

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2009-95015

(P2009-95015A)

(43) 公開日 平成21年4月30日(2009.4.30)

(51) Int.Cl.	F 1	テーマコード (参考)
H04N 5/74 (2006.01)	H04N 5/74	2K103
G03B 21/14 (2006.01)	G03B 21/14	5C058
G09G 3/20 (2006.01)	G09G 3/20	5C060
G09G 5/00 (2006.01)	G09G 3/20	5C080
H04N 9/31 (2006.01)	G09G 3/20	5C082

審査請求 未請求 請求項の数 17 O L (全 100 頁) 最終頁に続く

(21) 出願番号	特願2008-242132 (P2008-242132)	(71) 出願人	000002369 セイコーエプソン株式会社 東京都新宿区西新宿2丁目4番1号
(22) 出願日	平成20年9月22日 (2008.9.22)	(74) 代理人	100095728 弁理士 上柳 雅善
(31) 優先権主張番号	60/978,015	(74) 代理人	100107261 弁理士 須澤 修
(32) 優先日	平成19年10月5日 (2007.10.5)	(74) 代理人	100127661 弁理士 宮坂 一彦
(33) 優先権主張国	米国(US)	(72) 発明者	ヤング チャオ アメリカ合衆国 ロード アイランド州 プロビデンス ホーブ ストリート 18 2、ボックス D
(31) 優先権主張番号	12/179,371		
(32) 優先日	平成20年7月24日 (2008.7.24)		
(33) 優先権主張国	米国(US)		

最終頁に続く

(54) 【発明の名称】異なるデータ間の相互関係を物理的メモリ内で配列する方法

(57) 【要約】

【課題】望ましい画像を作成するために何をプロジェクタが映写する必要があるかをどの

ように判定するかという問題に光輸送行列の逆を用いて対応し、さらにプロジェクタ-カメラ

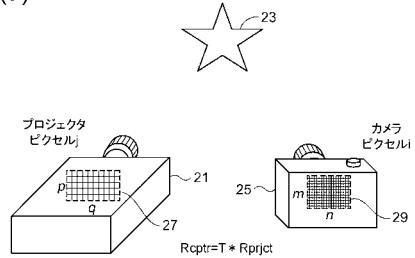
システムの較正を簡素化すること。

【解決手段】並列環境において行列を操作するためのデータ構造を最適化する方法および

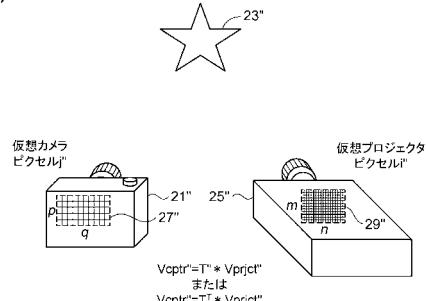
システムは演算分岐を制限する。データ構造は線形データ格納および複数の処理スレッド間の同期化でさらに最適化される。

【選択図】図2

(a)



(b)



【特許請求の範囲】

【請求項 1】

プロジェクタをカメラに関連付ける較正情報で定義される異なるデータ間の相互関係を物理的メモリ内で配列する方法であって、

(a) プロジェクタのピクセルバッファ、カメラピクセルの座標バッファ、およびカメラピクセルのRGBバッファを定義するステップと、

(b) 前記プロジェクタの各プロジェクタピクセルについて前記較正情報に従い、前記プロジェクタピクセルに対し較正関係を有するカメラピクセルを関連カメラピクセルグループに収集するステップと、

(c) 前記カメラピクセルの座標バッファ内で、前記カメラ内の各カメラピクセルのX-Y座標位置情報セットを維持することで、前記X-Y座標位置情報セットは前記関連カメラピクセルグループに対応する座標グループに配列され、各前記座標グループ内におけるX-Y座標位置情報セットは前記カメラピクセルの座標バッファ内で連続的に配列されるステップと、

(d) 前記カメラピクセルのRGBバッファ内で、前記較正情報で判定される各カメラピクセルの較正RGB値を維持することで、前記較正RGB値は前記関連カメラピクセルグループに対応するRGBグループに配列され、各RGBグループ内における較正RGB値セットは前記カメラピクセルの座標バッファ内におけるX-Y座標位置情報のセットの配列を映すような形で前記カメラピクセルRGBバッファ内において連続的に配列されるステップと、

(e) 前記プロジェクタピクセルのバッファ内で、前記プロジェクタのプロジェクタピクセル毎に別個のプロジェクタのピクセルデータセットを定義することで、前記プロジェクタのピクセルデータセットは各プロジェクタのピクセルデータセットと前記プロジェクタからそれに相關するプロジェクタピクセルとの間で識別可能な1対1の相關を維持するように前記プロジェクタ内のプロジェクタピクセル配列に相關する形で配列されるステップと、を有し、

前記ステップ(e)において、前記プロジェクタのピクセルデータセットは前記ステップ(b)で定義された対応する前記関連カメラピクセルグループ内のカメラピクセル数を示す関連カメラピクセル総数を含み、前記プロジェクタのピクセルデータセットはさらに対応する関連カメラピクセルグループ内の第1カメラピクセルの、前記カメラピクセルの座標バッファ内における格納位置を特定する位置インジケータを含むことを特徴とする異なるデータ間の相互関係を物理的メモリ内で配列する方法。

【請求項 2】

前記ステップ(e)において、前記プロジェクタのピクセルデータセットは、前記プロジェクタ内におけるプロジェクタピクセルの連鎖配列に相關した連鎖の形で配列されることを特徴とする請求項1に記載の異なるデータ間の相互関係を物理的メモリ内で配列する方法。

【請求項 3】

特定のプロジェクタピクセルに関連する任意のカメラピクセルのX-Y座標位置情報セットは、前記特定のプロジェクタピクセルに対応する関連カメラピクセルグループから望まれるカメラピクセルを選択するステップと、

前記プロジェクタピクセルのバッファ内で前記特定のプロジェクタピクセルに相關するプロジェクタのピクセルデータセットを特定するステップと、

前記特定されたプロジェクタのピクセルデータセット内の位置インジケータをアクセスするステップと、

望まれるカメラピクセルの関連カメラピクセルグループ内における連鎖位置の場所で判定されたオフセット分だけ前記アクセスされた位置インジケータを増分することにより目標アドレスを生成するステップと、

前記目標アドレスにおけるカメラピクセルの座標バッファ内に格納されるX-Y座標位置情報セットにアクセスするステップとにより得られることを特徴とする請求項1に記載

10

20

30

40

50

の異なるデータ間の相互関係を物理的メモリ内で配列する方法。

【請求項 4】

前記望まれるカメラピクセルの較正 R G B 値は、前記目標アドレスにおける前記カメラピクセルの R G B バッファ内に格納される較正 R G B 値をアクセスすることにより得られることを特徴とする請求項 3 に記載の異なるデータ間の相互関係を物理的メモリ内で配列する方法。

【請求項 5】

前記カメラピクセルの座標バッファにおけるアドレス増分は X - Y 座標位置情報セットにより決定され、

前記カメラピクセルの R G B バッファにおけるアドレス増分は較正 R G B 値により決定され、

前記各カメラピクセルの較正 R G B 値のメモリ格納は各 X - Y 座標位置情報セットとは異なったバイト数を必要とすることを特徴とする請求項 1 に記載の異なるデータ間の相互関係を物理的メモリ内で配列する方法。

【請求項 6】

複数の量子化数値レベルを定義するステップと、

前記関連カメラピクセルグループ各々内のカメラピクセル数に従い前記関連カメラピクセルグループを対応量子化数値レベルに収集するステップとをさらに有することを特徴とする請求項 1 に記載の異なるデータ間の相互関係を物理的メモリ内で配列する方法。

【請求項 7】

個別の量子化数値レベル内における関連カメラピクセルグループの収集をマルチコア処理システムの異なった処理コアに対し使用可能にするステップをさらに有することを特徴とする請求項 6 に記載の異なるデータ間の相互関係を物理的メモリ内で配列する方法。

【請求項 8】

前記複数の量子化数値レベルを定義する前に、前記関連カメラピクセルグループの最小および最大のものを特定することによって最小から最大までの幅を定義するステップと、前記量子化数値レベルを前記最小から最大までの幅の範囲内になるよう定義するステップとをさらに有することを特徴とする請求項 6 に記載の異なるデータ間の相互関係を物理的メモリ内で配列する方法。

【請求項 9】

前記量子化レベルは前記最小から最大までの幅の範囲内の区分として目標関連カメラピクセルグループを選択することによりさらに定義し、前記量子化レベルは前記目標関連カメラピクセルグループ内のカメラピクセル数に等しく設定されることを特徴とする請求項 8 に記載の異なるデータ間の相互関係を物理的メモリ内で配列する方法。

【請求項 10】

各量子化レベル内において、量子化レベルより少ないカメラピクセルを有する関連カメラピクセルグループは R G B 値がゼロであるダミーカメラピクセルで嵩上げされ各々のカメラピクセルの合計数をその量子化レベルに引き上げることを特徴とする請求項 6 に記載の異なるデータ間の相互関係を物理的メモリ内で配列する方法。

【請求項 11】

前記各量子化レベル内において、量子化レベルは前記ダミーカメラピクセルによる嵩上げ前の最大関連カメラピクセルグループに等しく設定されることを特徴とする請求項 10 に記載の異なるデータ間の相互関係を物理的メモリ内で配列する方法。

