

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6216143号
(P6216143)

(45) 発行日 平成29年10月18日 (2017.10.18)

(24) 登録日 平成29年9月29日 (2017.9.29)

| | |
|-----------------------------|----------------|
| (51) Int. Cl. | F I |
| G06T 19/00 (2011.01) | G06T 19/00 A |
| G06T 15/20 (2011.01) | G06T 15/20 500 |
| G06T 1/00 (2006.01) | G06T 1/00 315 |

請求項の数 9 (全 19 頁)

| | | | |
|--------------|------------------------------|-----------|-------------------|
| (21) 出願番号 | 特願2013-86916 (P2013-86916) | (73) 特許権者 | 000001007 |
| (22) 出願日 | 平成25年4月17日 (2013.4.17) | | キヤノン株式会社 |
| (65) 公開番号 | 特開2014-26641 (P2014-26641A) | | 東京都大田区下丸子3丁目30番2号 |
| (43) 公開日 | 平成26年2月6日 (2014.2.6) | (74) 代理人 | 100076428 |
| 審査請求日 | 平成28年4月5日 (2016.4.5) | | 弁理士 大塚 康徳 |
| (31) 優先権主張番号 | 特願2012-139158 (P2012-139158) | (74) 代理人 | 100112508 |
| (32) 優先日 | 平成24年6月20日 (2012.6.20) | | 弁理士 高柳 司郎 |
| (33) 優先権主張国 | 日本国 (JP) | (74) 代理人 | 100115071 |
| | | | 弁理士 大塚 康弘 |
| | | (74) 代理人 | 100116894 |
| | | | 弁理士 木村 秀二 |
| | | (74) 代理人 | 100130409 |
| | | | 弁理士 下山 治 |
| | | (74) 代理人 | 100134175 |
| | | | 弁理士 永川 行光 |

最終頁に続く

(54) 【発明の名称】 画像処理装置、その制御方法、およびプログラム

(57) 【特許請求の範囲】

【請求項 1】

複数の入力画像を用いて任意視点の画像を生成する画像処理装置であって、

前記任意視点と該任意視点の画像上の画素位置を通る第一の直線に沿う点が前記複数の入力画像の各々に投影される投影位置の画像情報を取得する取得手段と、

前記第一の直線に沿って、前記任意視点からの複数の距離に複数の点を設定し、各距離において前記複数の入力画像から前記取得手段により取得された画像情報についてマッチング評価値を算出し、当該マッチング評価値に基づく、前記画素位置における前記複数の入力画像の視差情報を算出する算出手段と、

前記画素位置の画像情報を、該画素位置に対する視差情報を用いて前記複数の入力画像から算出することにより、前記任意視点の画像を生成する生成手段と、

前記任意視点の画像を分割して得られた任意視点ブロックに含まれる複数の画素位置と前記任意視点を通る複数の第一の直線に沿う前記任意視点からの距離が等しい複数の点を設定し、前記複数の点と前記入力画像の視点を通る複数の第二の直線が前記入力画像の投影面と交わる複数の投影位置の画像情報を前記取得手段を用いて得ることにより、前記任意視点ブロックに対応した前記入力画像におけるブロック画像を決定する決定手段と、を有し、

前記取得手段は、前記第一の直線に沿う点と前記入力画像の視点を通る第二の直線が、前記入力画像の投影面と交わる位置を前記投影位置に決定し、前記投影位置における画像情報を、前記入力画像の該投影位置に隣接する画素の情報をを用いて補間することにより生

10

20

成し、

前記算出手段は、前記決定手段により決定されたブロック画像の情報に基づいて前記任意視点ブロックに対する視差情報を、該任意視点ブロックの中の全ての画素位置に対する視差情報として算出することを特徴とする画像処理装置。

【請求項 2】

前記算出手段は、前記複数の点のうち最も高いマッチング評価値が得られた距離に基づいて前記視差情報を算出する、ことを特徴とする請求項 1 に記載の画像処理装置。

【請求項 3】

前記算出手段は、最も高いマッチング評価値が前記第一の直線に沿う 2 つ以上の点について算出された場合は、前記任意視点に最も近い距離に基づいて前記視差情報を算出することを特徴とする請求項 1 または 2 に記載の画像処理装置。

10

【請求項 4】

前記生成手段は、前記複数の入力画像から取得された複数の画像情報を加重平均することにより前記画素位置の画像情報を生成することを特徴とする請求項 1 乃至 3 のいずれか 1 項に記載の画像処理装置。

【請求項 5】

前記入力画像の投影面は、平面、球面、円筒面のいずれかであることを特徴とする請求項 1 乃至 4 のいずれか 1 項に記載の画像処理装置。

【請求項 6】

複数の入力画像を用いて任意視点の画像を生成する画像処理装置の制御方法であって、
取得手段が、前記任意視点と該任意視点の画像上の画素位置を通る第一の直線に沿う点
が前記複数の入力画像の各々に投影される投影位置の画像情報を取得する取得工程と、

20

算出手段が、前記第一の直線に沿って、前記任意視点からの複数の距離に複数の点を設定し、各距離において前記複数の入力画像から前記取得手段により取得された画像情報についてマッチング評価値を算出し、当該マッチング評価値に基づく、前記画素位置における前記複数の入力画像の視差情報を算出する算出工程と、

生成手段が、前記画素位置の画像情報を、該画素位置に対する視差情報を用いて前記複数の入力画像から算出することにより、前記任意視点の画像を生成する生成工程と、

決定手段が、前記任意視点の画像を分割して得られた任意視点ブロックに含まれる複数の画素位置と前記任意視点を通る複数の第一の直線に沿う前記任意視点からの距離が等しい複数の点を設定し、前記複数の点と前記入力画像の視点を通る複数の第二の直線が前記入力画像の投影面と交わる複数の投影位置の画像情報を前記取得手段を用いて得ることにより、前記任意視点ブロックに対応した前記入力画像におけるブロック画像を決定する決定工程と、を有し、

30

前記取得手段は、前記第一の直線に沿う点と前記入力画像の視点を通る第二の直線が、前記入力画像の投影面と交わる位置を前記投影位置に決定し、前記投影位置における画像情報を、前記入力画像の該投影位置に隣接する画素の情報を用いて補間することにより生成し、

前記算出手段は、前記決定手段により決定されたブロック画像の情報に基づいて前記任意視点ブロックに対する視差情報を、該任意視点ブロックの中の全ての画素位置に対する視差情報として算出することを特徴とする画像処理装置の制御方法。

40

【請求項 7】

請求項 6 に記載された制御方法の各工程をコンピュータに実行させるためのプログラム。

【請求項 8】

前記任意視点ブロックに含まれる前記複数の画素位置、および画素数が、前記任意視点ブロックの画素値によって決定されることを特徴とする請求項 1 に記載の画像処理装置。

【請求項 9】

前記任意視点ブロックの画素値は、前記任意視点ブロックの平均画素値であることを特徴とする請求項 8 に記載の画像処理装置。

50

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、複数視点より撮像された複数の画像から任意視点における画像を生成する、画像処理装置、その制御方法、およびプログラムに関する。

【背景技術】

【0002】

異なる複数視点より撮像された全周画像（視点から360°の全周囲が記録された画像）を用いて、任意視点から撮像した場合に得られる画像を生成する技術（任意視点映像生成技術）が提案されている。一般に、このような任意視点画像生成処理を行なう技術として、特許文献1や非特許文献1などが知られている。

10

【先行技術文献】

【特許文献】

【0003】

【特許文献1】特開平11-175762号公報

【非特許文献】

【0004】

【非特許文献1】視点位置に応じた3次元メッシュモデルの逐次生成による全方位動画からの自由視点画像生成（「画像の認識・理解シンポジウム（MIURU2009）」-2009年7月）

20

【発明の概要】

【発明が解決しようとする課題】

【0005】

特許文献1には、複数の撮像装置から取得した複数の全周画像を用いて、任意視点からの画像を生成する構成が記載されている。特許文献1で示される技術では、入力画像から抽出される特徴点を用いて、任意視点画像を生成する際に用いる視差情報を算出している。より具体的には、一方の入力画像視点とその入力画像上の特徴点を結ぶ視線方向をマッチングの基準として、他方の入力画像から抽出される特徴点をマッチングさせていた。ここで、マッチングにより得られる特徴点間の位置ずれ量に関する情報が視差情報として検出される。このように、視差情報は、一方の入力画像の視点とその入力画像上の特徴点とを結ぶ視線方向を基準として検出されるため、この視線方向以外の画像領域において視差情報を検出することは困難であった。

30

【0006】

また、非特許文献1に示される技術では、視差情報により任意視点映像を生成する際に、視差が検出されない画像領域においては、隣接する特徴点位置における視差情報を用いて、画像情報を補償していた。用いた視差情報は幾何的に正確ではなく、このため、任意視点映像の画質が低かった。

