

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6847033号
(P6847033)

(45) 発行日 令和3年3月24日(2021.3.24)

(24) 登録日 令和3年3月4日(2021.3.4)

(51) Int.Cl. F I
HO 4 N 19/467 (2014.01) HO 4 N 19/467
HO 4 N 19/85 (2014.01) HO 4 N 19/85
HO 4 N 19/94 (2014.01) HO 4 N 19/94

請求項の数 15 (全 14 頁)

<p>(21) 出願番号 特願2017-519285 (P2017-519285) (86) (22) 出願日 平成27年10月9日 (2015.10.9) (65) 公表番号 特表2017-536740 (P2017-536740A) (43) 公表日 平成29年12月7日 (2017.12.7) (86) 国際出願番号 PCT/US2015/054921 (87) 国際公開番号 W02016/057908 (87) 国際公開日 平成28年4月14日 (2016.4.14) 審査請求日 平成30年9月18日 (2018.9.18) (31) 優先権主張番号 62/062, 211 (32) 優先日 平成26年10月10日 (2014.10.10) (33) 優先権主張国・地域又は機関 米国 (US)</p> <p>前置審査</p>	<p>(73) 特許権者 591016172 アドバンスト・マイクロ・デバイス・ インコーポレイテッド ADVANCED MICRO DEVI CES INCORPORATED アメリカ合衆国 95054 カリフォル ニア州、 サンタ クララ、 オーガスティ ンドライブ 2485 (74) 代理人 100108833 弁理士 早川 裕司 (74) 代理人 100111615 弁理士 佐野 良太 (74) 代理人 100162156 弁理士 村雨 圭介</p>
--	--

最終頁に続く

(54) 【発明の名称】 ハイブリッドブロックベースの圧縮

(57) 【特許請求の範囲】

【請求項 1】

ハードウェアでサポートされた圧縮形式を使用してテクスチャのブロックベースの圧縮を行う方法であって、

テクスチャを複数のブロックに分割することであって、各ブロックは、RGB色空間内の2つのエンドポイントを有するデータを含み、前記データは、エンドポイント間の線上の画素値を表す複数のインデックスを含む、ことと、

ブロック毎に、当該ブロックに使用する変換を決定することと、

各エンドポイントの上位ビットが同じ場合に、前記変換の少なくとも1つの特性を、参照コンポーネント値を表すのに利用可能な複数のビットに符号化することであって、前記複数のビットは、前記エンドポイントの上位ビットである、ことと、

前記ブロックを圧縮することと、

を含む、方法。

【請求項 2】

第1のブロックを符号化するのに使用される前記複数のビットの数は、第2のブロックを符号化するのに使用される前記複数のビットの数と異なる、

請求項1の方法。

【請求項 3】

前記特性は、スケール因子及び指数のうち1つを表す、

請求項1の方法。

【請求項 4】

画像を復元する方法であって、

前記画像の選択された領域に関連付けられた複数のブロックを復号化することであって、各ブロックは、RGB色空間内の2つのエンドポイントを有するデータを含み、前記データは、エンドポイント間の線上の画素値を表す複数のインデックスを含む、ことと、

ブロック毎に、当該ブロックに使用する変換の特性を取得することであって、前記特性は、各エンドポイントの上位ビットが同じ場合に、参照コンポーネント値を表すのに利用可能な複数のビットに符号化され、前記複数のビットは、前記エンドポイントの上位ビットである、ことと、

前記ブロックを復元することと、

前記特性を使用して、前記ブロックに適用する前記変換を取得することと、

前記変換を適用することと、

を含む、方法。

10

【請求項 5】

第1のブロック内で前記特性を符号化するのに使用される前記複数のビットの数は、第2のブロック内で前記特性を符号化するのに使用される前記複数のビットの数と異なる、請求項4の方法。

【請求項 6】

ハードウェアでサポートされた圧縮形式を使用してテクスチャのブロックベースの圧縮を行うシステムであって、

第1の圧縮形式をサポートするように構成されたプロセッサと、

エンコーダと、を備え、

前記エンコーダは、

ブロック毎に、前記第1の圧縮形式によってサポートされないデータを含むブロックから生じるブロックに使用する変換の特性を決定することであって、各ブロックは、RGB色空間内の2つのエンドポイントを有するデータを含み、前記データは、エンドポイント間の線上の画素値を表す複数のインデックスを含む、ことと、

各エンドポイントの上位ビットが同じ場合に、前記特性を、参照コンポーネント値を表すのに利用可能な複数のビットに符号化することであって、前記複数のビットは、前記エンドポイントの上位ビットである、ことと、

前記ブロックを圧縮することと、を行うように構成されている、

システム。

20

30

【請求項 7】

第1のブロックを符号化するのに使用される前記複数のビットの数は、第2のブロックを符号化するのに使用される前記複数のビットの数と異なる、

請求項6のシステム。

【請求項 8】

前記圧縮されたブロックを復元し、

前記特性を使用して、前記ブロックに適用する前記変換を取得し、

前記変換を適用する、ことを行うように構成されたデコーダをさらに備える、

請求項6のシステム。

40

【請求項 9】

ハードウェアでサポートされた圧縮形式を使用してテクスチャのブロックベースの圧縮を実行するための命令のセットであって、コンピューティングデバイスによって実行される命令のセットを記憶するコンピュータ可読記憶媒体であって、

前記命令のセットは、前記コンピューティングデバイスによって実行されると、

テクスチャを複数のブロックに分割することであって、各ブロックは、RGB色空間内の2つのエンドポイントを有するデータを含み、前記データは、エンドポイント間の線上の画素値を表す複数のインデックスを含む、ことと、