【請求項 12】

第 1 バイトサイズの第 1 データ単位を有する第 1 メモリを提供するステップをさらに有し、

前記カメラピクセルの R G B バッファは、前記第 1 バイトサイズより小さいバイトサイズを有する第 2 データ単位を有し、

前記カメラピクセルの R G B バッファは、前記第 1 メモリから第 1 データ単位で埋められ、

10

20

30

40

50

前記カメラピクセルの R G B バッファからのデータは前記第 2 データ単位でアクセスされることを特徴とする請求項 1 に記載の異なるデータ間の相互関係を物理的メモリ内で配列する方法。

【請求項 1 3】

前記第 1 データ単位は前記第 2 データ単位より 16 倍大きいことを特徴とする請求項 1 2 に記載の異なるデータ間の相互関係を物理的メモリ内で配列する方法。

【請求項 1 4】

各々偶数のバイトで定義される第 1 データ単位を有する第 1 メモリを提供するステップと、

前記第 1 メモリを奇数バイトのデータで埋めるステップと、

前記第 1 メモリ内における前記第 1 データ単位の 1 つ内にある前記奇数バイトのデータの選択された 1 つを、前記奇数バイトデータ内のバイト数にオフセット値を掛け、結果を前記第 1 データ単位内のバイト数で割ることによりアクセスするステップとをさらに有することを特徴とする請求項 1 に記載の異なるデータ間の相互関係を物理的メモリ内で配列する方法。

【請求項 1 5】

前記オフセット値は開始点から前記奇数バイトのデータの選択された 1 つまで増分する必要のある奇数バイトのデータの数となるよう選択されることを特徴とする請求項 1 4 に記載の異なるデータ間の相互関係を物理的メモリ内で配列する方法。

【請求項 1 6】

前記奇数バイトのデータは 3 バイトの R G B データであり、前記カメラピクセルの R G B バッファは前記第 1 メモリからの奇数バイトのデータ単位で埋められることを特徴とする請求項 1 4 に記載の異なるデータ間の相互関係を物理的メモリ内で配列する方法。

【請求項 1 7】

前記較正情報は以下のステップにより定義されるビュー投影行列を有することを特徴とする請求項 1 に記載の異なるデータ間の相互関係を物理的メモリ内で配列する方法。

(1) 前記プロジェクタを前記カメラに関連付ける光輸送行列 T を得るステップと、

(2) 前記光輸送行列 T における各行について前記光輸送行列の共通行に沿って行列入力を比較し、共通行における最高値の行列入力を留保し前記共通行における残りの入力にゼロ値を割り当て、改変光輸送行列 T^* を作成するステップと、

(3) 前記改変光輸送行列 T^* における各列を順に目標列として特定し、各目標列について前記目標列における非ゼロ値に対し前記目標列に関し正規化値を計算するステップと、

(4) 前記改変光輸送行列 T^* と同じサイズの中間行列を作成するステップと、

(5) 前記中間行列の各列を前記改変光輸送行列 T^* において対応する目標列の計算された正規化値で埋めることで、前記中間行列において埋められた各列の各正規化値は前記改変光輸送行列 T^* において対応する列の前記非ゼロの入力値と 1 対 1 の対応を維持するステップと、

(6) 前記中間行列に転置行列演算を適用してビュー投影行列を作成するステップ。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は並列計算環境において行列の操作を最適化する方法に関する。より具体的に、本発明は行列データ構造を最適化し、並列処理環境において最小限の計算プランチングで行列の操作を促進する方法に関する。本出願は T i t l e 3 5 , U . S . C . § 1 1 9 (e) の下で 2007 年 10 月 5 日に出願された米国特許出願番号 6 0 / 9 7 3 , 3 5 1 の特典を主張する。

【背景技術】

【0002】

プロジェクタとカメラを組み合わせると、光を投射しつつ取り込むことのできるハイブ

10

20

30

40

50

リッド装置およびシステムが生まれる。この新たに台頭してきた撮像装置およびシステムは研究の世界においてプロジェクタ-カメラシステムとして知られている。通常映写表面の幾何学的形状など表示環境に関する属性を推定するためには1台以上のカメラで取り込まれた画像が使用される。次にこれらのプロジェクタ-カメラシステムのプロジェクタが映像を適応させて、映写表面の形状不規則性を補正し、もたらされる画像を向上させる。言い換れば、カメラを使用することによりプロジェクタは投影画像における歪みを「見る」ことができ、それにより投影画像を調節して観察される歪みを低減することができる。

【0003】

これを達成するために、観察される画像の歪みは投影環境の凹凸（すなわち表面上の凹凸）によるもので、プロジェクタまたはカメラに内在する不整または互いの相対的配向によるものではないことを保証すべく、カメラとプロジェクタは互いの撮像パラメータに対し較正される必要がある。

【0004】

このように、プロジェクタ-カメラシステムの製作者が解決しなければならない重要な問題は各装置の内部撮像パラメータ（すなわち固有パラメータ）の判定およびシステム内のすべてのプロジェクタおよびカメラ間の幾何学的関係（すなわち外的パラメータ）の判定である。この問題は通常システム較正の問題と呼ばれる。

【0005】

しかしシステムが実質的に較正された後でも、映写を調節して映像の歪みを補正することは簡単でない。映写歪みを識別し補正することは非常に演算集約的な作業であり得、そのため従来非専門分野への適用は相当限定されていた。

【0006】

プロジェクタ-カメラシステムの較正をより良く理解するために、出願者はコンピュータ・ビジョンの分野で見られるマルチカメラ画像化システムを研究した。このようなマルチカメラ画像化システムは画像撮影装置のみで構成され、画像投影装置は含まないが、システムの較正に関しコンピュータ・ビジョンの分野において一連の研究があり、複数の画像撮影装置ではあるもののこれらの方から複数装置の較正に関し何らかの識見が得られるのではないかと思われた。

【0007】

コンピュータ・ビジョンの世界で通常用いられる方法はZhengyou Zhang著の論文「適応性を有するカメラ較正用の新規手法」（「A flexible new technique for camera calibration」）、IEEEパターン分析および人工知能紀要（Transactions on Pattern Analysis and Machine Intelligence）、22(11):1330～1334、2000年、に記載され、引用により全体として本明細書に組み入れられる。この方法において、いくつかの既知の特徴点（通常グリッドを形成）でマークされた平たい被写体をカメラに対しさまざまな角度のポーズに置き、平たい被写体の複数の画像がカメラにより取り込まれる。各特徴点の画像位置が抽出され、各特徴点の相対位置は知られているので次に特徴点位置の集合を用いてカメラを較正することができる。撮像システムに2台以上のカメラが存在する場合、存在するカメラ間の幾何学的関係と併せて固有パラメータはすべてのカメラに平たい被写体の各ポーズ角度における画像を取り込ませることにより推定することができる。

【0008】

プロジェクタとカメラは影像化の幾何学という点で非常に類似しているので、複数カメラの画像化システムにおけるカメラの較正に適した手法はプロジェクタ-カメラシステムにおけるプロジェクタの較正に適すると仮定することが合理的と思われるかもしれない。しかし、カメラの較正手法はすべて較正を必要とするカメラ（すなわち較正される撮像装置）が画像をいくつか取り込むことが必要であるため、プロジェクタは画像を取り込むことができないのでカメラの較正手法はプロジェクタに容易に適用できないようである。

【0009】

従って、従来のプロジェクタ-カメラシステムでは、プロジェクタに加え少なくとも2

10

20

30

40

50

台のカメラが必要とされている。まず2台のカメラが通常複数カメラの撮像システム較正手法を用いて較正され、ステレオカメラの対を確立する。より具体的に、これらのシステムは「ブートストラップ的」手順を用いて2台のカメラを較正しステレオの対を形成する。業界で知られるように、ステレオカメラの対は深度を推定する（すなわち擬似遠近の光景を達成する）のに用い、ステレオカメラの対に見える特徴点の外見上の深度知覚を確立することができる。較正されたステレオカメラの対は次にプロジェクタを較正するために用いられる。基本的にこの外見上の深度知覚の確立を用い映写表面の、従って、映写表面に投影された映像の、表面深度凹凸を識別する。プロジェクタを次に較正し、投影画像における表面深度凹凸を補正することができる。基本的に、外見上の深度知覚を用いてプロジェクタを較正するためには、まずプロジェクタに特徴点を、凹凸表面を有する可能性のある表示環境（すなわち映写表面）に投影させる。予め較正されたステレオカメラの対を用いて投射された点の遠近的深度位置を解像する。次に投射された点の深度位置による判定に基づき映写表面における表面／深度凹凸について補正するようにプロジェクタを較正することができる。このブートストラップ的手法はプロジェクタ・カメラシステム向けの実証済みの較正方法であるが、予め較正され、好都合に配置された外付けのステレオカメラの対を必要とし、オペレータの介入を多く必要とするため内蔵型のプロジェクタ・カメラシステムには適用できない。

10

【0010】

同様に関連するものとしてデュアル・フォトグラフィと呼ばれる手法があり、Sen他により論文「デュアル・フォトグラフィ」（「Dual Photography」）、ACM SIGGRAPH予稿、2005年、の中で提案されており、引用により本明細書に全体として組み入れられる。デュアル・フォトグラフィはヘルムホルツの相反定理を利用して実カメラで取り込んだ画像を用いプロジェクタから「見えた」（または事実上「取り込んだ」）画像をシミュレートする擬似画像（すなわちデュアル画像）を合成する。つまり、擬似画像はプロジェクタから「見た」取り込み画像をシミュレートし、従って、プロジェクタが画像を取り込むことができたとした場合プロジェクタにより取り込まれた画像がどうなるかを表す。この方法はプロジェクタを擬似カメラとして扱うことを可能にするので、プロジェクタの較正に関する問題のいくつかを除去できるかもしれない。