【0007】

本発明は、上記課題に鑑みてなされたものであり、複数の入力画像から生成される任意視点画像の画質を向上させることを目的とする。

40

【課題を解決するための手段】

【0008】

上記の目的を達成するための、本発明の一態様による画像処理装置は以下の構成を有する。すなわち、

複数の入力画像を用いて任意視点の画像を生成する画像処理装置であって、

前記任意視点と該任意視点の画像上の画素位置を通る第一の直線に沿う点が前記複数の入力画像の各々に投影される投影位置の画像情報を取得する取得手段と、

前記第一の直線に沿って、前記任意視点からの複数の距離に複数の点を設定し、各距離において前記複数の入力画像から前記取得手段により取得された画像情報についてマッチング評価値を算出し、当該マッチング評価値に基づく、前記画素位置における前記複数の

50

入力画像の視差情報を算出する算出手段と、

前記画素位置の画像情報を、該画素位置に対する視差情報を用いて前記複数の入力画像から算出することにより、前記任意視点の画像を生成する生成手段と、

前記任意視点の画像を分割して得られた任意視点ブロックに含まれる複数の画素位置と前記任意視点を通る複数の第一の直線に沿う前記任意視点からの距離が等しい複数の点を設定し、前記複数の点と前記入力画像の視点を通る複数の第二の直線が前記入力画像の投影面と交わる複数の投影位置の画像情報を前記取得手段を用いて得ることにより、前記任意視点ブロックに対応した前記入力画像におけるブロック画像を決定する決定手段と、を有し、

前記取得手段は、前記第一の直線に沿う点と前記入力画像の視点を通る第二の直線が、前記入力画像の投影面と交わる位置を前記投影位置に決定し、前記投影位置における画像情報を、前記入力画像の該投影位置に隣接する画素の情報を用いて補間することにより生成し、

前記算出手段は、前記決定手段により決定されたブロック画像の情報に基づいて前記任意視点ブロックに対する視差情報を、該任意視点ブロックの中の全ての画素位置に対する視差情報として算出する。

【発明の効果】

【0009】

本発明によれば、複数の入力画像から生成される任意視点画像の画質を向上させることができる。

【図面の簡単な説明】

【0010】

【図1】実施形態による任意視点画像の生成を説明する模式図。

【図2】実施形態のコンピュータのハードウェア構成例を示す図。

【図3】実施形態の入力装置を示す模式図。

【図4】実施形態の入力画像の投影形状を示す模式図。

【図5】実施形態による全体の処理を示すフローチャート。

【図6】実施形態による視差マップ生成処理を示すフローチャート。

【図7】実施形態による入力画像を用いたブロック分割処理を説明する図。

【図8】実施形態による処理を示すフローチャート。

【図9】実施形態による処理を示すフローチャート。

【図10】実施形態による画素補間生成処理を示す模式図。

【図11】実施形態による処理を示すフローチャート。

【図12】実施形態2による視差マップ生成処理を示すフローチャート。

【図13】実施形態2によるコンピュータグラフィックス用光源画像生成処理の概要を示す図。

【図14】実施形態2による視差マップ生成処理内のブロック画像解像度決定の概要を示す図。

【発明を実施するための形態】

【0011】

以下、添付の図面を参照して本発明の好適な実施形態について説明する。

【0012】

[実施形態1]

本実施形態の概要として、図を用いて本発明の従来技術に対する差異・特徴を説明する。図1は、特許文献1に記載された技術と本実施形態との違いを説明するための模式図である。一般的に、視差情報（視差ベクトル）を算出するためには、複数の画像を用いて、画像データの対応位置を特定する必要がある。特許文献1に記載の技術および本実施形態では、3次元空間における3次元モデルに基づいて、その視差情報を算出する。

【0013】

なお、本実施形態においては、2次元空間における2次元モデルを併用して説明を行な

10

20

30

40

50

う。以下の説明において、3次元空間の所定の位置に視点が配置される全周画像は、XZ平面上に配置されるものとする。また、説明中で用いられる光線関数は以下の直線関数である。

$$R(t) = O + tD \quad \text{数1}$$

数1において、 t はスカラーで表される距離パラメータであり、 $R(t)$ 、 O 、 D はベクトルである。3次元空間における3次元モデルでは、ベクトルは3次元ベクトルであり、座標値や方向を示すために用いられる。2次元空間における2次元モデルではベクトルは2次元ベクトルであり、同様に座標値や方向を示すために用いられる。

【0014】

図1(A)は、特許文献1に記載されている、視差情報の算出方法を示す模式図である。図1(A)において、101、102は入力画像を表す。特許文献1においては、これらの入力画像のうち一つを選択する。ここでは、入力画像101を選択し、さらに、入力画像101の投影面に対応した所定の円周上の一点 $p1_i1$ を選択する。なお、円周上の点には、画像情報が記録されている。入力画像101は、円中心 O_1 が定義されており、当該円中心 O_1 の座標値 $O1$ と所定の一点 $p1_i1$ より、以下の数2で表される直線 $R1$ を描くことができる。数2において、 $D1$ は円中心 O_1 と点 $p1_i1$ より算出される方向ベクトルであり、大きさが1の単位方向ベクトルである。 $t1$ は距離パラメータである。

$$R1(t1) = O1 + t1 \times D1 \quad \text{数2}$$

数2において、 $t1$ に所定の値を入力することで、数2で表される直線上の点を特定できる。例えば、点 $p1_i1$ が図1(A)のように特定されるときは $t1 = 1$ である。また、点 $p1_i2$ 、 $p1_i3$ 、 $p1_i4$ が図1(A)のように特定されるときは、それぞれ $t1 = a$ 、 $t1 = b$ 、 $t1 = c$ ($1 < a < b < c$) である。

【0015】

次に、平面上の特徴点 $p1_i2$ の、入力画像102における投影位置 $p2_i2$ を算出する。まず、投影位置 $p2_i2$ と円中心 O_2 を通る直線 $R2$ は数2のように表される。ここで $O2$ は円中心 O_2 の座標値あり、 $D2$ は円中心 O_2 と点 $p1_i2$ より算出される単位方向ベクトルである。 $t2$ は距離パラメータである。

$$R2(t2) = O2 + t2 \times D2 \quad \text{数3}$$

数3を数4のように式変形することで、投影位置 $p2_i2$ を得る。数4において、 $stand$ は入力されたベクトルの単位ベクトルを返す関数($R' = stand(R)$)である。

$$D2 = stand(R2(t2) - O2) \quad \text{数4}$$

ここで単位ベクトルと $stand$ について補足する。ベクトル R に対してその単位ベクトル R' は、 $R' = R / |R| = R / r$ と表される。ここで、 $r = |R|$ はベクトル R の長さである。

【0016】

xyz の3次元空間を扱うときは、 $R = xI + yJ + zK$ と表すことができる。 I 、 J 、 K はそれぞれ、 x 、 y 、 z の各軸方向の単位ベクトルである。ベクトルの長さは $|R| = \sqrt{x^2 + y^2 + z^2}$ と表される。 $\sqrt{}$ は入力された値の平方根を返す関数である。したがって、単位ベクトル R' は以下の数5で表される。

$$R' = stand(R) = x / |R| I + y / |R| J + z / |R| K \quad \text{数5}$$

xz の2次元平面を扱うときは、 $R = xI + zK$ と表すことができる。ベクトルの長さは $|R| = \sqrt{x^2 + z^2}$ と表される。したがって、単位ベクトル R' は以下の数6で表される。

$$R' = stand(R) = x / |R| I + z / |R| K \quad \text{数6}$$

【0017】

次に、点 $p1_i1$ に記録されている画像情報と投影位置 $p2_i2$ に記録されている画像情報とのマッチング評価値を算出する。このマッチング評価値を $i(1-2)$ とする。なお、マッチング評価値についての詳細は後述する。同様に、特徴点 $p1_i3$ につい

10

20

30

40

50

ても、入力画像 102 における投影位置 $p2_i3$ を算出する。そして、点 $p1_i1$ に記録されている画像情報と投影位置 $p2_i3$ に記録されている画像情報とのマッチング評価値 $i(1-3)$ を算出する。特徴点 $p1_i4$ についても同様にして、マッチング評価値 $i(1-4)$ を算出する。

【0018】

上述のようにして算出されたマッチング評価値 $i(1-2)$ 、 $i(1-3)$ 、 $i(1-4)$ を比較することにより、視差が算出される。この特許文献 1 に記載の視差とは、具体的には、マッチング評価値が一番高かった平面上の特徴点 ($p1_i2$ 、 $p1_i3$ 、 $p1_i4$ のいずれか) までの円中心 O_1 からの距離である。