ブロック毎に、当該ブロックに使用する変換を決定することと、

50

各エンドポイントの上位ビットが同じ場合に、前記変換の少なくとも1つの特性を、参照コンポーネント値を表すのに利用可能な複数のビットに符号化することであって、前記複数のビットは、前記エンドポイントの上位ビットである、ことと、

前記ブロックを圧縮することと、

をコンピューティングデバイスに行わせる命令を含む、

コンピュータ可読記憶媒体。

【請求項10】

第1のブロックを符号化するのに使用される前記複数のビットの数は、第2のブロックを符号化するのに使用される前記複数のビットの数と異なる、

請求項9のコンピュータ可読記憶媒体。

10

【請求項11】

前記コンピューティングデバイスによって実行されると、

前記ブロックを復元することと、

前記特性を使用して、前記ブロックに適用する前記変換を取得することと、

前記変換を適用することと、

を前記コンピューティングデバイスに行わせる命令をさらに備える、

請求項9のコンピュータ可読記憶媒体。

【請求項12】

ハードウェアでサポートされた圧縮形式を使用してテクスチャのブロックベースの圧縮を行う方法であって、

20

テクスチャを複数のブロックに分割することであって、各ブロックは、同じ色空間内の2つのエンドポイントを有するデータを含む、ことと、

ブロック毎に、当該ブロックに使用する変換を決定することと、

各エンドポイントの上位ビットが同じ場合に、前記変換の少なくとも1つの特性を、参照コンポーネント値を表すのに利用可能な複数のビットであって、各エンドポイントの上位ビットである複数のビットに符号化することと、

前記ブロックを圧縮することであって、第1のブロックを符号化するのに使用される前記複数のビットの数は、第2のブロックを符号化するのに使用される前記複数のビットの数と異なる、ことと、

を含む、方法。

30

【請求項13】

画像を復元する方法であって、

前記画像の選択された領域に関連付けられた複数のブロックを復号化することと、

同じ色空間内の2つのエンドポイントを有するデータを含むブロック毎に、当該ブロックに使用する変換の特性を取得することであって、前記特性は、参照コンポーネント値を表すのに利用可能な複数のビットであって、前記2つのエンドポイントの上位ビットである複数のビットに符号化される、ことと、

前記ブロックを復元することと、

前記特性を使用して、前記ブロックに適用する前記変換を取得することと、

前記変換を適用することであって、第1のブロック内で前記特性を符号化するのに使用される前記複数のビットの数は、第2のブロック内で前記特性を符号化するのに使用される前記複数のビットの数と異なる、ことと、

40

を含む、方法。

【請求項14】

ハードウェアでサポートされた圧縮形式を使用してテクスチャのブロックベースの圧縮を行うシステムであって、

第1の圧縮形式をサポートするように構成されたプロセッサと、

エンコーダと、を備え、

前記エンコーダは、

同じ色空間内の2つのエンドポイントを有するデータを含むブロック毎に、前記第1の

50

圧縮形式によってサポートされないデータを含むブロックから生じるブロックに使用する変換の特性を決定することと、

前記特性を、参照コンポーネント値を表すのに利用可能な複数のビットであって、前記2つのエンドポイントの上位ビットである複数のビットに符号化することと、

前記ブロックを圧縮することであって、第1のブロックを符号化するのに使用される前記複数のビットの数は、第2のブロックを符号化するのに使用される前記複数のビットの数と異なる、ことと、を行うように構成されている、

システム。

【請求項15】

ハードウェアでサポートされた圧縮形式を使用してテクスチャのブロックベースの圧縮を実行するための命令のセットであって、コンピューティングデバイスによって実行される命令のセットを記憶するコンピュータ可読記憶媒体であって、

前記命令のセットは、前記コンピューティングデバイスによって実行されると、

テクスチャを複数のブロックに分割することと、

同じ色空間内の2つのエンドポイントを有するデータを含むブロック毎に、当該ブロックに使用する変換を決定することと、

前記変換の少なくとも1つの特性を、参照コンポーネント値を表すのに利用可能な複数のビットであって、前記2つのエンドポイントの上位ビットである複数のビットに符号化することと、

前記ブロックを圧縮することであって、第1のブロックを符号化するのに使用される前記複数のビットの数は、第2のブロックを符号化するのに使用される前記複数のビットの数と異なる、ことと、

をコンピューティングデバイスに行わせる命令を含む、

コンピュータ可読記憶媒体。

【発明の詳細な説明】

【技術分野】

【0001】

(関連出願の相互参照)

本願は、2014年10月10日に出願された米国仮特許出願第62/062,211号の優先権を主張し、その全ての開示を明示的に援用するものである。

【0002】

本開示は、概して、ハードウェアがサポートする圧縮形式を用いてテクスチャのブロックベースの圧縮を行う方法及び装置に関し、特に、ソフトウェアを用いて、既存のハードウェアでサポートされた圧縮形式と互換性のないデータの圧縮を可能にする方法及び装置に関する。