20

【0011】

ヘルムホルツの相反定理は、光の流れはその輸送特性を変えることなく有効に逆向きにできるという概念に基づいている。ヘルムホルツの相反定理は計算の複雑さを低減するために多くのコンピュータ・グラフィックス用途に使用されている。コンピュータ・グラフィックス関連の文献でこの相反定理は入射（ i ）および出射（ o ）方向間の反射強度転移の対称性を説明する方程式として通常 $f_r(i, o) = f_r(o, i)$ と要約され、ここで f_r はある表面の双方反射率分布関数（BRDF）を表す。

30

【0012】

このようにデュアル・フォトグラフィは、理想的には映像および取り込み画像のデュアルな性質（すなわちデュアリティ関係）を利用して前者を後者からシミュレートする。さらに詳細に後述するように、デュアル・フォトグラフィ（より正確にはヘルムホルツの相反定理）はカメラとプロジェクタ間の光輸送特性を取り込む必要がある。より具体的に、デュアル・フォトグラフィは放射光線を取り込み光線に関連付ける光輸送特性（すなわち光輸送係数）の判定を必要とする。

40

【0013】

しかしデジタルカメラおよびデジタルプロジェクタを扱う場合、デュアル・フォトグラフィは各プロジェクタピクセル（すなわちすべての放射光線）を1つ1つのカメラピクセル（すなわち放射光線の一部を取り込むすべての光センサ）に関連づける別個の光輸送係数を双方の装置の解像度において取り込む必要がある。デジタルプロジェクタおよびデジタルカメラ双方とも各々何百万のピクセルを有し得るので、多数の光輸送係数の獲得、格納、および操作はこの手法の実際的な使用を制限する可能性がある。従って、デュアル・フォトグラフィは理論的に非常に利点を提供するように見えるが、実際上デュアル・フォ

50

トグラフィは非常に大量のコンピュータメモリ（アーカイブ用のディスク型メモリおよびアクティブ・ソリッドステート・メモリ双方）を必要とし、広範囲コンピュータ処理能力を要し、機器を設定してプロジェクタ・カメラシステムが使用されるすべての投影環境に対する大量の光線を放射し取り込むために多くの時間とユーザ介入を必要とするという物理的かつ非実用的な要件により深刻に制限される。

【0014】

デュアル・フォトグラフィのより明確な理解は図1(a)および図1(b)を参照して得られる。図1(a)において「根本構成」（すなわちデュアル変換以前の実際の物理的な装置の構成）は実デジタルプロジェクタ11、実映像13、および実デジタルカメラ15を含む。実デジタルプロジェクタ11から光が発射され、実デジタルカメラ15により取り込まれる。映写された各投射光線（実デジタルプロジェクタ11内の各プロジェクタピクセルeから）を対応する各取り込み光線（実デジタルカメラ15内の各カメラピクセルgにおいて取り込まれた）に関連づける係数は光輸送係数と呼ばれる。光輸送係数を用いると、取り込まれた光線から映写された光線の特徴を判定することが可能となる。

10

【0015】

本例において、実デジタルプロジェクタ11は象徴的に点線枠として示され、s行およびr列の個々のプロジェクタピクセルeからなるプロジェクタのピクセルアレイ17を有するデジタルプロジェクタであることが好ましい。各プロジェクタピクセルeは別個に放射された光線の源であり得る。プロジェクタのピクセルアレイ17のサイズは実デジタルプロジェクタ11の解像度に依存する。例えばVGA解像度は 640×480 ピクセルのアレイ（すなわち307,200個のプロジェクタピクセルe）からなることができ、SVGA解像度は 800×600 ピクセルのアレイ（すなわち480,000個のプロジェクタピクセルe）を有し得、XVG解像度は 1024×768 ピクセルのアレイ（すなわち786,732個のプロジェクタピクセルe）を有し得、SXVG解像度は 1280×1024 ピクセルのアレイ（すなわち1,310,720個のプロジェクタピクセルe）を有し得、以下同様により高い解像度のプロジェクタはより多数のプロジェクタピクセルeを必要とする。

20

【0016】

同様に実デジタルカメラ15は象徴的に点線枠として示され、u行およびv列の個々のカメラピクセルgからなるカメラピクセルアレイ19を有するデジタルカメラである。各カメラピクセルgは放射された光線の一部を受光する、すなわち取り込む、ことができる。カメラピクセルアレイ19のサイズもまた実デジタルカメラ15の解像度に依存する。但し、実デジタルカメラ15は4メガピクセル（すなわち4,194,304の受容を有するカメラピクセルg）以上を有しているのが普通である。

30

【0017】

カメラピクセルアレイ19内の各カメラピクセルgは区別可能なプロジェクタピクセルeから放射された光線の一部を取り込むことができ、別々のプロジェクタピクセルe各々は別個の光線を放射し得るので、別々のプロジェクタピクセルe各々をすべてのカメラピクセルgに関連付けるためには大量の光輸送係数が必要となる。言い換えれば、单一のプロジェクタピクセルeから放射された光線はカメラピクセルアレイ19全体を覆う可能性があり、従って、各カメラピクセルgは放射された光線の異なった量を取り込む。従って、個々のカメラピクセルgは個別に放射された光線をどれだけ受光したかを示す異なった光輸送係数を有することになる。カメラピクセルアレイ19が4,194,304個の個別のカメラピクセルgを有する（すなわち4メガピクセルの解像度を有する）場合、個別のプロジェクタピクセルeはカメラピクセルアレイ19に関連付けられるために各々4,194,304個の個別光輸送係数の組を別々に必要とすることになる。従って、プロジェクタのピクセルアレイ17全体をカメラピクセルアレイ19に関連付け、実デジタルプロジェクタ11および実デジタルカメラ15との間にデュアリティ関係を確立するためには別個に判定した何百万組の光輸送係数（プロジェクタピクセルe毎に1組）が必要となる。

40

50

【0018】

本例において別々のプロジェクタピクセル e 各々は実デジタルカメラ 15 に関連付けられるために 4, 194, 304 個の個別的に判定された光輸送係数の別個の組を必要とし、実デジタルプロジェクタ 11 は何百万個の別々のプロジェクタピクセル e を有する可能性があるので、光輸送係数の各組を別々の光輸送係数のアレイとみなし、これらの別々のアレイを集めて 1 つの光輸送行列 T にまとめることが有益である。光輸送係数の各アレイは光輸送行列 T 内で別々の列を構成する。従って、光輸送行列 T における各列は別個のプロジェクタピクセル e に対応する 1 組の光輸送係数を構成する。

【0019】

本例において実デジタルプロジェクタ 11 は個別のプロジェクタピクセル e のアレイを有するデジタルプロジェクタで、実デジタルカメラ 15 は個別のカメラピクセル g のアレイを有するデジタルカメラであるので、実デジタルプロジェクタ 11 および実デジタルカメラ 15 の間のデュアリティ関係を定義するのに光輸送行列 T が用いられる。以下の考察において行列要素 T_{ge} は個別のプロジェクタピクセル e を個別のカメラピクセル g の関連付ける個別の光輸送係数（光輸送行列 T 内の）を特定する。

10

【0020】

実デジタルカメラ 15 に取り込まれた実画像はカメラピクセルアレイ 19 内の各カメラピクセル g により個別的に取り込まれたすべての光線からなる。従って、カメラピクセルアレイ 19 で判定された実取り込み画像を実画像取り込み行列 C に編成することが役に立つ。同様に、プロジェクタのピクセルアレイ 17 内における個々のプロジェクタピクセル e の活性化により構成される実映写画像を実画像映写行列 P に編成することが有用である。この記号を用い、実取り込み画像（実画像取り込み行列 C で定義された）は $C = T P$ という関係に従い光輸送行列 T により実映写画像（実画像映写行列 P ）に関連付けることができる。

20

【0021】

図 1 (a) のデュアリティ変換、すなわちデュアル構成、が図 1 (b) に示される。このデュアル構成において、図 1 (a) の実デジタルプロジェクタ 11 は仮想カメラ 11" に変換され、図 1 (a) の実デジタルカメラ 15 は仮想プロジェクタ 15" に変換される。仮想カメラ 11" および仮想プロジェクタ 15" は各々実デジタルプロジェクタ 11 および実デジタルカメラ 15 のデュアルな対応を表し、それら自体実装置でないことが理解されよう。つまり、仮想カメラ 11" は仮定上のカメラ（すなわち仮想カメラ 11"）が図 1 (a) の実プロジェクタが映写する実映像 13 に類似した仮定上の映像 13" を取り込むためにどのように挙動するかを数学的に表したものである。同様に、仮想プロジェクタ 15" は仮定上のプロジェクタ（すなわち仮想プロジェクタ 15"）が実デジタルカメラ 15 (図 1 (a) の) で取り込まれた実映像 13 に実質的に合致する仮定上のデュアル映像 13" を映写するためにどのように挙動するかを数学的に表したものである。このように図 1 (a) における実デジタルプロジェクタ 11 および実デジタルカメラ 15 の位置は図 1 (b) において仮想カメラ 11" および仮想プロジェクタ 15" に交換される。