【0019】

視差が $i(1-2)$ と算出された場合、任意視点画像 103 からは、任意視点画像 103 の円中心 O_3 と特徴点 $p1_i2$ を結ぶ直線上の投影位置 $p3_i2$ に画像情報が記録されるのが幾何的に妥当である。ここで、投影位置 $p3_i2$ は以下のように算出される。まず、投影位置 $p3_i2$ と円中心 O_3 を通る直線 $R3$ は数 7 のように表され、ここで、 $O3$ は円中心 O_3 の座標値あり、 $D3$ は円中心 O_3 と特徴点 $p1_i2$ より算出される単位方向ベクトルである。 $t3$ は距離パラメータである。

$$R3(t3) = O3 + t3 \times D3 \quad \text{数 7}$$

数 7 を数 8 のように式変形することで、投影位置 $p3_i2$ を得る。

$$D3 = \text{stand}(R3(t3) - O3) \quad \text{数 8}$$

同様に、投影位置 $p3_i3$ と投影位置 $p3_i4$ も算出できる。

【0020】

このように、特許文献 1 に記載の技術では、入力画像 101 の円中心 O_1 を通る数 2 で表される直線 $R1$ に視差が算出される。このため、任意視点画像 103 における投影位置は、入力画像 101 と入力画像 102 の視差に依存してしまい、一意に特定されない。

【0021】

一方、図 1(B) は、本実施形態による視差情報の算出方法を示す模式図である。本実施形態においては、まず任意視点画像 106 が選択される。選択された任意視点画像 106 において、投影面に対応した所定の円周上の一点 $p6_i1$ (画素位置) が選択される。任意視点画像 106 は、円中心 O_6 (任意視点) が定義されており、円中心 O_6 の座標値 $O6$ と円周上の所定の一点 $p6_i1$ により、以下の数 9 で表される直線を描くことができる。数 9 において、 $D6$ は円中心 O_6 と点 $p6_i1$ より算出される単位方向ベクトルである。また、 $t6$ は距離パラメータであり、 $t6$ に所定の値を入力することで、数 9 で表される直線上の点を特定できる。

$$R6(t6) = O6 + t6 \times D6 \quad \text{数 9}$$

例えば、点 $p6_i1$ が図 1(B) のように特定されているときは $t6 = 1$ である。また、点 $p6_i2$ 、 $p6_i3$ 、 $p6_i4$ が図 1(B) のように特定されているときは、それぞれ $t6 = d$ 、 $t6 = e$ 、 $t6 = f$ ($1 < d < e < f$) である。

【0022】

次に、平面上の一点 $p6_i2$ について、入力画像 107 における投影位置 $p7_i2$ を以下の数 10、数 11 を用いて算出する。ここで $O7$ は、入力画像 107 の投影面に対応した円の円中心 O_7 の座標値あり、 $D7$ は円中心 O_7 と点 $p6_i2$ より算出される単位方向ベクトルである。 $D7$ は入力画像 107 における円中心 O_7 から距離 1 の位置に形成される画像平面上の投影位置 $p7_i2$ に一致する。 $t7$ は距離パラメータである。

$$R7(t7) = O7 + t7 \times D7 \quad \text{数 10}$$

$$D7 = \text{stand}(R7(t7) - O7) \quad \text{数 11}$$

【0023】

同様に、入力画像 108 において、平面上の一点 $p6_i2$ に対する投影位置 $p8_i2$ を以下の数 12、数 13 を用いて算出する。 $O8$ は入力画像 107 の投影面に対応した円の円中心 O_8 の座標値あり、 $D8$ は円中心 O_8 と点 $p6_i2$ より算出される単位

10

20

30

40

50

方向ベクトルである。D 8 は入力画像 1 0 8 における円中心 O _ 8 から距離 1 の位置に形成される画像平面上の投影位置 p 8 _ i 2 に一致する。t 8 は距離パラメータである。

$$R 8 (t 8) = O 8 + t 8 \times D 8 \quad \text{数 1 2}$$

$$D 8 = s t a n d (R 8 (t 8) - O 8) \quad \text{数 1 3}$$

【 0 0 2 4 】

次に、入力画像 1 0 7 の投影位置 p 7 _ i 2 に記録されている画像情報と、入力画像 1 0 8 の投影位置 p 8 _ i 2 に記録されている画像情報とのマッチング評価値を算出する。そのマッチング評価値を i (6 - 2) とする。同様に、点 p 6 _ i 3 についても、入力画像 1 0 7 における投影位置 p 7 _ i 3 と、入力画像 1 0 8 における投影位置 p 8 _ i 3 を算出し、この 2 つの投影位置に記録されている画像情報のマッチング評価値を算出する。そのマッチング評価値を i (6 - 3) とする。点 p 6 _ i 4 についても同様にして、マッチング評価値 i (6 - 4) を算出する。

10

【 0 0 2 5 】

上述のように算出された評価値 i (6 - 2)、i (6 - 3)、i (6 - 4) を比較することで、点 p 6 _ i 1 (画素位置) における視差が算出される。本実施形態による視差とは、具体的には、マッチング評価値が一番高かった平面上の一点までの円中心 O _ 6 からの距離である。評価値 i (6 - 2)、i (6 - 3)、i (6 - 4) のいずれが最大の評価値となっても、任意視点画像 1 0 6 の円中心 O _ 6 と点 p 6 _ i 2 ~ 4 を結ぶ直線上の点 p 6 _ i 1 の位置に画像情報が記録されるのが幾何的に妥当である。このように、本実施形態では任意視点画像 1 0 6 の円中心 O _ 6 を通る数 9 で表される直線 R 6 に視差が算出される。このため、任意視点画像 1 0 6 における投影位置は、視差に依存せず一意に特定される。なお、入力画像 1 0 7、1 0 8、および、これらから生成される任意視点画像 1 0 6 は、静止画像であっても動画画像であってもよい。

20

【 0 0 2 6 】

(画像処理装置)

図 2 は、本実施形態によるコンピュータのハードウェア構成の一例を示す図である。C P U (C e n t r a l P r o c e s s i n g U n i t) 2 0 1 は、外部記憶装置 (ハードディスク) 2 0 7 に格納されている O S (O p e r a t i n g S y s t e m)、アプリケーションプログラム等を実行して画像処理装置 2 0 0 の動作を制御する。なお、C P U 2 0 1 は、対応する処理プログラムを実行する。また、R A M (R a n d o m A c c e s s M e m o r y) 2 0 3 にプログラムの実行に必要な情報、ファイル等を一時的に格納する制御を行う。R O M (R e a d O n l y M e m o r y) 2 0 2 は、基本 I / O プログラム等のプログラムが記憶されている。R A M 2 0 3 は、C P U 2 0 1 の主メモリ、ワークエリア等として機能する。ネットワーク I / F 2 0 4 は、P A N (P e r s o n a l A r e a N e t w o r k)、L A N (L o c a l A r e a N e t w o r k)、W A N (W i d e A r e a N e t w o r k) に接続して外部装置と通信するためのインタフェースである。入力装置 2 0 5 は、ユーザーからの入力を受け付けるマウス、キーボードの他、被写体の画像を撮影して画像入力を行うための撮像装置を含むことができる。当該撮像装置を有することにより、画像処理装置 2 0 0 はデジタルカメラ、或いは、デジタルビデオカメラとして機能することができる。

30

40

【 0 0 2 7 】

入力装置 2 0 5 における撮像装置は、図 3 の模式図で示される。撮像装置は、カメラボディ部 3 0 1、第一の撮像レンズ部 3 0 2、第二の撮像レンズ部 3 0 3 により構成される。第一の撮像レンズ部 3 0 2 および第二の撮像レンズ部 3 0 3 は、画角 1 8 0 ° 以上の超広角レンズ相当の光学系を備えている。本発明において、レンズの数は構成要件ではなく、視点の数が構成要件となる。視点の数は 2 つ以上の複数である。本実施形態では視点は 2 つとする。また、入力装置 2 0 5 における撮像装置は、図 4 に示されるような、4 0 1 で示される全周画像を、複数入力するものとする。全周画像 4 0 1 は、球面に投影された画像であり、球面は視点位置 4 0 2 を中心として、所定の半径 4 0 3 を持つ球の表面である。

50

【0028】

図2に戻り、出力装置206は、液晶ディスプレイのような表示装置である。外部記憶装置207は、アプリケーションプログラム、ドライバプログラム、OS(Operating System)、制御プログラム、本実施形態に対応する処理を実行するための処理プログラム等を格納している。システムバス208は、装置内のデータの流れを司る。

【0029】

(画像処理方法)

図5は、CPU201が実行する本実施形態の画像処理プログラム(以下、本処理(メインプロセス)プログラムという)を説明するフローチャートである。このフローチャートで表される本処理プログラムは、通常、外部記憶装置207に記録されており、処理を開始するにあたりCPU201が外部記憶装置207から読み込んで、RAM203に展開し、実行する。