【背景技術】

【0003】

コンピュータグラフィックス処理システムは、通常、大部分の処理を行う1つ以上のグラフィックスプロセッシングユニット(GPU)を用いて、大量のデータを処理する。GPUは、レンダリングと呼ばれるプロセスにおいて、画像を構成する画素をそのコンポーネントの高レベルの記述から生成する。GPUは、通常、コンピューティング要素を用いて連続的なレンダリングを実行して、画素、テクスチャ及び幾何学的データを処理する。コンピューティング要素は、ラスタライザ、セットアップエンジン、カラーブレンダ、隠面除去及びテクスチャマッピングの機能を実行することができる。コンピューティング要素は、シェーダ、シェーダプロセッサ、シェーダアレイ、シェーダパイプ、シェーダパイプアレイ、シェーダパイプライン又はシェーダエンジンと呼ばれることが多い。「シェーダ」は、ソフトウェア命令(より正確には「シェーダプログラム」と呼ばれる)を実行するために使用される実際のハードウェアコンポーネント又はプロセッサを指す場合もある。シェーダプロセッサ又はシェーダプログラムは、データを読み出してレンダリングし、任意のタイプのデータの処理を実行することができる。

10

20

30

40

50

【 0 0 0 4 】

複雑なグラフィックシーンを生成することに関わる処理の多くは、テクスチャデータを伴う。テクスチャは、画像データのようなハイトマップデータ、画像データ、色データ若しくは透明度データ、粗さ/平滑度データ、反射率データ、ルックアップテーブル、又は、他のデータ等の様々なタイプのデータの何れかであってもよい。テクスチャは、鏡面反射のような表面のプロパティや、通常のマップやバンプマップのような細かい表面の詳細を含む多くのプロパティを指定することができる。時には、「テクセル (texel)」という用語は、「画素」が画像の素子であると同様に、「テクセル」がテクスチャの素子である技術分野において使用される。しかしながら、「テクセル」及び「画素」という用語は、よく区別することなく使用される（「テクセル」は、単に「画素」と呼ばれることがある）。したがって、何れの用語も、本開示内では区別することなく使用され得る。

10

【 0 0 0 5 】

現実的なコンピュータグラフィックスを提供するには、通常、高品質で詳細なテクスチャを多く必要とする。しかしながら、テクスチャを使用すると、大量のストレージスペースと、メモリからGPUへの帯域幅と、を消費する可能性があり、その結果、テクスチャは、通常圧縮され、ストレージスペース及び帯域幅の使用率が低下する。

【 0 0 0 6 】

したがって、テクスチャ圧縮は、一般的にグラフィックスハードウェア、特に3Dグラフィックスハードウェアで広く受け入れられる機能となっている。テクスチャ圧縮の目的は、グラフィックスシステムのストレージコストと帯域幅コストとを削減しながら、元のテクスチャの品質を可能な限り維持することである。多くのテクスチャ圧縮技術は、以下でさらに説明するように、エンドポイントと補間された画素値とを符号化するための様々なブロックベースの圧縮方法のうちの1つを使用する。DirectXで使用されるマイクロソフト（登録商標）BCn形式は、かかる方法の一例である。これらの形式と他の形式とをより柔軟に使用して、形式が、元々指定されたデータ以外の新しいタイプのデータ（例えば、ハイトマップデータ）の符号化を可能にする方法を開発することが望ましい。

20

【 0 0 0 7 】

より具体的には、新しいテクスチャ圧縮技術は、開発することと、ハードウェア形態で市場に提供することとに長い時間がかかる。ハードウェアベースの圧縮の或るバージョンから次のバージョンへの移行中に、ソフトウェア開発者は、既存の圧縮技術が不適切な結果をもたらすような、新しい圧縮技術を必要とする新しいタイプのデータを生成する。これは、これらの新しい技術を特定することと、これらを市場に実際に導入することと、の間にギャップを生成する。

30

【 0 0 0 8 】

したがって、カスタムデータを既存の圧縮技術を使用して符号化し、（画像全体にグローバルに適用するのではなく）ブロック毎に変化させて、ソフトウェアベースの復号化技術とハードウェアベースの復号化技術との柔軟なハイブリッドを開発するのを可能にすることによって、既存のハードウェアを使用する新しいテクスチャ圧縮の可能性を提供する方法が必要とされている。

【 発明の概要 】

40

【 課題を解決するための手段 】

【 0 0 0 9 】

本開示の一実施形態によれば、ハードウェアでサポートされた圧縮形式を使用してテクスチャのブロックベースの圧縮を行う方法が提供される。この方法は、テクスチャを複数のブロックに分割することと、ブロックと共に使用する変換であって、エラー量を最小化する変換をブロック毎に決定することと、前記変換の少なくとも1つの特性を、参照コンポーネント値を表すのに利用可能な複数のビットに符号化することと、ブロックを圧縮することと、を含む。本実施形態の一態様によれば、各ブロックは、RGB色空間内の2つのエンドポイントと、エンドポイント間の線上の画素値を表す複数のインデックスと、を有するデータを含み、複数のビットは、エンドポイントの上位ビットである。本態様の変

50

形例では、特性は、各エンドポイントの上位ビットが同じ場合に、複数のビットに符号化される。別の態様では、第1のブロックを符号化するのに使用される複数のビットの数は、第2のブロックを符号化するのに使用される複数のビットの数と異なる。さらに別の態様では、ブロックに関連付けられたデータは、ハードウェアがサポートする圧縮形式によって直接サポートされていない。さらに別の態様では、特性は、スケール因子及び指数のうち1つを表す。

