコンピューティング・ハードウェアを含む、前記デバイス中で用いられるよう構成され、

前記方法は、

（a）前記ソフトウェア開発キットが前記一つ以上のソフトウェア・アプリケーションと同時に実行するよう構成することと；

（b）前記グラフィカル画像データを生成するため、グラフィックス・エンジンおよび／またはグラフィックス・ライブラリに対して一つ以上の関数コールを行うように、前記一つ以上のソフトウェア・アプリケーションを動作させることと；

（c）前記一つ以上の関数コールが前記グラフィックス・エンジンおよび／または前記グラフィックス・ライブラリによって受信される前に、前記一つ以上の関数コールを修正するように、前記ソフトウェア開発キットを動作させることであって、前記ソフトウェア開発キットが、前記グラフィックス・エンジンおよび／または前記グラフィックス・ライブラリによって生成された前記グラフィカル画像データにアクセス、読み取り、コピー、および／または修正すること、ならびに／あるいは、前記グラフィックス・エンジンおよび／またはグラフィックス・ライブラリの動作を制御するために、別途に任意の方法で前記関数コールを修正することを可能にする、前記動作させることと；

を含む、方法。

【誤訳訂正2】

【訂正対象書類名】図面

【訂正対象項目名】図6

【訂正方法】変更

【訂正の内容】

【 図 6 】