30

【0022】

実装置のピクセル解像度は対応する仮想装置（すなわちデュアル装置）に持ち越されることが特記される。従って、仮想カメラ 11" は s 行および r 列からなる仮想光カメラピクセルアレイ 17" を有し、実デジタルプロジェクタ 11 のプロジェクタのピクセルアレイ 17 に合致している。同様に仮想プロジェクタ 15" は u 行および v 列からなる仮想プロジェクタのピクセルアレイ 19" を有し実デジタルカメラ 15 のカメラピクセルアレイ 19 に合致している。

40

【0023】

デュアル光輸送行列 T " がこのデュアル構成の光輸送行列で、デュアル画像取り込み行列 C " (仮想カメラ 11" で取り込まれるデュアル映像 13" を定義する) はデュアル画像映写行列 P " (仮想プロジェクタ 15" で映写されるデュアル映像 13" を定義する) に対し $C" = T" * P"$ という形で関係していると仮定すると、 $T"_{eg}$ が個別の仮想プロ

50

ジェクタピクセル g ”を個別の仮想カメラピクセル e ”に関連付ける個別のデュアル光輸送係数となる。

【0024】

ヘルムホルツの相反定理はピクセルからピクセルへの輸送係数は双方向において（すなわち実デジタルプロジェクタ11から実デジタルカメラ15へ、および仮想プロジェクタ15”から仮想カメラ11”へ）等しいと規定している。すなわち $T_{eg} = T_{ge}$ で、これは $T = T^T$ を意味する（すなわちデュアル光輸送行列 T ”は光輸送行列 T に対する数学的行列転置の結果に等しい）。従って、光輸送行列 T が与えられると、デュアル光輸送行列 T^T を用いてデュアル構成において得られるデュアル、または仮想、画像を合成することができる。

10

【0025】

このように、光輸送行列 T はプロジェクタにより取り込まれたように見える画像を作り出すことを可能にし、カメラは第2プロジェクタの役割を果たす。しかし、上記に説明するように、光輸送行列 T を生成し操作することの高度な複雑さのため今までその適用は、特にプロジェクタ-カメラシステムの較正分野において、非常に限定されていた。

【0026】

プロジェクタ-カメラシステムに関する他の問題はプロジェクタの視線を遮る可能性のある光拡散物体をどのように補正するかである。関連するものとしてプロジェクタ-カメラシステムが典型より複雑な画像を達成するのに用いることができるかという問題である。例えば、このようなシステムは複数のプロジェクタによる複数の画像を組み合わせて単一の合成画像を作成することができるであろうか。あるいは、「3-D」画像または以前はより複雑な機器およびより複雑な映写設備を必要とした他の視覚上の効果を生成することができるであろうか。さらに、プロジェクタ-カメラシステムのカメラをより有効に活用し、カメラが画像作成プロセスの能動的な部分になることができるであろうか。さらに、このようにプロジェクタ-カメラシステムにおいて低解像度で安価なカメラを用いることはどのような影響があるであろうか。

20

【0027】

以前の研究 [Raskar 他 1998; Underkoffler および Ishii 1998] はインテリジェント照明の概念を提示し、プロジェクタを用いて仕事場のやりとりを向上させ問題解決の新規ツールの役割を果たせることを示している。以来プロジェクタ-カメラの世界はインテリジェント・プロジェクタに関する多くの技術的問題を解決している。特に複数プロジェクタの自動モザイク化において相当な進歩がなされている [Chen 他 2002; Yang 他 2001; Raij 他 2003; Sukthankar 他 2001; Raskar 他 2003]。

30

【0028】

[Raskar 他 2001] は複雑な物体への映写を実証している。前以て作成された物体の3Dモデルを用い、複数のプロジェクタで些細でない複雑な形状を有する実際の物理的物体に仮想テクスチャおよびアニメーションを加えることができる。

【0029】

[Fujii 他 2005] は同軸プロジェクタ-カメラシステムを用いてリアルタイムで物体の外観を改変する方法を提案した。[Grossberg 他 2004] は区分的な多項式3Dモデルを組み入れて非同軸プロジェクタ-カメラシステムが光景の映写を行えるようにした。

40

【0030】

プロジェクタ-カメラシステムは深度マップを抽出するためにも用いられ [Zhang および Nayar 2006]、深度エッジを回復する手段として時空多重化照明が提案されている [Raskar 他 2004]。

【0031】

【非特許文献1】Zhengyou Zhang著の論文「適応性を有するカメラ較正用の新規手法」(「A flexible new technique for camera calibration」)、IEEEパターン分析および人工知能紀要 (Transactions on Pattern Analysis and Machine Intelligence)、22(11): 1330 ~ 1334、2000年

50

【発明の開示】**【発明が解決しようとする課題】****【0032】**

詳細に後述されるように、本発明は望ましい画像を作成するために何をプロジェクタが映写する必要があるかをどのように判定するかという問題に、光輸送行列の逆を用いて対応しており、その適用はさらにプロジェクタ-カメラシステムの較正を簡素化する。

【0033】

本発明のさらなる目的は平坦でない場面（すなわち映写画面）に投影される画像の歪み補正を動く物体に投影される画像の歪み補正に広げることである。

【0034】

本発明のさらなる目的は消費者レベルの演算装置における、および特に並列処理システムにおける使用を促進できるよう較正データの編成をどのように最適化するかである。

10

【課題を解決するための手段】**【0035】**

このような目的は以下の本発明により解決される。

【0036】

本発明の異なるデータ間の相互関係を物理的メモリ内で配列する方法は、プロジェクタをカメラに関連付ける較正情報で定義される異なるデータ間の相互関係を物理的メモリ内で配列する方法であって、(a)プロジェクタのピクセルバッファ、カメラピクセルの座標バッファ、およびカメラピクセルのRGBバッファを定義するステップと、(b)前記プロジェクタの各プロジェクタピクセルについて前記較正情報に従い、前記プロジェクタピクセルに対し較正関係を有するカメラピクセルを関連カメラピクセルグループに収集するステップと、(c)前記カメラピクセルの座標バッファ内で、前記カメラ内の各カメラピクセルのX-Y座標位置情報セットを維持することで、前記X-Y座標位置情報セットは前記関連カメラピクセルグループに対応する座標グループに配列され、各前記座標グループ内におけるX-Y座標位置情報セットは前記カメラピクセルの座標バッファ内で連続的に配列されるステップと、(d)前記カメラピクセルのRGBバッファ内で、前記較正情報で判定される各カメラピクセルの較正RGB値を維持することで、前記較正RGB値は前記関連カメラピクセルグループに対応するRGBグループに配列され、各RGBグループ内における較正RGB値セットは前記カメラピクセルの座標バッファ内におけるX-Y座標位置情報のセットの配列を映すような形で前記カメラピクセルのRGBバッファ内において連続的に配列されるステップと、(e)前記プロジェクタピクセルのバッファ内で、前記プロジェクタのプロジェクタピクセル毎に別個のプロジェクタのピクセルデータセットを定義することで、前記プロジェクタのピクセルデータセットは各プロジェクタのピクセルデータセットと前記プロジェクタからそれに相關するプロジェクタピクセルとの間で識別可能な1対1の相關を維持するように前記プロジェクタ内のプロジェクタピクセル配列に相關する形で配列されるステップと、を有し、前記ステップ(e)において、前記プロジェクタのピクセルデータセットは前記ステップ(b)で定義された対応する前記関連カメラピクセルグループ内のカメラピクセル数を示す関連カメラピクセル総数を含み、前記プロジェクタのピクセルデータセットはさらに対応する関連カメラピクセルグループ内の第1カメラピクセルの、前記カメラピクセルの座標バッファ内における格納位置を特定する位置インジケータを含むことを特徴とする。

20

【0037】

また、本発明の異なるデータ間の相互関係を物理的メモリ内で配列する方法において、前記ステップ(e)において、前記プロジェクタのピクセルデータセットは、前記プロジェクタ内におけるプロジェクタピクセルの連鎖配列に相關した連鎖の形で配列されることを特徴とする。

30

【0038】

また、本発明の異なるデータ間の相互関係を物理的メモリ内で配列する方法において、特定のプロジェクタピクセルに関連する任意のカメラピクセルのX-Y座標位置情報セッ

40

50

トは、前記特定のプロジェクトピクセルに対応する関連カメラピクセルグループから望まれるカメラピクセルを選択するステップと、前記プロジェクトピクセルのバッファ内で前記特定のプロジェクトピクセルに相關するプロジェクトのピクセルデータセットを特定するステップと、前記特定されたプロジェクトのピクセルデータセット内の位置インジケータをアクセスするステップと、望まれるカメラピクセルの関連カメラピクセルグループ内における連鎖位置の場所で判定されたオフセット分だけ前記アクセスされた位置インジケータを増分することにより目標アドレスを生成するステップと、前記目標アドレスにおけるカメラピクセルの座標バッファ内に格納されるX-Y座標位置情報セットにアクセスするステップと、により得られることを特徴とする。

【0039】

また、本発明の異なるデータ間の相互関係を物理的メモリ内で配列する方法において、前記望まれるカメラピクセルの較正RGB値は、前記目標アドレスにおける前記カメラピクセルのRGBバッファ内に格納される較正RGB値をアクセスすることにより得られることを特徴とする。

【0040】

また、本発明の異なるデータ間の相互関係を物理的メモリ内で配列する方法において、前記カメラピクセルの座標バッファにおけるアドレス増分はX-Y座標位置情報セットにより決定され、前記カメラピクセルのRGBバッファにおけるアドレス増分は較正RGB値により決定され、前記各カメラピクセルの較正RGB値のメモリ格納は各X-Y座標位置情報セットとは異なったバイト数を必要とすることを特徴とする。