【0010】

本開示の別の実施形態によれば、画像を復元する方法が提供される。この方法は、画像の選択された領域に関連付けられた複数のブロックを復号化することと、ブロックと共に使用する変換であって、エラー量を最小化する変換の特性をブロック毎に取得することと、特性は、参照コンポーネント値を表すのに利用可能な複数のビットに符号化される、ことと、ブロックを復元することと、特性を使用して、ブロックに適用する変換を取得することと、変換を適用することと、を含む。本実施形態の一態様によれば、各ブロックは、RGB色空間内の2つのエンドポイントと、エンドポイント間の線上の画素値を表す複数のインデックスと、を有するデータを含み、複数のビットは、エンドポイントの上位ビットである。本態様の変形例では、特性は、各エンドポイントの上位ビットが同じ場合に、複数のビットに符号化される。別の態様では、第1のブロック内で特性を符号化するのに使用される複数のビットの数は、第2のブロック内で特性を符号化するのに使用される複数のビットの数と異なる。

【0011】

本開示のさらに別の実施形態によれば、ハードウェアでサポートされた圧縮形式を使用してテクスチャのブロックベースの圧縮を行うシステムが提供される。このシステムは、第1の圧縮形式をサポートするように構成されたプロセッサと、エンコーダと、を備える。エンコーダは、ブロックと共に使用する変換であって、第1の圧縮形式によってサポートされないデータを含むブロックから生じるエラー量を最小化する変換の特性をブロック毎に決定し、特性を、参照コンポーネント値を表すのに利用可能な複数のビットに符号化し、ブロックを圧縮するように構成されている。本実施形態の一態様によれば、各ブロックは、RGB色空間内の2つのエンドポイントと、エンドポイント間の線上の画素値を表す複数のインデックスと、を有するデータを含み、複数のビットは、エンドポイントの上位ビットである。本態様の変形例では、特性は、各エンドポイントの上位ビットが同じ場合に、複数のビットに符号化される。別の態様では、第1のブロックを符号化するのに使用される複数のビットの数は、第2のブロックを符号化するのに使用される複数のビットの数と異なる。さらに別の態様では、圧縮されたブロックを復元し、特性を使用して、ブロックに適用する変換を取得し、変換を適用するように構成されたデコーダを備える。

【0012】

本開示のさらに別の実施形態によれば、ハードウェアでサポートされた圧縮形式を使用してテクスチャのブロックベースの圧縮を実行するための命令のセットであって、コンピューティングデバイスによって実行される命令のセットを記憶するコンピュータ可読記憶媒体が提供される。命令のセットは、コンピューティングデバイスによって実行されると、テクスチャを複数のブロックに分割することと、ブロックと共に使用する変換であって、エラー量を最小化する変換をブロック毎に決定することと、前記変換の少なくとも1つの特性を、参照コンポーネント値を表すのに利用可能な複数のビットに符号化することと、ブロックを圧縮することと、をコンピューティングデバイスに行わせる命令を含む。本実施形態の一態様によれば、各ブロックは、RGB色空間内の2つのエンドポイントと、エンドポイント間の線上の画素値を表す複数のインデックスと、を有するデータを含み、複数のビットは、エンドポイントの上位ビットである。本態様の変形例では、特性は、各エンドポイントの上位ビットが同じ場合に、複数のビットに符号化される。別の態様では、第1のブロックを符号化するのに使用される複数のビットの数は、第2のブロックを符号化するのに使用される複数のビットの数と異なる。さらに別の態様では、コンピューティングデバイスによって実行されると、ブロックを復元することと、特性を使用して、ブ

10

20

30

40

50

ロックに適用する変換を取得することと、変換を適用することと、をコンピューティングデバイスに行わせる命令を備える。

【0013】

実施形態は、以下の図面と共に以下の説明を考慮してより容易に理解される。図面において、同様の参照番号は同様の要素を表す。

【図面の簡単な説明】

【0014】

【図1】1つ以上の開示された実施形態を実装することができる例示的なデバイスのブロック図である。

【図2】RGB色空間内の線として表されるデータブロックの概念図である。

10

【図3】本開示による、テキストを圧縮する方法のフローチャートである。

【図4】本開示による、テキストを復元する方法のフローチャートである。

【発明を実施するための形態】

【0015】

簡潔には、一実施例では、ハードウェアでサポートされた圧縮形式を使用してテキストのブロックベースの圧縮を行う方法及び装置が提供される。方法は、テキストを複数のブロックに分割することと、ブロックと共に使用する変換であって、エラー量を最小化する変換をブロック毎に決定することと、変換の少なくとも1つの特性を、参照コンポーネント値を表すのに利用可能な複数のビットに符号化することと、ブロックを圧縮することと、を含む。したがって、本開示は、カスタムデータを既存のハードウェアベースの圧縮技術を使用して符号化し、（画像全体にグローバルに適用するのではなく）ブロック毎に変化させるアプローチを提供することによって、ソフトウェアベースの復号化技術とハードウェアベースの復号化技術との柔軟なハイブリッドを開発することを可能にし、既存のハードウェアを使用する新しいテキスト圧縮の可能性を提供する。

20

【0016】

図1は、1つ以上の開示された実施形態を実装することができる例示的なデバイス100のブロック図である。特定の実施形態では、デバイス100は、ゲームデバイス、携帯型デバイス、セットトップボックス、テレビ、携帯電話若しくはタブレット、又は、他のタイプのコンピュータである。デバイス100は、一般に、プロセッサ102と、1つ以上の入力デバイス104と、1つ以上の出力デバイス106と、1つ以上のストレージデバイス108と、1つ以上のメモリデバイス110と、を含む。

30

【0017】

プロセッサ102は、中央処理装置（CPU）、グラフィックスプロセッシングユニット（GPU）、同じダイ上に配置されたCPU及びGPU、又は、1つ以上のプロセッサコアを含むことができ、各プロセッサコアは、CPU又はGPUであってもよい。他のタイプの集積回路及びプロセッサも、デジタルシグナルプロセッサ（DSP）、アクセラレーテッドプロセッシングユニット（APU）、フィールドプログラマブルゲートアレイ（FPGA）等を含む本開示の態様を具現化することができる。