【0030】

S501は、処理条件入力ステップである。CPU201は、本処理プログラムが必要とする情報を入力する。入力情報は、本処理プログラムが外部記憶装置207から読み込むことや、ネットワークI/F204を介することによって取得することができる。また、入力装置205を介して、ユーザーによる入力を読み込むようにしても良い。

【0031】

本実施形態では、入力情報として、図4に示されるような、視点402に対して全周囲の画像を用いるものとし、異なる2視点以上の全周囲の画像が入力される。また、各視点の視点位置、視線方向も入力する。視点位置や視線方向が固定されている場合や、事前に視点位置や視線方向の変動が取得できる場合は、外部記憶装置207から読み込むようにする。

【0032】

ここで、入力する画像の投影面について説明する。一般的には撮像時において、シーンは平面に投影されて画像に記録されることが多い。このことを鑑みて、本実施形態では、平面の投影面を持つ画像を入力可能とする。また、更に平面の投影面に投影変換できる、平面以外の別の投影面を持つ画像を入力可能とする。そのような投影面として円筒面・球面・複数の平面が挙げられる。本実施形態では、円筒面の投影面を持つ画像を入力する。このように、多くの投影面形状を入力可能とすることで、本実施形態の画像処理装置は、より広範な画像を扱うことが可能な、汎用的な構成とすることができる。なお、各視点の画像を撮影するときの撮像設定(ISO感度、露光時間、絞り値、ピント)は、同じとする。このように撮像設定を同じにすることで、後に説明する画像同士を比較する処理の精度を上げることができる。ただし、画像間で撮像設定は同じでなくても、本実施形態では良いものとする。

【0033】

また、入力情報として、本実施形態の概要で述べた、数1において距離パラメータ t_1 の取る値を設定する。すなわち、距離パラメータのステップ t_s 、距離パラメータの上限値 t_{end} 、距離パラメータの下限値 t_{start} を入力する。これらを用いた処理については後述する。

【0034】

S501ではさらに、任意視点画像の視点位置を入力し、任意視点画像の投影面を設定する。ここで、任意視点画像の投影面について説明する。本実施形態の画像処理装置は、最終的な出力である任意視点画像において、平面の投影面を持つ画像を出力可能とする。また、更に平面の投影面に投影変換できる、平面以外の別の投影面を持つ画像を出力可能とする。そのような投影面として円筒面・球面・複数の平面が挙げられる。以下では、円筒面の投影面を持つ画像を出力する場合を説明する。

【0035】

S502は、視差マップ生成ステップである。CPU201は、S501で設定された

10

20

30

40

50

任意視点画像を所定のブロックに分割し、入力された全周囲の画像を用いて、そのブロック単位の視差情報（深度情報）を算出する。そして、算出した視差情報を、任意視点画像上における分割したブロックにマッピングすることで、視差マップを生成する。

【 0 0 3 6 】

視差マップ生成ステップ S 5 0 2 について、図 6 を用いて説明する。なお、以下の説明では、図 6 に示されるフローチャートを本サブフローチャート - 1 と称する。

【 0 0 3 7 】

S 6 0 1 は、ブロック生成ステップであり、C P U 2 0 1 は、任意視点画像における所定の画像領域を指定する。本実施形態では、円筒座標系に投影された任意視点画像を生成する。円筒座標系は球座標系に投影変換できる。また、球座標系は円筒座標系に投影変換できる。このことを利用して、円筒座標系における任意視点画像の視差マップを生成する。

【 0 0 3 8 】

S 6 0 2 は、探索基準ブロック更新ステップである。S 6 0 2 の処理について、図 7 を用いて説明する。図 7 に示される模式図は、円筒座標系の平面展開図である。円筒座標系の投影面は平面に展開可能であり、画像領域を分割することでブロックが生成される。本実施形態では等間隔の正方形にブロック分割される。x 方向を $M + 1$ 、y 方向を $N + 1$ に分割することで、総ブロック数が $(M + 1) \times (N + 1)$ になるように円筒座標系の平面展開図をブロック分割する。生成したブロック群において、処理対象のブロックを指示するポイントが更新される。図 7 (A) では m_p 行 n_q 列のブロックが選択されている。図 7 (A) のように選択された後、図 7 (B) のように、 $m_p + 1$ 行 $n_q + 1$ 列のブロックを選択するように、図 7 (A) で選択されていた m_p 行 n_q 列のブロックに対して異なるブロックが選択される。なお、ブロックの選択順は特に規定されるものではないが、以前選択したブロックが再び選択されることはない。

【 0 0 3 9 】

S 6 0 3 は、光源推定距離パラメータ更新ステップである。数 1 4 で表されるように、第一の光線関数（第一の直線）の距離パラメータ t を、距離パラメータのステップ t_s だけ更新する。距離パラメータ t が取り得る値の範囲は、距離パラメータの上限値 t_{end} 、距離パラメータの下限値 t_{start} によって決まる。すなわち、 $t_{start} \leq t \leq t_{end}$ の関係を有する。ここで t_{start} は 0 以上の値を取り、 t_{end} は無限大より小さい値を取る。

$$t = t + t_s \quad \text{数 1 4}$$

距離パラメータ t の初期値は、距離パラメータ t が取り得る値の範囲内で所望の値を取ることができる。また、 t_s は定数と変数のいずれかを取ることができる。 t_s が変数を取る場合、 $t_s = f(x) = ax + b$ のような関数の形をとっても良いし、 $t_s 0 = 0$ 、 $t_s 1 = 1$ 、 $t_s 2 = 2$ 、 $t_s 3 = 3$ 、 $t_s 4 = 5$ 、 $t_s 5 = 7 \cdots$ のように配列に格納された値を順次参照するようにしてもよい。

【 0 0 4 0 】

S 6 0 4 は、ブロック画像生成ステップである。本ステップでは、以下の 3 つの処理が行なわれる。

(1) 第一の光線関数（第一の直線）を用いた平面上の所定の一点 P の算出

(2) 平面上の所定の一点 P と入力画像の視点による第二の光線関数（第二の直線）の算出

(3) 第二の光線関数と入力画像の投影面との交わる点における画素の補間生成

上記 S 6 0 4 における処理 (1) ~ (3) について、図 8 を用いて説明する。

【 0 0 4 1 】

図 8 は、図 6 における S 6 0 4 の処理を詳細化させた処理フローチャートである。なお、以下の説明では、図 8 に示される処理フローチャートを本サブフローチャート - 2 と称する。また、本サブフローチャート - 2 の処理の説明を補足するため、図 9 の模式図を用いる。なお、以下では、入力画像 1 0 7 から得られるブロック画像 B K 1 の生成方法につ

10

20

30

40

50

いて述べる。同様の方法で入力画像 108 から得られるブロック画像 BK2 については説明を省略する。

【0042】

図9(A)は、任意視点画像106の円筒座標系の平面展開図(任意視点ブロック画像)であり、総ブロック数 $(M+1) \times (N+1)$ のうち、 m_p 行 n_q 列のブロックが選択されている状態であり、その様子が網掛け部分によって図示されている。それぞれのブロックは複数の画素によって構成される。例えば、本実施形態では、1ブロックは 3×3 の9画素によって構成されている。

【0043】

図9(B)は、3次元空間上に仮想的に配置された光源から射出される光が、二つの異なる視点へ到達する際の、光線のベクトルを模式的に示した図である。なお、図9(B)において図1(B)と同一の内容については、図1(B)と同じ参照番号が付されている。また、図9(C)は、入力画像107の円筒座標系の平面展開図である。図9(C)では、第二の光線関数と入力画像の投影面との交わる点の集合により構成されるブロック画像BK1が示されている。

【0044】

S801は、ブロック画素座標値更新ステップである。図9(A)におけるブロックは 3×3 の9画素で構成されており、CPU201はこれらの画素の一つずつについて処理を行なうため、 3×3 の9画素のうち一つを選択する。

【0045】

S802は、第一光線ベクトル算出ステップである。図9(A)で示される m_p 行 n_q 列目のブロック内において、隣り合う画素の座標値(画素位置)である $C6_1$ と $C6_1_1$ は、図9(B)で示される3次元空間上では、それぞれ異なる単位方向ベクトル $p6_i1$ と $p6_i1_1$ となる。円中心 O_6 と画素の座標値から単位方向ベクトルは一意に決まる。これらの単位方向ベクトルを用いて、ブロック内の画素ごとにそれぞれ第一の光線関数を算出する。このように、光線関数に用いる単位方向ベクトルと画像中の画素の座標値を画像の中心位置を用いて相互に変換することが本処理の特徴である。