【0041】

また、本発明の異なるデータ間の相互関係を物理的メモリ内で配列する方法において、複数の量子化数値レベルを定義するステップと、前記関連カメラピクセルグループ各々内のカメラピクセル数に従い前記関連カメラピクセルグループを対応量子化数値レベルに収集するステップとをさらに有することを特徴とする。

【0042】

また、本発明の異なるデータ間の相互関係を物理的メモリ内で配列する方法において、個別の量子化数値レベル内における関連カメラピクセルグループの収集をマルチコア処理システムの異なった処理コアに対し使用可能にするステップをさらに有することを特徴とする。

【0043】

また、本発明の異なるデータ間の相互関係を物理的メモリ内で配列する方法において、前記複数の量子化数値レベルを定義する前に、前記関連カメラピクセルグループの最小および最大のものを特定することによって最小から最大までの幅を定義するステップと、前記量子化数値レベルを前記最小から最大までの幅の範囲内になるよう定義するステップとをさらに有することを特徴とする。

【0044】

また、本発明の異なるデータ間の相互関係を物理的メモリ内で配列する方法において、前記量子化レベルは前記最小から最大までの幅の範囲内の区分として目標関連カメラピクセルグループを選択することによりさらに定義し、前記量子化レベルは前記目標関連カメラピクセルグループ内のカメラピクセル数に等しく設定されることを特徴とする。

【0045】

また、本発明の異なるデータ間の相互関係を物理的メモリ内で配列する方法において、各量子化レベル内において、量子化レベルより少ないカメラピクセルを有する関連カメラピクセルグループはRGB値がゼロであるダミーカメラピクセルで嵩上げされ各々のカメラピクセルの合計数をその量子化レベルに引き上げることを特徴とする。

【0046】

また、本発明の異なるデータ間の相互関係を物理的メモリ内で配列する方法において、前記各量子化レベル内において、量子化レベルは前記ダミーカメラピクセルによる嵩上げ前の最大関連カメラピクセルグループに等しく設定されることを特徴とする。

10

20

30

40

50

【0047】

また、本発明の異なるデータ間の相互関係を物理的メモリ内で配列する方法において、第1バイトサイズの第1データ単位を有する第1メモリを提供するステップをさらに有し、前記カメラピクセルのRGBバッファは、前記第1バイトサイズより小さいバイトサイズを有する第2データ単位を有し、前記カメラピクセルのRGBバッファは、前記第1メモリから第1データ単位で埋められ、前記カメラピクセルのRGBバッファからのデータは前記第2データ単位でアクセスされることを特徴とする。

【0048】

また、本発明の異なるデータ間の相互関係を物理的メモリ内で配列する方法において、前記第1データ単位は前記第2データ単位より16倍大きいことを特徴とする。

10

【0049】

また、本発明の異なるデータ間の相互関係を物理的メモリ内で配列する方法において、各々偶数のバイトで定義される第1データ単位を有する第1メモリを提供するステップと、前記第1メモリを奇数バイトのデータで埋めるステップと、前記第1メモリ内における前記第1データ単位の1つ内にある前記奇数バイトのデータの選択された1つを、前記奇数バイトデータ内のバイト数にオフセット値を掛け、結果を前記第1データ単位内のバイト数で割ることによりアクセスするステップとをさらに有することを特徴とする。

【0050】

また、本発明の異なるデータ間の相互関係を物理的メモリ内で配列する方法において、前記オフセット値は開始点から前記奇数バイトのデータの選択された1つまで増分する必要のある奇数バイトのデータの数となるよう選択されることを特徴とする。

20

【0051】

また、本発明の異なるデータ間の相互関係を物理的メモリ内で配列する方法において、前記奇数バイトのデータは3バイトのRGBデータであり、前記カメラピクセルのRGBバッファは前記第1メモリからの奇数バイトのデータ単位で埋められることを特徴とする。

【0052】

また、本発明の異なるデータ間の相互関係を物理的メモリ内で配列する方法において、前記較正情報は以下のステップにより定義されるビュー投影行列を有することを特徴とする。

30

(1) 前記プロジェクタを前記カメラに関連付ける光輸送行列Tを得るステップと、(2) 前記光輸送行列Tにおける各行について前記光輸送行列の共通行に沿って行列入力を比較し、共通行における最高値の行列入力を留保し前記共通行における残りの入力にゼロ値を割り当て、改変光輸送行列T*を作成するステップと、(3) 前記改変光輸送行列T*における各列を順に目標列として特定し、各目標列について前記目標列における非ゼロ値に対し前記目標列に關し正規化値を計算するステップと、(4) 前記改変光輸送行列T*と同じサイズの中間行列を作成するステップと、(5) 前記中間行列の各列を前記改変光輸送行列T*において対応する目標列の計算された正規化値で埋めることで、前記中間行列において埋められた各列の各正規化値は前記改変光輸送行列T*において対応する列の前記非ゼロの入力値と1対1の対応を維持するステップと、(6) 前記中間行列に転置行列演算を適用してビュー投影行列を作成するステップ。

40

【0053】

ここで、本発明を要約すると以下のようにすることができる。

【0054】

まず発明の概要として、前述の本発明の目的は光輸送行列Tの生成を簡素化し、デュアル・フォトグラフィにおける光輸送係数の実施を簡素化し、光輸送行列Tをさらに簡素化しながら改変して光拡散効果または光ノイズに対する補正を組み入れたシステムにより満足される。

【0055】

デュアル・フォトグラフィの適用は(p × q)のプロジェクタのピクセルアレイの光輸

50

送行列 T を生成するのに必要な取り込み画像数を $(p \times q)$ の画像から $(p + q)$ の画像に削減することにより簡素化される。光輸送行列 T の操作は、完全に埋められた光輸送行列 T の使用を非ゼロの光輸送値にのみ結び付けるインデクスで置き換えることにより、簡素化される。ゼロ値の光輸送係数の使用を排除することにより、デュアル・フォトグラフィを実施するためのメモリおよび処理要件が非常に低減される。このデュアル・フォトグラフィ手法がプロジェクタ-カメラシステムの較正に適用される。

【0056】

本発明の第1実施形態で、デジタルプロジェクタをデジタルカメラに関連付ける光輸送係数を生成する方法が示される。デジタルプロジェクタが映写ピクセルのアレイを有し、デジタルカメラがセンサピクセルのアレイを有する場合、方法は、プロジェクタ内の映写ピクセルの第1グループを同時に活性化し映写場面上に第1テストパターンを投影することで、前記第1テストパターンに入っていない映写ピクセルは暗く維持されること、映写場面上の第1テストパターンの第1画像を取り込むこと、プロジェクタ内の映写ピクセルの第2グループを同時に活性化し映写場面上に2テストパターンを映写することで、前記第2テストパターンに入っていない残りの映写ピクセルは暗く維持され、映写ピクセルの第1グループおよび第2グループは目標映写ピクセルを定義する1つのみの共通映写ピクセルを有すること、前記映写場面上の前記第2テストパターンの第2画像を取り込むこと、第1画像の画像ピクセルを第2画像の対応画像ピクセルと比較し、2つの比較された画像ピクセルの暗い方を留保することで、留保された画像ピクセルは合成画像を形成すること、および合成画像において暗くないすべての画像ピクセルを特定することで、暗くない画像ピクセルは目標映写ピクセルに結び付いた非ゼロの光輸送係数を定義すること、を含む。

10

20

20

【0057】

あるいは、光輸送係数を生成する方法はさらに選択された数の目標映写ピクセルに対する光輸送係数を特定すること、選択された数の目標映写ピクセルを対応して結び付いた非ゼロの光輸送係数に結び付けるインデクスを生成すること、および非ゼロの光輸送係数のみを格納すること、を含むことができる。

30

【0058】

光輸送行列は全行列形式または光輸送係数入力の第1アレイと、第1アレイのどの光輸送係数が光輸送行列のどの列に対応するかを示す記録を維持する第2アレイからなる簡略化された2アレイ形式のいずれかで書くことができるので、光ノイズおよび光拡散効果を補正する改変プロセスも全行列形式またはまたは2アレイ形式のいずれかに適用することができる。いずれの場合もプロセスは表示制約を課すことを含み、課さなければ適用することができない。

40

【0059】

従って、最初の工程は任意の場面においてプロジェクタ-カメラシステムの光輸送行列 T にシミュレートされた表示制約を課すことで、表示制約はプロジェクタから放射された光線は区別可能な部分でカメラの画像センサに当たることを規定する。これは次の方法で行うことができる。光輸送行列 T の各列について、光輸送行列の共通行に沿って行列の入力を比較し、共通行における最高値の行列入力を留保する。光輸送アレイに前記光輸送行列 T の各行からの最高値の行列入力を入れる。光輸送アレイの各入力が光輸送行列のどの列から由来するかの記録を維持する。必要に応じ記録および光輸送アレイを用い、プロジェクタの各プロジェクタピクセルについて光ファットプリント情報を抽出する。この方法では、任意の場面において前記プロジェクタから前記カメラへの光のパスに光拡散物体が含まれても構わず、前記方法は光拡散効果を補正するのに有効である。