【0018】

入力104は、キーボード、キーパッド、タッチスクリーン/パッド、検出器、マイクロフォン、加速度計、ジャイロスコープ、生体スキャナ、又は、ネットワーク接続（例えば、ワイヤレスIEEE802信号の送信及び/又は受信のためのワイヤレスローカルエリアネットワークカード）のうち1つ以上を含むことができる。出力106は、ディスプレイ、スピーカ、プリンタ、触覚フィードバックデバイス、1つ以上の測定器、アンテナ、又は、ネットワーク接続（例えば、ワイヤレスIEEE802信号の送信及び/又は受信のためのワイヤレスローカルエリアネットワークカード）のうち1つ以上を含むことができる。

40

【0019】

ストレージ108は、例えばハードディスクドライブ、半導体ドライブ、光ディスク又はフラッシュドライブ等の固定又は取り外し可能なストレージコンポーネントを含むこと

50

ができる。メモリ 110 は、プロセッサ 102 と同じダイ上に配置されてもよいし、プロセッサ 102 とは別に配置されてもよい。メモリ 110 は、例えばランダムアクセスメモリ (RAM)、ダイナミック RAM 又はキャッシュ等の揮発性又は不揮発性のメモリを含むことができる。

【0020】

例えばデバイス 100 等のデバイス用の様々な GPU サポートのテクスチャ圧縮形式は、長年にわたって開発されており、この形式には、BC1 から BC7 までの 7 つの標準形式のセット (つまり、上記で言及したマイクロソフト (登録商標) BCn) が含まれる。これらの形式は、テクスチャマップのメモリ使用 (ストレージ及び帯域幅) を減らすために、現実的な 3D ゲーム等で広く使用されている。かかるアプリケーションでは、ディフューズカラー (diffuse color)、法線マップ (normal map)、鏡面ハイライト (specular highlight)、光沢 (gloss)、放射グロー (emissive glow) 等のより多くのテクスチャを要する高解像度のグラフィックスが必要である。期待される視覚的詳細を達成するためには、非常に大きなメモリ帯域幅を必要とする大量のデータをプロセッサに供給しなければならない。テクスチャ圧縮は、画像データをシェーダコアに提供するために必要なメモリ帯域幅を削減する。

10

【0021】

全ての BCn 形式によって可能になるテクスチャ圧縮は、ブロック圧縮、より具体的には、4x4 画素のブロックに基づいている。処理される各画像は、これらのブロックに分割され、ブロックは、形式に応じて 8 バイト又は 16 バイトのサイズに固定されている。この標準的なレイアウトとメモリ内のブロックの連続的な記憶とは、GPU がテクスチャの任意の部分を含むブロックをすばやく見つけてアクセスできるので、効率的な GPU レンダリングを可能にする。

20

【0022】

各ブロックは、画像全体のうち小さな領域を表している。多くの画像では、特定の小さな領域 (又はブロック) 内に非常に限られた色の変化が存在する。一般的に、ブロックは、単色の陰影又は 2 色間の勾配を含む。BCn 形式は、ブロック内の色の定義 (definition) を空間分布から分離させることによって、この事実を利用する。

【0023】

本開示の一実施形態では、例えば、マイクロソフト (登録商標) BC5 仕様を用いて、以下に説明する圧縮が、単一のコンポーネントハイトマップテクスチャに対して実行される。かかる実施形態では、BC5 を使用したハイトマップテクスチャの圧縮は、ハイトマップが単一のコンポーネントテクスチャであるため、復元 (及び圧縮中の対応する逆変換) 中に 2 コンポーネントテクスチャを 1 コンポーネントテクスチャに変換するような変換を必要とする。この場合、圧縮されるデータ及びハードウェアでサポートされる形式は、本開示の原理を適用することによって、異なる数のコンポーネント (チャンネル) を有することができる。このアプローチは、1 チャンネル (例えば、赤チャンネル (red channel)) からの上位ビットを使用して制御ビット (すなわち、変換の少なくとも 1 つの特性) を伝搬させることによって、ハイトマップテクスチャが 16 ビットのフルダイナミックレンジを使用しないという事実を利用する。符号化された特性は、最小化されたエラーを提供するブロックに使用する変換をエンコーダが取得するのを可能にする。

30

40

【0024】

一例として、BC5 デコーダの 2 コンポーネント出力が C1 及び C2 である (すなわち、8.6 形式の 2 つの符号なしチャンネル) と仮定すると、以下の表 1 に示す命令を、本開示の原理にしたがって実装することができる (以下、参照のために行を番号付けする)。

【表 1】

```

float transform(float C1,float C2){           (1)
static int mask1=(1<<7);                     (2)
static int mask2=(1<<6);                     (3)
static int scale1=4;                         (4)
static int scale2=8;                         (5)
static int scale3=32;                        (6)
static int bias1=64;                         (7)
static int bias2=32;                         (8)
static int bias3=32;                         (9)
if((int)C2&mask1==0){                        (10)
    return                                   (11)
        C1*256+(C2-bias1)*scale1           (12)
else if ((int)C2&mask2==0){                 (13)
    return                                   (14)
        C1*256+(C2-mask1-bias1)*scale2     (15)
else{                                       (16)
    return                                   (17)
        C1*256+(C2-(mask1+mask2)-bias3)*scale3 (18)
}                                           (19)
}                                           (20)