【0046】

単位方向ベクトル $p6_i1_1$ を選択した場合の第一の光線関数の算出について、図9(B)を用いて説明する。数9において、距離パラメータを $t6 = f$ とすると、円中心 O_6 と単位方向ベクトル $p6_i1_1$ によって以下の数15で示される3次元空間上の位置が算出される。

$$p6_i4_1 = O6 + t6 \times p6_i1_1 \quad \text{数15}$$

後述のS805によりS801とS802が繰り返されることにより、 m_p 行 n_q 列目のブロック内の9画素のすべてについて、単位方向ベクトルと円中心 O_6 を用いてそれぞれ第一の光線関数が算出される。

【0047】

図9(B)に示されるように、位置 $p6_i4$ と $p6_i4_1$ により、円中心 O_6 を中心とする距離 f の円弧が形成されるのがわかる。なお、実際には、3次元空間上で処理を行なうので、球面の一部が形成されることになる。

【0048】

S803は、補間画素投影座標値算出ステップである。位置 $p6_i4_1$ と入力画像107の円中心 O_7 から算出される第二の光線関数より、単位方向ベクトル $p7_i4_1$ は以下のように求められる。

$$p7_i4_1 = \text{stand}(p6_i4_1 - O7) \quad \text{数16}$$

CPU201は、得られた単位方向ベクトル $p7_i4_1$ と円中心 O_7 から図9(C)のように、画素の座標値 $C7_1_1$ を特定する。図9(C)は入力画像107を平面に展開した模式図である。ブロック画像BK1はこのようにして特定された画素の座標値から構成される。

【0049】

10

20

30

40

50

S 8 0 4 は、補間画素生成ステップである。画素座標値 C 7 _ 1 _ 1 は、小数精度の座標値となるため、C P U 2 0 1 は、入力画像 1 0 7 を構成する整数精度の座標値を持つ画素を用いて、画素値を補間生成する。画素の補間生成方法としては、最近傍補間法、バイリニア補間法、バイキュービック補間法など、周知の補間技術が適用可能である。最近傍補間法を用いて画素座標値 C 7 _ 1 _ 1 における画素値を補間生成する場合、入力画像 1 0 7 における画素のうち、その配置座標が画素座標値 C 7 _ 1 _ 1 に最も近い画素を、画素座標値 C 7 _ 1 _ 1 における画素値として出力する。

【 0 0 5 0 】

図 1 0 では、バイキュービック補間法における入力画素の選定方法を図示している。図 1 0 において、D で表される画素座標値 C 7 _ 1 _ 1 が整数値をとるように、小数値を切り捨てることで、(x i n t , y i n t) で示される入力画素を特定する。特定された入力画素を基準にして、C で示される画素座標値 C 7 _ 1 _ 1 の周囲の近傍 1 6 点を選定することで、D をバイキュービック補間法するための入力画素を得る。複数の入力画素を得た後は、バイキュービック補間関数を適用して補間画素値を算出する。バイリニア補間法についても画素座標値 C 7 _ 1 _ 1 に最近傍に隣接する入力画素の座標値から、複数の画素値を得ることで、バイリニア補間関数を適用して補間画素値を算出する。

【 0 0 5 1 】

S 8 0 5 は、ブロック画像生成終了判定ステップである。S 8 0 1 でブロック内の全ての画素が選択されたら S 6 0 5 へ処理を進め、全て選択されていなければ S 8 0 1 へ処理を進める。

【 0 0 5 2 】

以上が図 8 に示される本サブフローチャート - 2 (図 6 の S 6 0 4) の処理である。本サブフローチャート - 2 の終了によりブロック画像 B K 1 が生成される。また、同様に入力画像 1 0 8 を用いてブロック画像 B K 2 が生成される。

【 0 0 5 3 】

図 6 に戻り、S 6 0 5 は、マッチング評価値算出ステップである。C P U 2 0 1 は、入力画像 1 0 7 のブロック画像 B K 1 と入力画像 1 0 8 のブロック画像 B K 2 のマッチングを行なう。本実施形態では、マッチング評価値として、数 1 7 で示される S A D 値 (差分絶対値和) を算出する。なお、数 1 7 において、本実施形態では $M = N = 3$ である (ブロックサイズに一致する)。 $g (x)$ はブロック画像 B K 1 の画像平面を表し、 $h (x)$ はブロック画像 B K 2 の画像平面を表す。

$$SAD = \sum_{m=0}^M \sum_{n=0}^N |g(x_0+m, y_0+n) - h(x_1+m, y_1+n)| \quad \text{数 1 7}$$

また、上記の他にも相関係数や、最小二乗マッチング法を適用してもよい。

【 0 0 5 4 】

C P U 2 0 1 は、マッチング演算により算出されるマッチング評価値 (S A D 値) を出力する。そして、後述する S 6 0 6 でマッチング終了と判定されるまで、S 6 0 3 で距離パラメータ t 6 が更新され、S 6 0 5 で新たなマッチング評価値の算出と比較が行われる。

【 0 0 5 5 】

マッチング演算に S A D 値を用いている場合、C P U 2 0 1 は、以前のマッチング演算結果の中で最も高くマッチングした (最も低い値を示した) マッチング評価値 SAD_{pre} と、今回のマッチング評価値 SAD_{cur} を用いて、以下の比較結果 Result を算出する。

$$Result = \begin{cases} 1 & (SAD_{cur} \leq SAD_{pre}) \\ 0 & (SAD_{cur} > SAD_{pre}) \end{cases} \quad \text{数 1 8}$$

【 0 0 5 6 】

Result = 0 の場合、以前に演算したマッチング評価値の中で最もマッチングした

10

20

30

40

50

評価値の方が信頼できる値であると判断して、今回のマッチング評価値は破棄する。Result = 1 の場合、以前に演算した全てのマッチング評価値よりも、今回のマッチング評価値の方がマッチングしている。すなわち、今回のマッチング演算に用いたブロック画像を算出するために用いた視差情報が、より正しい視差情報であると判断する。視差情報については、後述する。そして、距離パラメータ t_6 が更新された新たなマッチング評価値との比較で用いるため、数 19 に示されるように、今回のマッチング演算結果を、最もマッチングしたマッチング評価値を SAD_{pre} に設定する。

$$SAD_{pre} = SAD_{cur} \quad \text{数 19}$$

【0057】

10

また、Result = 1 の場合は、以前に演算したマッチング評価値の中で最もマッチングした評価値と今回のマッチング評価値が同じである場合も含まれる。この場合、任意視点に最も近い点についてのマッチング評価値を SAD_{pre} に設定する。

【0058】

比較結果を得ると同時に、今回のマッチング演算に用いたブロック画像 (BK1 と BK2) を算出するために用いた、視差情報を保持する。保持する視差情報には、距離パラメータ t_6 (本実施形態では $t_6 = f$) が含まれる。また、ブロック画像 BK1 とブロック画像 BK2 を算出するに用いた単位方向ベクトル p_{6_i1} や $p_{6_i1_1}$ も含まれる。また、単位方向ベクトル p_{6_i1} と、隣り合う画素の座標値から変換される単位方向ベクトルの差分が含まれるようにしても良い。隣り合う画素からそれぞれ変換される2つの単位方向ベクトルの差分は、単位方向ベクトル p_{6_i1} と $p_{6_i1_1}$ の差を算出することで得られる。または、ブロック画像 BK1 とブロック画像 BK2 を直接保持するようにしてもよい。

20

【0059】

S606 は、マッチング終了判定ステップである。距離パラメータ t_6 には、ステップ t_s 、下限値 t_{start} 、上限値 t_{end} により、取り得る値が定められている。上限値または下限値を超える値が距離パラメータ t_6 に設定される場合、マッチングを終了して、S607 へ処理を進める。他方、距離パラメータ t_6 に設定される値が取り得る値の範囲内である場合は S603 へ処理を進める。このようにして、マッチング終了と判定されたときに保持されている視差情報は、S605 によるマッチング評価の最も高い視差情報であり、この視差情報が S602 で選択された探索基準ブロックに対応する視差情報に決定される。

30

【0060】

S607 は、視差情報マッピングステップである。CPU201 は、S605 で保持した視差情報をメモリに保持する。本実施形態では、図9(A)に示すように、総ブロック数 $(M+1) \times (N+1)$ であり、 m_p 行 n_q 列のブロックについては、 m_p 行 n_q 列のメモリ位置にマッピングされる。こうして、S602 で選択された探索基準ブロックに視差情報が対応付けられる。

【0061】

S608 は、視差マップ生成終了判定ステップである。S602 で総ブロック数 $(M+1) \times (N+1)$ のブロックが全て選択されたら、本サブフローチャート - 1 の処理は終了であり、そうでなければ S602 へ処理を進める。以上が、図6に示される本サブフローチャート - 1 (図5の S502) の処理である。