【0060】

さらにこの方法において、光輸送行列 T の各列はプロジェクタの単一のプロジェクタピクセルを活性化することにより実現されカメラの画像センサ全体で取り込まれた画像に対応し、各光ファットプリント情報はそれに対応するプロジェクタピクセルの光輸送値を構成する。

50

【0061】

光輸送アレイおよび記録は前記光輸送行列 T の代わりに使用できる改変光輸送行列をなし、改変光輸送行列はデュアル・フォトグラフィでの使用に適していることがさらに特記される。

【0062】

上述の方法は次の工程により光輸送行列の推定逆行列を生成するのに用いることができる。正規化光フットプリント情報を各々のプロジェクタピクセルに対応するグループで計算する。中間アレイを作成し、その中間アレイに対応する光フットプリント情報の計算された正規化値を入れる。中間アレイの各入力を対応する正規化光フットプリントに結び付けるために中間アレイ用の中間記録を維持する。中間アレイおよび中間記録を中間行列の表記として解釈し、前記中間アレイに転置行列演算を適用する。

10

【0063】

この正規化光フットプリント情報を各々のプロジェクタピクセルに対応するグループで計算するプロセスは1つの光フットプリントを構成する光輸送アレイのグループの二乗の総計を生成し、光輸送アレイ入力の各入力を総計で割ることからなることもさらに特記される。

【0064】

任意の場面においてプロジェクタ - カメラシステムの光輸送行列 T にシミュレートされた表示制約を課す別の方法では次の方法に従うことができる。光輸送行列 T の各行について、光輸送行列 T の共通行に沿った行列入力を比較し、共通行における最高値の行列入力以外をすべて無効化する。結果として得られた行列は改変光輸送行列 T^* となる。この方法で、行列入力はゼロ値を割り当てられることにより無効化される。前と同じように、この方法は表示制約が真にサポートされていないものも含みどのような任意の場面にでも用いることができる。従って、任意の場面が前記プロジェクタから前記カメラへの光パスの間に光を拡散する物体が含まれていても問題ない。改変光輸送行列は相変わらず光拡散効果を補正するのに有效である。改変光輸送行列 T^* は相変わらずデュアル・フォトグラフィでの使用に適していることはさらに有用である。

20

【0065】

前と同様、改変光輸送行列 T^* の推定逆行列は、改変光輸送行列 T^* における各列を順に目標列として特定し、前記目標列に関し目標列の無効化されていない入力値の正規化値を計算すること、前記改変光輸送行列 T^* と等しいサイズの中間行列を作成すること、改変光輸送行列における対応目標列の計算された正規化値を中間行列の各列に入れることで、中間行列の埋められた各列における各正規化値は改変光輸送行列 T^* の対応列における無効化されていない入力値と1対1の対応を維持している、および転置行列演算を前記中間行列に適用することにより生成することができる。

30

【0066】

この方法で、改変光輸送行列の対応目標列における無効化されていない入力値の正規化値が入っていない中間行列への値の入力はゼロ値が入れられる。さらに、目標列に関し目標列の無効化されていない入力値の正規化値を計算するプロセスは目標列の無効化されていない入力値のみの二乗の総計を生成し、目標値の無効化値をすべて無視し、無効化されていない各値を総計で割ることからなる。

40

【0067】

別の言い方をすると、中間行列を行列 T とし、改変光輸送行列 T^* の目標列を列 T^*r とし、行列 T の対応する列を列 T^*r とすると、行列 T の構成と入力は $r = T^*r / (T^*r)^2$ と定義される。

【0068】

本発明の別の実施形態はプロジェクタをカメラに関連付ける較正情報で定義される異なったデータ（すなわち較正データ）間の相互関係を物理的メモリ内で配列する方法である。方法は次の工程を有することができる。（a）プロジェクタのピクセルバッファ、カメラピクセル座標バッファ、およびカメラピクセルのRGBバッファを定義すること、（b）

50

) プロジェクタの各プロジェクタピクセルについて較正情報に従い、カメラのプロジェクタピクセルに対し較正関係を有するカメラピクセルを関連カメラピクセルグループに収集すること、(c) カメラピクセルの座標バッファ内で、カメラ内の各カメラピクセルの X - Y 座標位置情報セットを維持することで、X - Y 座標位置情報セットは関連カメラピクセルグループに対応する座標グループに配列され、各座標グループ内における X - Y 座標位置情報セットはカメラピクセルの座標バッファ内で連續的に配列されること、(d) カメラピクセルの R G B バッファ内で、較正情報で判定される各カメラピクセルの較正 R G B 値を維持することで、較正 R G B 値は関連カメラピクセルグループに対応する R G B グループに配列され、各 R G B グループ内における較正 R G B 値セットはカメラピクセルの座標バッファ内における X - Y 座標位置情報のセットの配列を映すような形でカメラピクセルの R G B バッファ内において連續的に配列されること、および(e) プロジェクタピクセルのバッファ内で、プロジェクタのプロジェクタピクセル毎に別個のプロジェクタのピクセルデータセットを定義することで、プロジェクタのピクセルデータセットは各プロジェクタのピクセルデータセットとプロジェクタからそれに相關するプロジェクタピクセルとの間で識別可能な 1 対 1 の相關を維持するようにプロジェクタ内のプロジェクタピクセル配列に相關する形で配列され、プロジェクタのピクセルデータセットは工程(b)で定義された対応関連カメラピクセルグループ内のカメラピクセル数を示す関連カメラピクセル総数を含み、プロジェクタのピクセルデータセットはさらに対応関連カメラピクセルグループ内の第 1 カメラピクセルの、カメラピクセルの座標バッファ内における格納位置を特定する位置インジケータを含むこと。

10

20

30

40

50

【0069】

本方法の工程(e)内で、プロジェクタのピクセルデータセットは前記プロジェクタ内におけるプロジェクタピクセルの連鎖配列に相關した連鎖の形で配列される。

【0070】

本方法において、特定のプロジェクタピクセルに関連する任意のカメラピクセルの X - Y 座標位置情報セットは特定のプロジェクタピクセルに対応する関連カメラピクセルグループから望まれるカメラピクセルを選択し、プロジェクタピクセルのバッファ内で特定のプロジェクタピクセルに相關するプロジェクタのピクセルデータセットを特定し、特定されたプロジェクタのピクセルデータセット内の位置インジケータをアクセスし、望まれるカメラピクセルの関連カメラピクセルグループ内における連鎖位置の場所で判定されるオフセット分アクセスされた位置インジケータを増分することにより目標アドレスを生成し、目標アドレスにおけるカメラピクセルの座標バッファ内に格納される X - Y 座標位置情報セットをアクセスすることにより得られる。さらに、望まれるカメラピクセルの較正 R G B 値は目標アドレスにおけるカメラピクセルの R G B バッファ内に格納される較正 R G B 値をアクセスすることにより得られる。

【0071】

さらに本方法において、カメラピクセルの座標バッファにおけるアドレス増分は X - Y 座標位置情報セットにより決定され、カメラピクセルの R G B バッファにおけるアドレス増分は較正 R G B 値により決定され、各カメラピクセルの較正 R G B 値のメモリ格納は各 X - Y 座標位置情報セットとは異なったバイト数を必要とする。

【0072】

本方法はさらに、複数の量子化数値レベルを定義すること、および各関連カメラピクセルグループ内のカメラピクセル数に従い関連カメラピクセルグループを対応量子化数値レベルに収集すること、を含むことができる。さらに、個別の量子化数値レベル内における関連カメラピクセルグループの収集をマルチコア処理システムの異なった処理コアに対し使用可能にすることができる。加えて、複数の量子化数値レベルを定義する前に関連カメラピクセルグループの最小および最大のものを特定し、量子化数値レベルを最小から最大の範囲内になるよう定義することによりすることにより最小から最大の範囲を定義することができる。量子化レベルは最小から最大の範囲内の区分として目標関連カメラピクセルグループを選択することによりさらに定義することができ、量子化レベルは目標関連カメ

ラピクセルグループ内のカメラピクセル数に等しく設定される。

【0073】

加えて各量子化レベル内において、量子化レベルより少ないカメラピクセルを有する関連カメラピクセルグループはRGB値がゼロであるダミーカメラピクセルで嵩上げされ各々のカメラピクセルの合計数をその量子化レベルに引き上げる。このように、各量子化レベル内において、量子化レベルはダミーカメラピクセルによる嵩上げ前の最大関連カメラピクセルグループに等しく設定される。

【0074】

本方法はさらに第1バイトサイズの第1データ単位を有する第1メモリを提供することを含むことができ、カメラピクセルのRGBバッファは第1バイトサイズより小さいバイトサイズを有する第2データ単位を有し、カメラピクセルのRGBバッファは第1メモリからのデータ単位で埋められ、カメラピクセルのRGBバッファからのデータは第2データ単位でアクセスされる。この場合第1データ単位は第2データ単位より16倍大きいことがあり得る。

10

【0075】

上述の方法はさらに、各々偶数のバイトで定義される第1データ単位を有する第1メモリを提供すること、第1メモリを奇数バイトのデータで埋めること、第1メモリ内における第1データ単位の1つ内にある奇数バイトのデータの選択された1つを、奇数バイトデータ内のバイト数にオフセット値を掛け、結果を第1データ単位内のバイト数で割ることによりアクセスすること、を含むことができる。この場合、オフセット値は開始点から前記奇数バイトのデータの選択された1つまで増分する必要のある奇数バイトのデータの数となるよう選択することができる。加えて奇数バイトのデータは3バイトのRGBデータであることができ、カメラピクセルのRGBバッファは第1メモリからの奇数バイトのデータ単位で埋められることがある。