```

(2) 行目及び(3) 行目は、変換パラメータを復号するためのマスクを提供する。この例では、これらは、実装するパラメータのコードを提供するC2のビットである(すなわち、ビット7及びビット6は、パラメータコード情報を含むことができる)。(4) 行目から(9) 行目は、各変換の3つのスケールと、各変換の3つのバイアスと、の例示的な定義を提供する。(10) 行目から(20) 行目は、変換を復号する(すなわち、適用する変換のコードを決定する)ための命令を提供する。本実施形態では、コードは可変長である。ビット7が0の場合には、変換1(何でもよい)を適用する必要がある。これは、(12) 行目に反映される。本実装形態は、変換のコードを符号化するためにC2の1ビット(本例では、C2の最上位ビット(MSB))のみを使用するので、他の7ビットは、他の情報を伝えるために利用可能である。MSB=0の場合には、マスクする必要がない。通常、コードを復号した後はビットがマスクされるが、当該ビットが既にゼロである場合には、その必要がない。MSB=1の場合には、命令は次のビットを評価する。これは、(13) 行目及び(14) 行目に反映される。第2のビットがゼロである場合には、第2の変換が適用される((15) 行目及び(16) 行目を参照)。最後に、両方のビットが1である場合には、第3の変換を使用して、両方のビットがマスクされる((18) 行目を参照)。

【0025】

各ブロックの画素又はテクセルは、ブロックのカラーパレットへのインデックスとしてコード化され、このパレットは、通常、上述したように極めて限定されている。パレットが非常に小さいため、画素当たりで必要とされるビット数が少ない。パレットは、その全ての色が、2つのエンドポイント間に延在するRGB色空間内の線分(又は補間軸)に沿う等間隔の点に当てはまる、と仮定することによって圧縮される。この仮定を使用すると、線のエンドポイントのみを記憶する必要があり、他の色は、2つのエンドポイントを異なる割合で混ぜ合わせることによって再構成される。

【0026】

図2は、RGB色空間内の上述した線分200の概念図である。この例は例示的なもの

であるが、本開示の一実施形態は、例えばマイクロソフト（登録商標）BC5仕様を用いるハードウェアによってサポートされた2コンポーネント形式を用いて1コンポーネント（スカラー）入力データ圧縮を使用する、ということに留意すべきである。図示したように、RGB色空間は、3Dデカルト座標系の単位立方体サブセットを含み、R、G、B軸を用いて原色（赤、緑及び青）をマッピングし、原点を黒（光なし）でマッピングする。したがって、モデルが単位立方体であるため、各軸は、0から1の間の強度範囲を有する。線200は、エンドポイント202と、エンドポイント204と、を含む。本例では、線200は、線形補間によって線200に沿って等間隔に配置された2つの中間点又は画素値206をさらに含む。実装に応じて、当業者によって理解されるように、画素値206の数は、2つより大きくてもよいし、2つ未満であってもよい。画素値206は、線200に対応するブロックに、3ビットのインデックスの形式のデータとして含まれる。本開示にしたがって、3ビットよりも多く又は3ビット未満の各画素値のインデックスを使用可能であることを理解すべきである。

10

【0027】

本開示によれば、エンドポイント202、エンドポイント204からいくつかのビットの精度を犠牲にすることによって、追加のデータをブロック毎に符号化することが可能であり、これにより、デコーダが、カスタムデータ、又は、既存のハードウェア圧縮技術が対応するように設計されていないデータに対して特別な処理を行うのを可能にする、ということが認識される。例示のみを目的として、8ビット値で特定されるエンドポイント202、エンドポイント204と、3ビットインデックスを用いて選択された画素値206と、を有するデータのチャンネル（赤チャンネル）を符号化する単一コンポーネント形式を仮定する。ブロック内の任意の画素に対して可能な色は、以下の表2に示すように得られる。

20

【表2】

```
red_0=endpoint_0;
red_1=endpoint_1;
red_2=(6*red_0+1*red_1)/7.0f; //bit code 010
red_3=(5*red_0+2*red_1)/7.0f; //bit code 011
red_4=(4*red_0+3*red_1)/7.0f; //bit code 100
red_5=(3*red_0+4*red_1)/7.0f; //bit code 101
red_6=(2*red_0+5*red_1)/7.0f; //bit code 110
red_7=(1*red_0+6*red_1)/7.0f; //bit code 111
```

30

【0028】

3ビットインデックスは、ブロック内の各画素について、どの色を使用するのかを選択するために用いられる。返される最終的な値は、0から1の範囲に含まれる。

【0029】

画素値206は線形補間によって得られるので、各エンドポイント202、204の上位ビットが同じ場合には、画素値206の上位ビットはブロックで一定である。また、各エンドポイント202、204の上位ビットが同じ場合には、これらの上位ビットの値は、データの低位ビットの何れかの補間値に影響を与えない。この結果、いくつかの上位ビット（例えば、エンドポイント__0及びエンドポイント__1の各々からの3ビット）は、デコーダによる特別な処理のために予約されてもよい。予約されたビットの数は、ブロック毎に異なってもよいことを理解すべきである。別の言い方をすれば、これらのビットを使用して、ブロックの特別なデータを符号化することができる。

40

【0030】

カラー値をテクスチャから取り出した後に、以下の表3に示すソフトウェア復号ステップを、上述したハードウェア復号の後に追加してもよい。

【表 3】

```

color=returned_color_from_hardware_decoder; //range(0->1.0)
color=255.0 //expand the color to match the original endpoint precision
reservedBitmask=0xe0; //Binary mask 11100000
intColor=(int)color; //get rid of the fractional bits
specialBlockData=(intColor & reservedBitmask)>>5; //Extract special block bits
color=(float)intColor-(specialBlockData<<5)
color/=31.0; //rescale the remaining data block int the 0->1range