40

【0062】

図5に戻り、S503 は、任意視点画像生成ステップである。S502 で算出された視差マップを用いて任意視点画像 106 を補間生成する。図11を用いて本実施形態における任意視点画像の生成方法を説明する。図11は、図5における S503 の処理を詳細化させた処理フローチャート図である。なお、以下の説明では、図11に示されるフローチャートを本サブフローチャート - 3 と称する。

【0063】

50

S 1 1 0 1 は、探索基準ブロック更新ステップである。処理内容は、図 6 における S 6 0 2 と同じである。S 1 1 0 2 は、視差情報取得ステップである。S 5 0 2 で算出された視差マップから、当該ブロックに相当する視差情報を取得する。本実施形態では、総ブロック数 $(M + 1) \times (N + 1)$ である。以下、ブロックを補間生成する処理について、 m_p 行 n_q 列のメモリ位置にマッピングされている m_p 行 n_q のブロックの場合を例として説明する。

【 0 0 6 4 】

S 1 1 0 3 は、ブロック画像生成である。CPU 2 0 1 は、取得された視差情報（距離パラメータ t_6 、単位方向ベクトル $p_6_i_1$ 、単位方向ベクトル $p_6_i_1_1$ ）から、 m_p 行 n_q 列のブロック画像（ブロック画像 B K 1、ブロック画像 B K 2）を生成する。詳細な処理内容は、既に述べた図 6 における S 6 0 4 の処理内容と同じである。なお、前述したように、S 6 0 5 において評価の最も高い視差情報としてブロック画像 B K 1、B K 2 を保持しておいてもよい。

【 0 0 6 5 】

S 1 1 0 4 は、ブロック画像マッピングステップである。CPU 2 0 1 は、 m_p 行 n_q 列のブロックの座標位置にブロック画像を複写する。ブロック画像は、入力画像 1 0 7 と 1 0 8 について、それぞれ B K 1、B K 2 と 2 つあるので、ブロック画像の複写においては、これらブロック画像（B K 1、B K 2）を加算平均処理して任意視点のブロック画像を生成する。

【 0 0 6 6 】

ここで、B K 1 と B K 2 の加重平均処理について説明する。B K 1 は、 $B K 1(x, y)$ で表される 2 次元の画素配列である。同様に B K 2 は、 $B K 2(x, y)$ で表される 2 次元の画素配列である。新たに生成されるブロック画像 B K 3 の 2 次元の画素配列 $B K 3(x, y)$ は、以下の数 2 0 で表されるように生成される。A および B は $A + B = 1$ になるように実数の範囲で値を設定してよい。例えば、入力画像 1 0 7 と任意視点画像 1 0 6 の視点位置間の距離を K、入力画像 1 0 8 と任意視点画像 1 0 6 の視点位置間の距離を L とした場合に、 $A = L / (K + L)$ 、 $B = K / (K + L)$ のように、距離に逆比例する重みづけを行なうことが考えられる。なお、 $B K 1(x, y)$ は、 $B K 3(x, y)$ を通る第 1 の光線関数上の視差情報により決定される点と入力画像 1 0 7 の視点とにより得られる第 2 の光線関数が入力画像投影面と交わる点である。同様に、 $B K 2(x, y)$ は、 $B K 3(x, y)$ を通る第 1 の光線関数上の視差情報により決定される点と入力画像 1 0 8 の視点とにより得られる第 2 の光線関数が入力画像投影面と交わる点である。

【 0 0 6 7 】

$$B K 3(x, y) = A \times B K 1(x, y) + B \times B K 2(x, y) \quad \text{数 2 0}$$

S 1 1 0 5 は、ブロック更新終了判定ステップである。CPU 2 0 1 は、S 1 1 0 2 で総ブロック数 $(M + 1) \times (N + 1)$ のブロックが全て選択されたら本サブフローチャート - 3 の処理は終了であり、そうでなければ S 1 1 0 1 へ処理を進める。

【 0 0 6 8 】

以上が、図 5 に示される本処理プログラムの処理である。以上の説明から明らかなように、本実施形態においては、視差情報を算出するのに、入力画像の特徴点を必要としない。また、入力画像に基づく光線関数を用いて視差を算出するのではなく、任意視点映像に基づく光線関数を用いて視差を算出するので、任意視点画像の画素の配置座標値が幾何的に正確となる視差情報を算出することが可能になる。

【 0 0 6 9 】

以上のように、上記実施形態によれば、入力画像に特徴点が存在しない画像領域においても、幾何的に正確な視差情報を算出することが可能になり、任意視点画像の画質を向上させることができる。

【 0 0 7 0 】

なお、上記実施形態では、2 つの入力画像から任意視点の画像を生成する例を説明したが、3 つ以上の入力画像を用いて任意視点の画像を生成できる構成としてもよいことは明

10

20

30

40

50

らかである。

【0071】

また、上記実施形態では、視差情報（視差ベクトル）を全てのブロックについて決定する視差情報マッピングを行ない（図6）、その後、視差情報を読み出しながら任意視点画像を生成する（図11）が、これに限られるものではない。たとえば、静止画像を処理する場合は、図6のステップS607において1つのブロックに関して視差情報（任意視点からの距離）が確定しているので、この時点で該1つのブロックについてステップS1103、S1104を実行するようにしてもよい。このようにすれば、全ブロックに対応した視差マップを保持しておくことが不要になる。他方、視差マップを生成し、保持しておけば、入力画像の視点および任意視点の3者の位置関係が変化しない限り、生成した視差マップを利用して任意視点からの画像を生成することができる。また、動画像を処理する場合は、保持された視差マップを用いて、入力画像の各フレームから任意視点画像の各フレームを生成できる。

10

【0072】

[実施形態2]

本発明の実施形態1では、視差情報を生成し、その視差情報から任意視点を生成した。本実施形態では、コンピュータグラフィックス（CG）のレンダリングに用いられる周囲環境光画像を、複数の視点画像をもとに、任意位置で生成する例を示す。

【0073】

一般的に、CGのレンダリングを行う場合には、物体などのオブジェクトデータと、光源データが必要になる。特に、光源データについては、あるオブジェクトが光源となる場合と、レンダリングする周囲環境の明るさを画像として保持し、その画像を光源データとしてレンダリングする場合とがある。ここで、視点画像を周囲の明るさ画像とみなしたものを、周囲環境光画像と呼ぶ。周囲環境光画像は、レンダリング対象となる被写体位置に応じて取得する必要がある。

20

【0074】

例えば、図13（A）に示すように、あるオブジェクト（被写体）1301および1302からなるシーンを、視点1303から、周囲環境光画像を光源としてレンダリングする場合、周囲環境光画像は1304に示す様に位置Tを中心とした全周囲の画像が必要となる。また、周囲環境光画像1304は、図13（B）に示すように、被写体が位置Tにある場合は周囲環境光画像1304となる。一方で、被写体が位置Aにある場合は周囲環境光画像1305、また、位置Bにある場合は周囲環境光画像1306となる。すなわち、被写体の位置に応じた周囲環境光画像が必要となる。そのため、任意の位置で周囲環境光画像を用いたCGレンダリングを行うためには、無数の周囲環境光画像が必要となる。ここで、本実施形態によれば、位置Aの周囲環境光画像1305と、位置Cの周囲環境光画像1306から、位置Tの周囲環境光画像1304を生成することができる。つまり、任意の周囲環境光画像を、少なくとも2か所の視点画像より生成可能となる。そのため、無数の周囲環境光画像を用意することなく、任意位置での周囲環境光画像を用いてのCGレンダリングが可能となる。

30

【0075】

また、周囲環境光画像を生成するにあたり、光源としての寄与が高いのは画像の明るい部分、すなわち、画素値の高い部分である。そこで、補間元の視点画像ブロックのうち、平均画素値の高い部分については詳細にブロック画像を生成し、平均画素値の低い部分については粗いブロック画像を生成する。このようにすることで、周囲環境光画像の生成を高速に行うことができる。

40

【0076】

以下、図を用いて、本実施形態の処理と実施形態1の処理との違いについて述べる。本実施形態では、実施形態1と比較して、図5におけるS502処理の内容が異なる。具体的には、生成する視差マップのうち、CGレンダリングにおける周囲環境光源としての寄与が大きいと考えられる平均画素値の高いブロック画像は通常通り詳細な視差マップを作

50

成し、平均画素値の低いブロックは、視差マップを粗く作成する。

【 0 0 7 7 】

図 1 2 は、実施形態 2 における、視差マップ生成処理（周囲環境光生成処理）の概要を示すフローチャートである。以下、図 1 2 のフローチャートを本サブフローチャート 4 とする。なお、図 1 2 の S 1 0 2 4 ~ S 1 2 1 0 は、図 6 の S 6 0 2 ~ S 6 0 7 と同様の処理を行うので、説明を省略する。