20

【発明を実施するための最良の形態】

【0076】

添付図面と併せて以下の説明およびクレームを参照することにより他の目的および成果と共に発明のより深い理解が明らかになろう。

【0077】

カメラ較正手法をプロジェクタに適用するためには、プロジェクタが画像を取り込めなければならない。つまり、プロジェクタが画像を取り込むことができたら、プロジェクタ-カメラシステムはマルチカメラシステムのように扱うことができ、プロジェクタ-カメラシステムを較正する標準的なカメラ較正手法（上記に説明）を用いることができるかもしれない。言い換えれば、プロジェクタを擬似カメラとして扱うことができれば、上述のマルチカメラシステムのカメラ較正段階に類似した方法で実カメラとともに較正することができ、以前にプロジェクタ-カメラシステムを較正するために用いられた「ブーストストラップ的」プロジェクタ較正段階を排除できるかもしれない。

30

【0078】

図2(a)に関し、本発明に従った撮像構成は実プロジェクタ21および実カメラ25を含み得る。実プロジェクタ21はデジタルプロジェクタであることが好ましく、p行およびq列の個別の撮像映写要素（すなわちプロジェクタピクセル j ）からなる撮像映写アレイ（すなわちプロジェクタピクセルアレイ27）を含む撮像要素を有する。プロジェクタピクセルアレイ27は実プロジェクタ21の内部にあり、説明の便宜上図2(a)において点線の正方形内の十字線で示される。実プロジェクタ21は液晶表示（LCD）型、デジタル光処理（DPL（登録商標））型、反射型液晶パネル（Liquid Crystal On Silicon = LCO S）、または他のデジタル映写技術型であることが好ましい。

40

【0079】

実カメラ25はm行およびn列の個別のカメラピクセル i （すなわち画像センサ要素または受光ピクセル）からなる実カメラピクセルアレイ29（すなわち受光アレイまたは画像センサアレイ）を含む画像センサを有するデジタルカメラであることが好ましい。簡単

50

にするため、実カメラピクセルアレイ 2 9 は実カメラ 2 5 上に示されるが、実カメラピクセルアレイ 2 9 は実カメラ 2 5 の内部にあることが理解されよう。

【 0 0 8 0 】

実プロジェクト 2 1 および実カメラ 2 5 を用いたこの物理的構成は「根本」構成と呼ばれることが好ましい。実プロジェクト 2 1 内の個別のプロジェクトピクセル j から発射された光線は不規則なまたは平面の形状を有し得る映写表面（すなわち表示環境または場面）から反射されることにより実映像 2 3 を形成し、光線の一部は最終的に実カメラ 2 5 内の画像センサに到達する。一般的に各光線は場面で分散、反射、および屈折し、実カメラピクセルアレイ 2 9 にわたりいくつか異なった位置でカメラの画像センサに当たる。このように、実プロジェクト 2 1 の撮像されたプロジェクトピクセル j から放射された 1 つの光線が実カメラ 2 5 に到達すると、個々に投射された光線は $m \times n$ の画像を実カメラピクセルアレイ 2 9 上で形成し、個々のカメラピクセル i は投射光線から一定量の光を受ける。

10

【 0 0 8 1 】

従って、理想的には 1 つのプロジェクトピクセル j をオンにして 1 つの光線を放射させ 1 つのカメラピクセル i に当たるようにして 1 つのプロジェクトピクセル j を 1 つのカメラピクセル i に関連付ける 1 つの光輸送係数を判定することができるよう見えるかもしれないが、実際にはそうでない。実際には実カメラピクセルアレイ 2 9 全体が放射された 1 つの光線からの光強度の異なった寄与を受ける。従って、個々のプロジェクトピクセル j 各々から放射される各光線は実カメラピクセルアレイ 2 9 内のカメラピクセル i 各々に對し 1 つの異なった組、またはアレイ、の個別の光輸送係数を生成する。従って、各組（すなわちアレイ）は各カメラピクセル i に對し 1 つとなる ($m \times n$) [すなわち m 掛ける n] の個別の光輸送係数からなることになる。

20

【 0 0 8 2 】

光輸送係数の各組を係数の列として配列し複合された光輸送行列 T を形成すると、複合された光輸送行列 T は個別のプロジェクトピクセル j 各々に對し異なった光輸送係数の列を有することになる。さらに、各列内の各光輸送係数入力（すなわち行列要素）と各カメラピクセルとの間には 1 対 1 の対応があるので、各列は 1 つのプロジェクトピクセル j をオンにした結果、実カメラ 2 5 で取り込まれた画像全体を表す。従って、光輸送行列 T 全体（すなわち複合の）は ($p \times q$) [すなわち p 掛ける q] 列（オンにされた各個別のプロジェクトピクセル j につき 1 列 [すなわち取り込み画像] ）および ($m \times n$) 行（各個別のカメラピクセル i につき 1 行）からなる。

30

【 0 0 8 3 】

以下の考察において実投影画像を、($p \times q$) 要素（各プロジェクトピクセル j につき 1 つ）を有する実取り込みベクトルとしての第 1 ベクトル $R_{p\ r\ j\ c\ t}$ （すなわち実投影ベクトル）として、また結果としてもたらされる実取り込み画像を、($m \times n$) 要素（各カメラピクセル i につき 1 つ）を有する実取り込みベクトルとしての第 2 ベクトル $R_{c\ p\ t\ r}$ （すなわち実取り込みベクトル）として見ることが役立つ。

【 0 0 8 4 】

実投影画像（すなわちプロジェクトピクセルアレイ 2 7 全体を用いて投影した画像）は ($p \times q$) の実取り込みベクトルとしての第 1 ベクトル $R_{p\ r\ j\ c\ t}$ （すなわち「 p 掛ける q ベクトル」と表される記号、および対応する実取り込み画像（すなわち実カメラピクセルアレイ 2 9 全体を用いて取り込まれた画像）は ($m \times n$) の実取り込みベクトル（すなわち「 m 掛ける n ベクトル」）と表される記号を用いると、実プロジェクト 2 1 と実カメラ 2 5 との間の光輸送関係は、

40

【 0 0 8 5 】

【 数 1 】

$$R_{cptr} = T * R_{prjct}$$

50

と書くことができ、光輸送行列 T は実プロジェクト 2 1 を実カメラ 2 5 に関連付ける複合光輸送行列である。光輸送行列 T は例えば個々かつ別々に各プロジェクトピクセル j をオンにして各個別的なカメラピクセル i の対応光輸送係数を判定することにより前以て生成されていることが理解されよう。

【0086】

要約すると、各プロジェクトピクセル j は実カメラピクセルアレイ 2 9 全体にわたり拡散される光線をもたらすので、各個別のカメラピクセルは各個別のプロジェクトピクセル j から受けた所定の光特性の強度値を示す異なった値の光輸送係数を有する。本実施形態で、この光特性は光強度の尺度であることが好ましい。従って、各プロジェクトピクセル j は $(m \times n)$ の個別的な光輸送係数からなる列をもたらし、各係数は各カメラピクセル i が受けた光強度の量を示す。実プロジェクト 2 1 は $(p \times q)$ のプロジェクトピクセル j を有するので、光輸送行列 T は $(p \times q)$ 列 [各プロジェクトピクセル j につき 1 列] および $(m \times n)$ 行 [各カメラピクセル i につき 1 行] を有する。従って、光輸送行列 T は従来非常に大きく、 $(p \times q \times m \times n)$ の個別的な光輸送係数からなっていた。

10

【0087】

図 2 (b) に関し、「デュアル」な構成において実プロジェクト 2 1 はプロジェクトピクセルアレイ 2 7 とサイズが等しい仮想カメラピクセルアレイ 2 7 ”を有する仮想カメラ 2 1 ”で置き換えられる。このように仮想カメラ 2 1 ”は p 行掛ける q 列の仮想カメラピクセル j ”からなる仮想カメラピクセルアレイ 2 7 ”を有する。同様にこの「デュアル」な構成において、実カメラ 2 5 は実カメラピクセルアレイ 2 9 とサイズが等しい仮想プロジェクトピクセルアレイ 2 9 ”を有する仮想プロジェクト 2 5 ”で置き換えられる。このように仮想プロジェクト 2 5 ”は m 行掛ける n 列の仮想プロジェクトピクセル i ”からなる仮想プロジェクトピクセルアレイ 2 9 ”を有する。

20

【0088】

この場合、仮想映像 2 3 ” (仮想プロジェクト 2 5 ”により投影される) は $(m \times n)$ の仮想投影ベクトル $V_{p\ r\ j\ c\ t}$ ”で表すことができる。同様に、仮想カメラ 2 1 ”により取り込まれた仮想取り込み画像は $(p \times q)$ 仮想取り込みベクトル $V_{c\ p\ t\ r}$ ”で表すことができる。従って、このデュアル構成において仮想取り込みベクトル $V_{c\ p\ t\ r}$ ”はデュアル光輸送行列 T ”により仮想投影ベクトル $V_{p\ r\ j\ c\ t}$ ”に関連付けられる。ヘルムホルツの相反性の原理により、光輸送は双方向において等しい。従って、デュアル構成 (すなわちデュアリティ変換構成) に対するデュアル光輸送行列 T ”は次の関係で仮想取り込みベクトル $V_{c\ p\ t\ r}$ ”を仮想投影ベクトル $V_{p\ r\ j\ c\ t}$ ”に関連づける。