```

【 0 0 3 1】

10

かかるソフトウェア復号化ステップは、（ 8 つの異なる変更を符号化に提供するのに十分な）3 ビットのデータをシェーダに伝える。本開示の一実施形態では、データのビットは、データに適用されるスケール因子又は指数を表す等して、復号されたデータを変換するのに用いられてもよい。当業者には明らかであるように、データのビットを、多くの他の用途に使用することができる。上述した方法は、マルチコンポーネント圧縮形式で使用するために、当業者によって容易に変更され得る（すなわち、ビットは、上述した方法と同じ方法で、独立したコンポーネントのうち任意のものに符号化することができる）ことを理解すべきである。

【 0 0 3 2】

20

上記の方法では、カスタムデータを変換して、（カスタムデータを処理するように設計されていない）既存の圧縮技術からより良い結果を得ることができる。データは、圧縮時に変換され、圧縮後（つまり、復元及びレンダリング時）に元の状態に戻されなければならない。異なるブロック毎に変換が異なる場合があることを理解すべきである。このようにして、ブロック単位で最適であると決定された様々な技術を用いることによって、エラー低減及び品質向上を実現することができる。このようなブロック毎の変換を可能にするために、変換を各ブロックのソフトウェアデコーダに渡す必要がある。例えば、画像内の各ブロックについて、エラー量を最小にするために、ブロックと共に使用する変換が特定される。各ブロックと共に使用される変換を決定する方法のさらなる詳細は、2012年10月12日に出願された「REGION-BASED IMAGE COMPRESSION」という名称の同時係属中の米国特許出願第13/651,020号にて提供されており、その開示全体は、参照によって本明細書に明示的に組み込まれる。本開示の一実施形態では、上述したエンドポイントの上位ビットに変換を含むことによって、当該変換がデコーダに渡される。ブロックの符号化された画素をデコーダが受信すると、パラメータが画素のMSBに提供される。本明細書で説明するように、本変換は、2コンポーネント形式（BC5）のハードウェア復元からの2つのコンポーネントを、1コンポーネントにまとめることを含む。これに応じて、圧縮器は、単一コンポーネントデータを2コンポーネントデータに変換する。

30

【 0 0 3 3】

図3は、本開示の原理による、テクスチャを圧縮する方法300のフローチャートである。図示したように、ブロック302において、圧縮するテクスチャが選択される。ブロック304において、テクスチャは、（上述した4×4ブロック等の）複数のブロックに分割される。ブロック306において、現在のブロックが圧縮のために選択される。次に、ブロック308において、現在のブロックのエンドポイント202, 204の上位ビットが同じか否かが判別され、それにより、かかるエンドポイントが、本明細書で説明した変換で置換されるのを可能にする。上位ビットが同じ場合には、ブロック310において、現在のブロックと共に使用される変換が決定される。同じでない場合には、ブロック309において、データが丸められ、又は、符号化されたブロック内でビットが使用されるように「空間を生成する」ために変換される。ブロック310において、変換が必要でないと判別され、1つ以上のビットを使用して、残りの変換の必要がないと示すことができることに留意されたい。また、可変長コードを特定の実施形態で使用して、（少なくとも

40

50

1ビットが常に符号化に使用されるが)変換ビット数がブロック毎に異なるように変換を符号化してもよいことも理解されたい。

【0034】

ブロック310において、現在のブロックのエンドポイント202, 204の上位ビットが同じであり、変換パラメータが決定されると、ブロック312において、変換は、本発明の技術分野において周知の原理にしたがって、エンドポイント202, 204の上位ビットに符号化される。その後、ブロック314において、現在のブロックが圧縮される。次に、ブロック316において、全てのブロックが処理されたか否かが判別される。全てのブロックが処理された場合には、プロセスは、ブロック318において終了する。そうでない場合には、新たな現在のブロックがブロック306において選択され、方法の残りが繰り返される。

10

【0035】

図4は、本開示の原理による、テクスチャを復元する方法400のフローチャートである。図示したように、ブロック402において、図3の方法300にしたがって圧縮されたブロックが受信される。ブロック404において、ブロックが復元される。ブロック406において、復元されたブロックが変換を含むか否かが判別される。含まれていない場合には、ブロックは、ブロック408において復号される。ブロック406の判別において、復元されたブロックが変換を含む場合、ブロック410において、変換を使用して、ブロックに対応するデータを変換する。その後、ブロック408において、ブロックが復号される。

20

【0036】

ブロック412において、全ての圧縮ブロックが処理されたか否かが判別される。処理された場合、プロセスは、ブロック414において終了する。そうでない場合、ブロック404において、処理のための次のブロックが復元され、上述した処理が繰り返される。本明細書において、「変換」という用語は、極めて一般的な意味で使用され、任意のデータ処理を含むことを理解すべきである。

【0037】

上述した方法では、ブロック毎に用いられる圧縮方法に対して細かい修正を行って、より複雑な圧縮方法の実装を可能にし得る。したがって、本開示の方法及び装置は、より複雑な圧縮技術をサポートするように設計されていない既存のグラフィックスハードウェア上で、比較的短時間に、より複雑な圧縮技術を使用するのを可能にする広範囲な可能性のあるアプリケーションを促進する。このようにして、ハイトマップデータ又は現在利用可能なハードウェア圧縮技術が無い他のデータ等の新しいタイプのデータ(又はカスタムデータ)を、既存のハードウェアを使用して処理することができる。他の利点は、当業者には明らかであろう。