【 0 0 7 8 】

まず、S 1 2 0 1 で、任意視点画像における所定の画像領域を指定する。ここでは、図 1 4 (A) に示すように、画像領域は $(N + 1) \times (M + 1)$ 個のブロックに分割され、 (m_p, n_q) 、 (m_{p+1}, n_{p+1}) で囲まれる位置の画素値が明るくなっていると仮定する。また、各ブロックは、図 9 と同様、縦 3 画素、横 3 画素で区切られているとする。なお、ブロックの個数やブロック内の画素数は、特に上記の数に規定されるものではない。

10

【 0 0 7 9 】

次に、S 1 2 0 2 で、各ブロック画像の平均画素値を計算する。具体的には、以下のようにして平均画素値 A_{img} を求める。

$$A_{img} = (1 / n_{img}) \sum I_{img} \quad \text{数 2 1}$$

ここで、 n_{img} はある視点画像ブロックの画素数、 I_{img} はある視点画像ブロックの画素値である。例えば、ある視点画像から周囲環境光画像を作成する場合は、それぞれの視点画像に対応するブロック画像が $B_k(m_p, n_q)$ 、平均画素値は $A_{img}(m_p, n_q)$ 、となる。

20

【 0 0 8 0 】

次に、S 1 2 0 3 で、ブロック画像の解像度を決定する。ここで、各ブロック画像の平均画素値 $A_{img}(m_p, n_q)$ が所定の閾値を超えた場合には、図 1 4 (B) に示す様に、そのようなブロックの解像度を縦 3 画素、横 3 画素とする。超えなかった場合は、図 1 4 (B) に示す様に、そのようなブロックの解像度を 1 画素とし、画素値として中心 1 画素を代表値とする。なお、本実施形態では、ブロック画像の平均画素値と比較する所定の閾値を 1 つと設定し、また、閾値以上であればブロック内画素値を 3×3 画素、閾値より小さければ 1 画素、と設定したが、閾値の数や、ブロック内の画素数はこれに限定するものではない。その後、S 1 2 0 4 ~ S 1 2 1 0 においてそれぞれの処理が行われる。

30

【 0 0 8 1 】

以上のようにして、本サブフローチャート 4 を行うことによって、視点画像のうち、画素値の高い部分、すなわち、光源としての寄与が高い部分に関して優先的に計算を行うためのブロック画像を生成することができる。

【 0 0 8 2 】

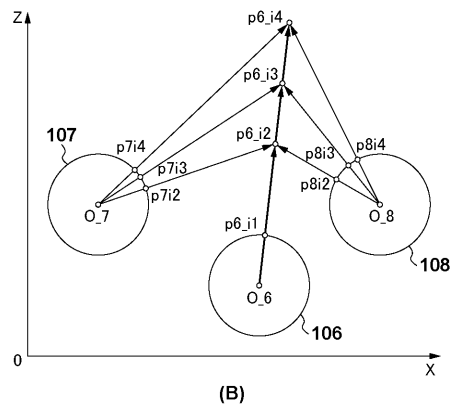
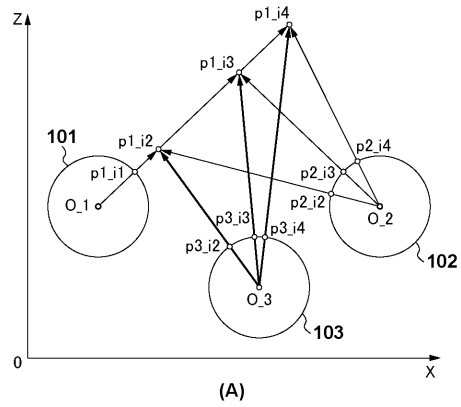
以上のように、本実施形態によれば、CG レンダリングのための任意位置での周囲環境光画像を高速に生成することができる。

【 0 0 8 3 】

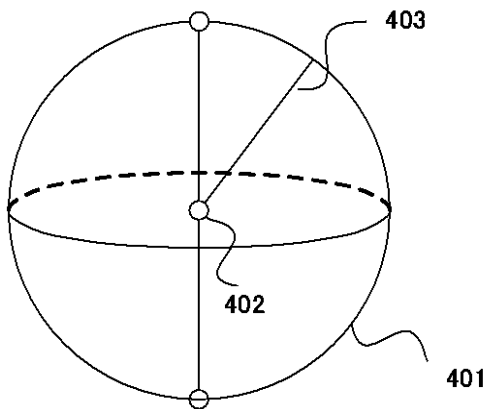
また、本発明は、以下の処理を実行することによっても実現される。即ち、上述した実施形態の機能を実現するソフトウェア（プログラム）を、ネットワーク又は各種記憶媒体を介してシステム或いは装置に供給し、そのシステム或いは装置のコンピュータ（または CPU や MPU 等）がプログラムを読み出して実行する処理である。