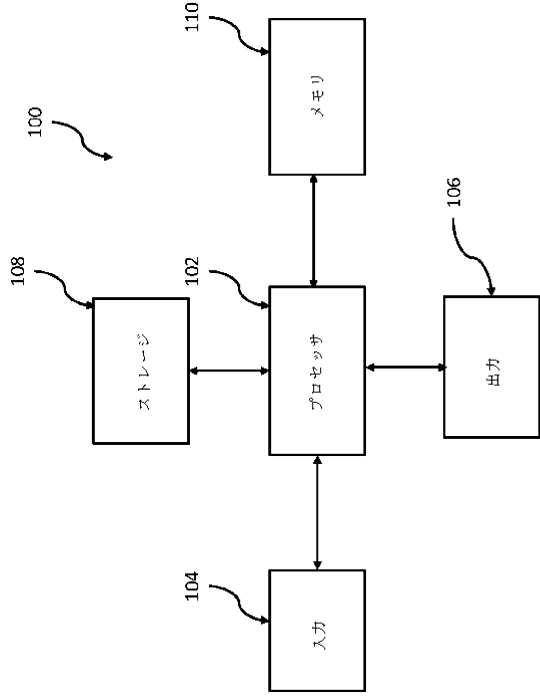
30

【0038】

上記の詳細な説明及び本明細書で説明した実施例は、例示及び説明のために提示されたものであり、本発明を限定するものではない。したがって、本開示は、上記で開示され、本明細書で特許請求された基本的な原理の趣旨及び範囲に含まれる、任意及び全ての修正物、変形物又は均等物を含むことが意図される。

40

【図 1】



【図 2】

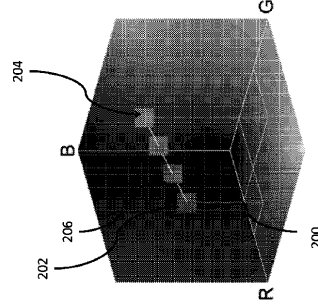
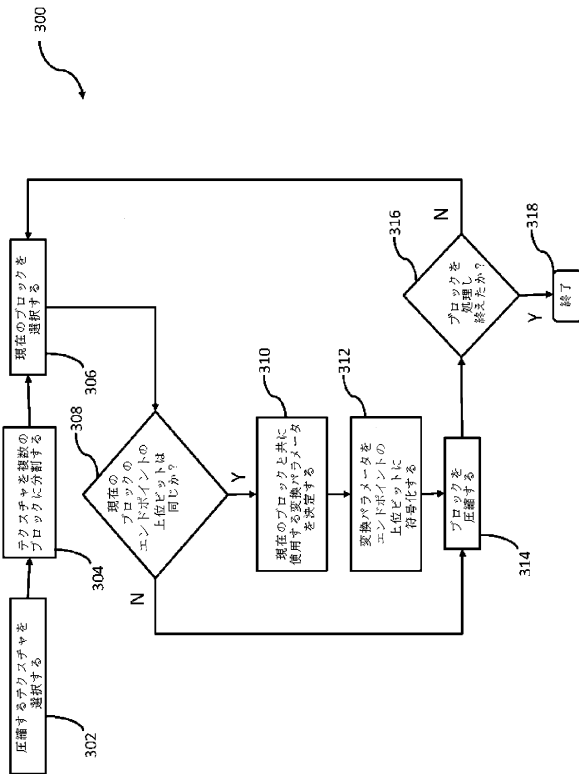
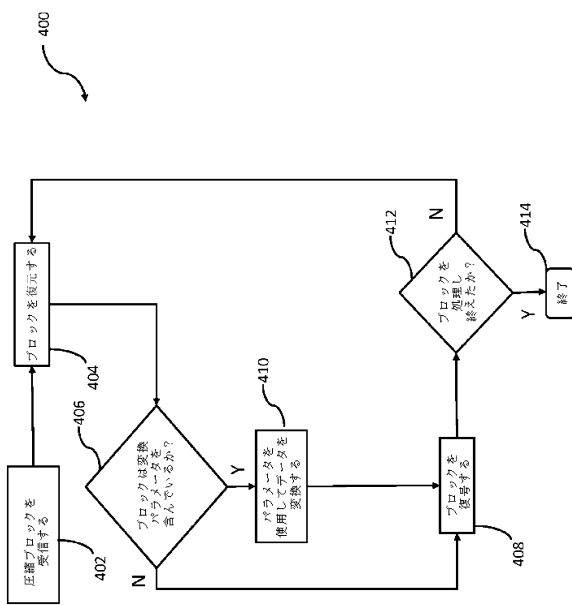


FIGURE 2

【図 3】



【図 4】



フロントページの続き

- (72)発明者 アンドルー エス．シー． ポミアノフスキー
アメリカ合衆国 94087 カリフォルニア州、サニーベール、キャリック コート 577
- (72)発明者 コンスタンチン アイオールチャ
アメリカ合衆国 95120 カリフォルニア州、サン ノゼ、ウッド レイク ドライブ 78
6

審査官 坂東 大五郎

- (56)参考文献 特開2013-034141(JP,A)
特開2001-060875(JP,A)
米国特許第07385611(US,B1)

- (58)調査した分野(Int.Cl., DB名)
- H04N 19/00 - 19/98
G06T 9/00 - 9/40
G06T 15/00 - 19/20