40

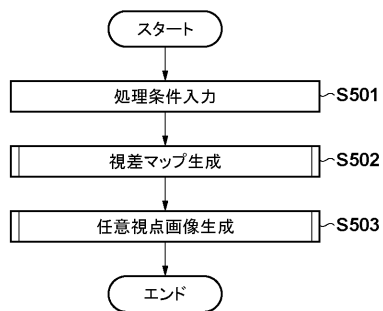
【図 1】



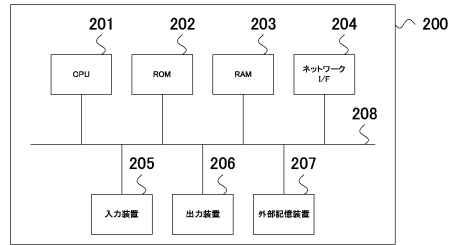
【図 4】



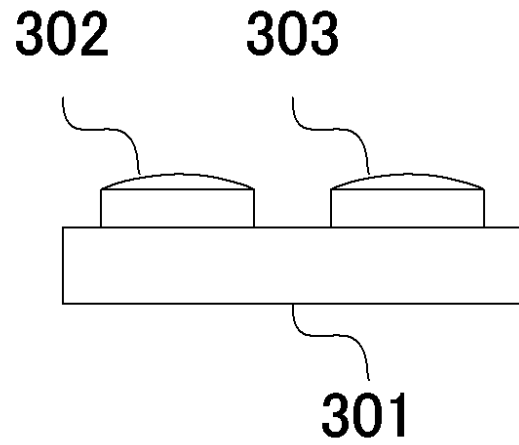
【図 5】



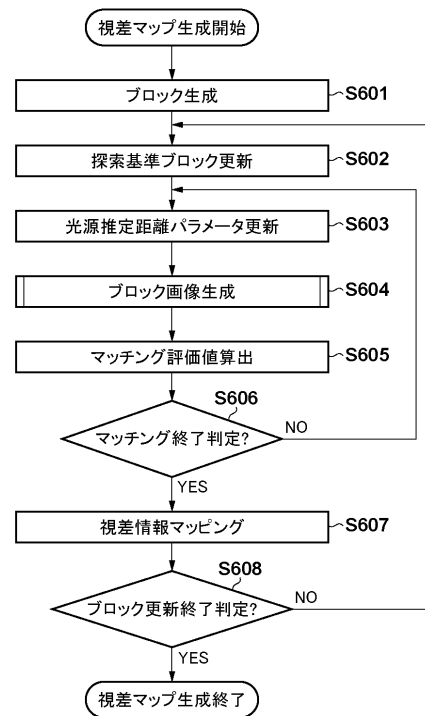
【図 2】



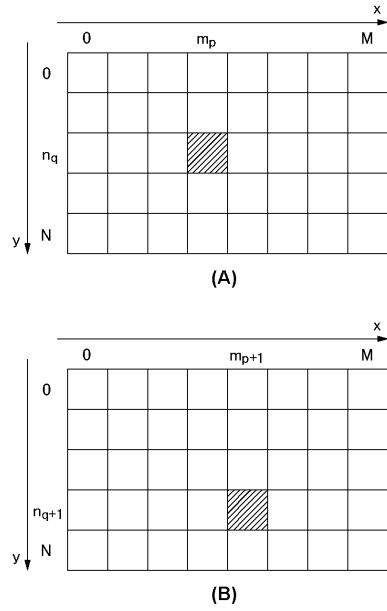
【図 3】



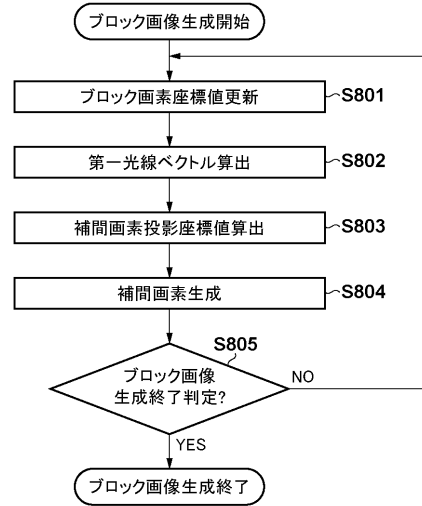
【図 6】



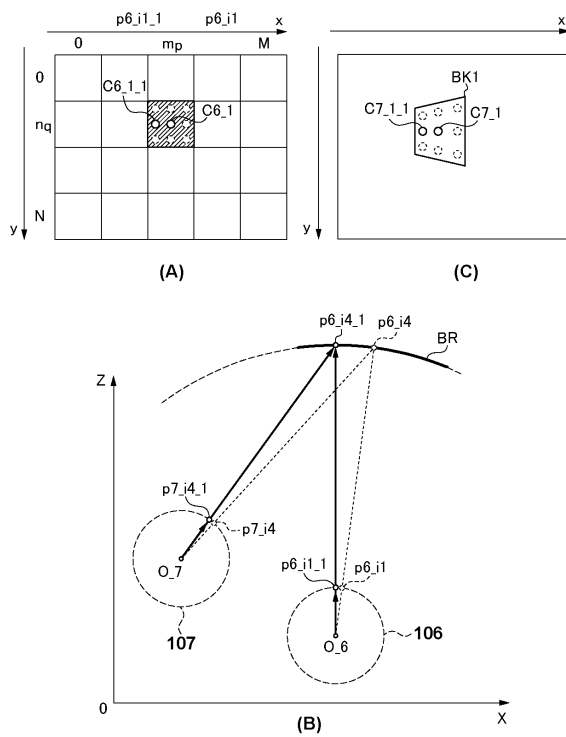
【図 7】



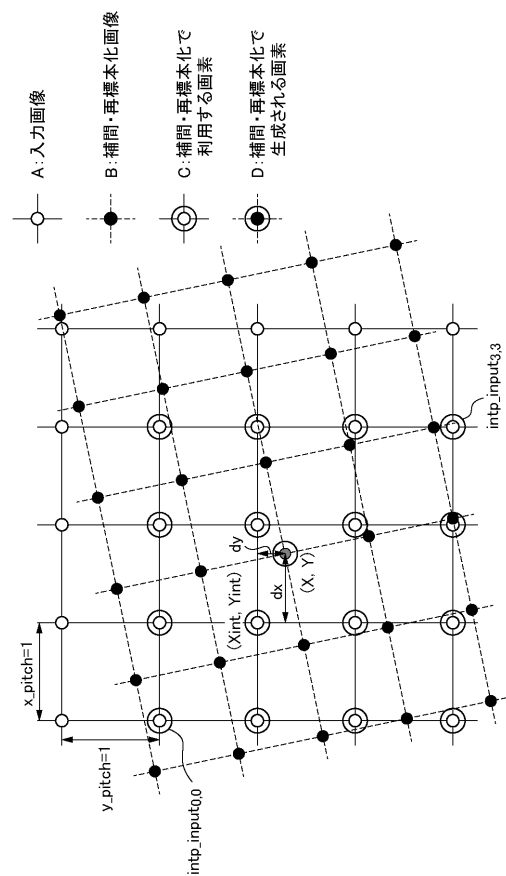
【図 8】



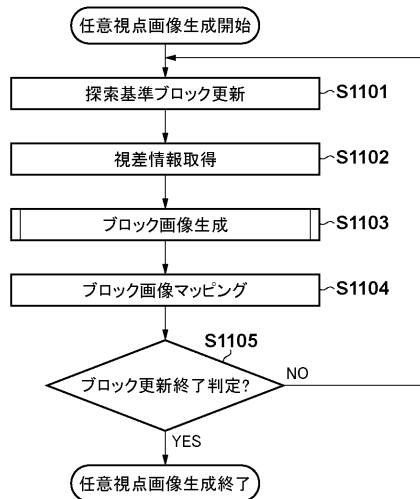
【図 9】



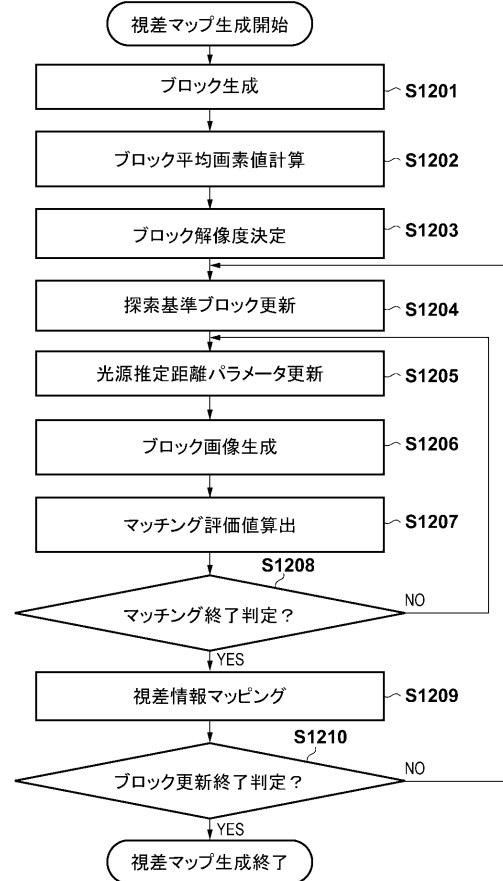
【図 10】



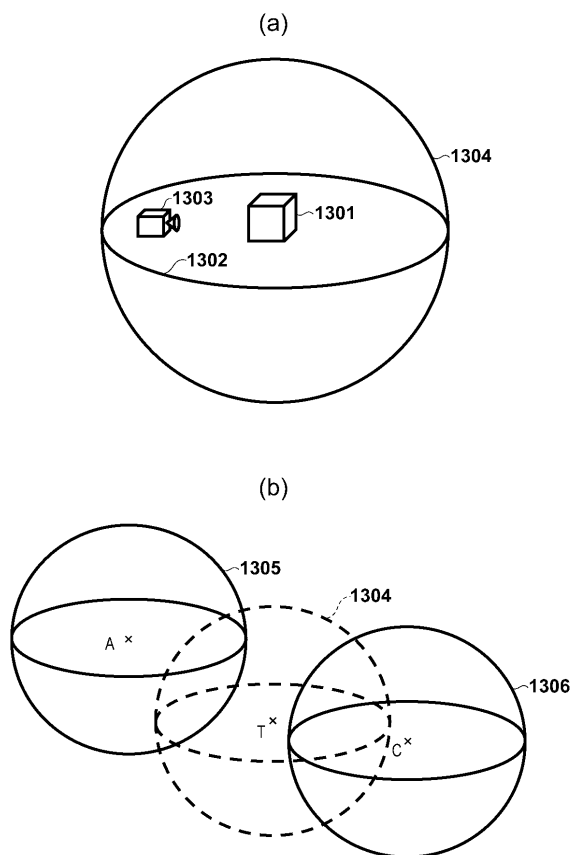
【図 1 1】



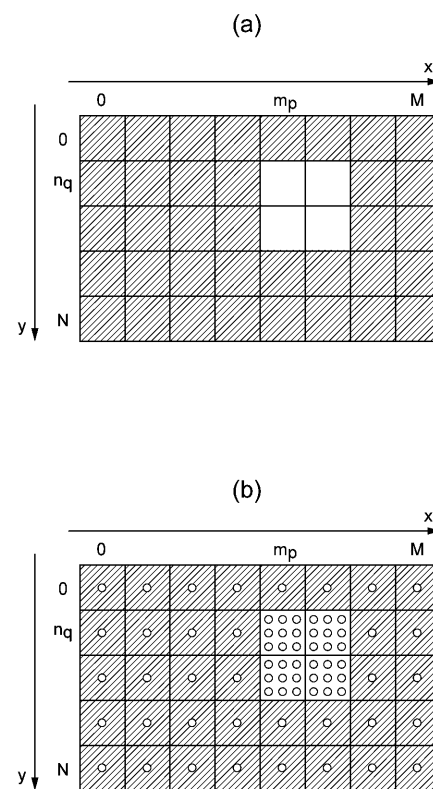
【図 1 2】



【図 1 3】



【図 1 4】



フロントページの続き

- (72)発明者 伊藤 靖浩
東京都大田区下丸子3丁目30番2号 キヤノン株式会社内
- (72)発明者 佐藤 陽輔
東京都大田区下丸子3丁目30番2号 キヤノン株式会社内

審査官 村松 貴士

- (56)参考文献 米国特許第06975756(US, B1)
特開2002-216163(JP, A)
特開2004-246667(JP, A)
中村芳浩, 外1名, “Visual Hullを利用した多視点画像からの仮想視点画像生成”, 情報処理学会研究報告, 社団法人情報処理学会, 2002年 5月 9日, Vol. 2002, No. 34, p. 111-118
- (58)調査した分野(Int.Cl., DB名)
G06T 1/00
G06T 15/00 - 19/20