30

【0089】

【数 2】

$$V_{cptr}'' = T'' * V_{prjct}''$$

【0090】

しかしこの関係はデュアル光輸送行列 T ”を構成し、仮想的に投影された画像を仮想的に取り込まれた画像に関連付ける大量の「仮想」光輸送係数をどのように判定できるかという問題を提議する。注目すべきは、光輸送行列 T からそのデュアル光輸送行列 T ”へのデュアリティ変換は実光輸送行列に対し転置行列演算を行うことにより達成される。従って、デュアル光輸送行列 T ”は光輸送行列 T の転置で、 $T'' = T^T$ および $V_{c\ p\ t\ r}'' = T^T * V_{p\ r\ j\ c\ t}$ ”となる。以降、 T^T をデュアル光輸送行列という。

40

【0091】

行列計算方法で周知のように、一般的な $[x \times y]$ 行列 A の転置行列演算は上付き文字「 T 」を加えることにより示され (例えば A^T のように) 、第 1 列が行列 A の第 1 行、第 2 列は行列 A の第 2 行、第 3 列は行列 A の第 3 行、以下同様、である $[y \times x]$ の行列で定義される。容易に明らかのように、この行列演算は単に第 1 要素を中心として元の行列 A を反転させ、その第 1 要素 (すなわち位置 (1, 1) において) は変化せず、第 1 列の下端が第 1 行の終りになるようにする。従って、根本構成に対する光輸送行列 T を獲得、

50

あるいは別途判定できれば、デュアル構成に対するデュアル光輸送行列 T^T は上述のように光輸送行列 T を反転させてその転置されたデュアル光輸送行列 T^T を得ることにより容易に計算できる。以下のさらなる考察において、デュアル光輸送行列および転置されたデュアル光輸送行列 T^T は互換的に用いることができる。

【0092】

しかし、光輸送行列 T を判定するのに簡略化された方法を提供できれば、デュアリティ変換については光輸送行列 T から直接にデュアル光輸送行列 T^T を判定できることが特記される。光輸送行列 T の判定を簡略化する方法を提案する前に、まず光輸送行列 T を判定するための従来の方法の難しさをいくつか考察することが有益である。

【0093】

上記で説明されたように、光輸送行列 T は実プロジェクト 21 の各個別のプロジェクトピクセル j と実カメラの個別のカメラピクセル i すべてとの間で対応する個別の光輸送係数を留保している。従って、個別のプロジェクトピクセル j とすべてのカメラピクセル i との間で対応する各個別の光輸送係数の判定はプロジェクトピクセルアレイ 27 における他のプロジェクトピクセル j からの光線寄与を避けるべきである。

【0094】

これを達成するために、まず j 番目のプロジェクトピクセルを「1」の値に設定し（すなわちオンにし）、プロジェクトピクセルアレイ 27 における他のすべての要素をゼロ値に設定する（すなわちオフにする）ことにより作成された初期の実投影画像 $_j$ を検討することができる。この場合、 j 番目のピクセルは光輸送係数を判定するためにテストされるプロジェクトピクセルである。これらのテスト条件は j 番目のプロジェクトピクセルに結び付いた入力位置において「1」の値を有し、すべての他のプロジェクトピクセルをゼロの値に設定している実投影ベクトル $R_{p\ r\ j\ c\ t\ _j}$ として記号化することができる。次に実投影画像 $_j$ を取り込み対応する ($m \times n$) の実取り込みベクトル $_j$ を判定することができ、これは光輸送行列 T の j 番目の列を定義する。

【0095】

所定の j 番目のプロジェクトピクセルに対する光輸送行列 T の光輸送係数の列を得るこの方法は光輸送行列 T 全体を取り込む系統だった方法は実プロジェクト 21 の各プロジェクトピクセル j を順にオンにし（プロジェクトピクセルを1つずつ）、実カメラでその対応する実画像としての取り込み画像ベクトル $R_{c\ p\ t\ r\ _j}$ を取り込むことであることを示唆している。 $p \times q$ 個のプロジェクトピクセル j がすべて順にオンにされ、それらに対応する実画像としての取り込み画像ベクトル $R_{c\ p\ t\ r\ _j}$ が取り込まれると、取り込まれたすべての取り込み画像ベクトル $R_{c\ p\ t\ r\ _1} \sim R_{c\ p\ t\ r\ _{(p \times q)}}$ はグループ化され光輸送行列 T を形成する。各取り込み画像ベクトル $R_{c\ p\ t\ r\ _j}$ は光輸送行列 T において光輸送係数入力の1列をなす。これは ($p \times q$) 列および ($m \times n$) 行の個別の光輸送係数を有する光輸送行列 T をもたらす。

【0096】

しかし光輸送行列 T を判定するためのこの簡単で系統だったプロセスは ($p \times q$) の画像投影取り込み工程を必要とする、明らかに時間のかかるプロセスである。さらに、得られる光輸送行列 T は非常に大きく、($p \times q \times m \times n$) の要素からなる。光輸送行列 T のサイズから、デュアル画像を計算するのはデュアル光輸送行列 T^T (光輸送行列 T と同数の要素を有する) と仮想投影ベクトル $V_{p\ r\ j\ c\ t}$ (($m \times n$) の要素を有する長いベクトルである)との間で行列乗算を必要とする極めて計算集中的な演算である。

【0097】

過去において、光輸送行列 T を判定するプロセスを速めるために実プロジェクト (図 1 (a) の実デジタルプロジェクト 11 のような) の複数ピクセルをいちどきに適応的にオンにすることにより光輸送行列 T を判定する仕組みが提案されている。しかし、この方法は複雑で、投影される画像が複数の区域に分割され、各区域から 1 つのプロジェクトピクセルを選択する必要がある。選択されたプロジェクトピクセルはすべて同時に点灯され、かつ同時に点灯されたプロジェクトピクセルはその間の光干渉を排除するために互いに充

10

20

30

40

50

分な距離を確保するようにする。従って、この仕組みは選択された個々のプロジェクタピクセル間の光干渉が最小限であることを確保する必要があるので場面に依存する（すなわち表示環境に依存する）。このように複数区域の作成は表示環境に依存する。この仕組みは投影区域の物理的な検査および手動の設定を必要とするので、一般的な使用または自動化にあまり適さない。

【0098】

本発明の一特徴は光輸送行列 T の決定における手順工程を削減する方法を提案する。すなわち、 $(p \times q)$ の画像投影取り込み工程（および $p \times q$ の取り込み画像の格納）を必要とする代わりに、ここで提案される方法は「 p 足す q 」[すなわち $(p + q)$] 画像しか取り込みず、以下に説明する特定の状況においてこの数はさらに劇的に減らすことができる。

10

【0099】

本方法は次の仮定に基づいている。つまり殆どのプロジェクタ - カメラ表示用途において、実プロジェクタ 21 から放射される任意の別個の光線 1b および光線 1c は通常、実カメラピクセルアレイ 29 における別個の部分に当たる。すなわち、プロジェクタピクセル j から放射された任意の光線は実カメラピクセルアレイ 29 全体にわたって拡散されず、実カメラピクセルアレイ 29 内の明確な部分で光フットプリントを形成する小グループのカメラピクセル内に主に集中する。さらに、隣接する光フットプリントにおいて重なり合いはないか無視できる（すなわち別個のプロジェクタピクセル j から投射された別個の光線が当たるカメラピクセル i に重複が殆どない）。この理想的な光線分布条件は実カメラ 25 の解像度が実プロジェクタ 21 の解像度よりはるかに高い（すなわち少なくとも 2 倍高い）場合に仮定できる。

20

【0100】

別個の光線が重ならないというこの仮定はもちろん一般的に正しくない。例えば、実カメラ 25 の解像度が実プロジェクタ 21 の解像度に匹敵するか、またはそれより小さい場合、または光拡散材料が実プロジェクタ 21 から実カメラピクセルアレイ 29 に投影された光線の光パスの間に位置する場合理想的な光線分布条件は達成できないかもしれない。例えば、映写場面（すなわち表示環境または映写表面 / 領域）がコップ一杯の牛乳など光拡散材料を含む場合、投射された光線は拡散され、実カメラピクセルアレイ 29 において異なった光線間でかなり光の重なり合いがある可能性は非常に増える。しかし高解像度の映写を確保するよう設計された表示構成はこのような光拡散材料がない可能性が高く、投射された光のフットプリントは各々隣とは実質的に区別されることが殆ど保証される。つまり、高解像度の映写が望ましい会場や設定の場合、投影画像の光パスに沿って光拡散物品はない可能性が高い。

30

【0101】

本第 1 実施形態において、実カメラ 25 のピクセル解像度は実プロジェクタ 21 のそれより少なくとも 4 倍高く、映写場面には光拡散材料がなく、理想的な光線分布状態の条件が満足されると仮定される。しかし、実カメラ 25 の解像度が実プロジェクタ 21 より高いものに限定されず、または実プロジェクタ 21 と実カメラ 25 との間の光パスが光拡散材料のないものに限定されない、より一般的な状況に関する別の実施形態も下記に説明される。

40

【0102】

この仮定およびベクトル生成方法の最初の図形的描写が図 3 ~ 図 5 に示される。図 3 を参照すると、本発明の図示例は実カメラピクセルアレイ 29 より低いピクセル解像度を有するプロジェクタピクセルアレイ 27 を示す。説明の便宜上、プロジェクタピクセルアレイ 27 は 6 行と 7 列（すなわち $p = 6$ で $q = 7$ ）のみからなり合計解像度が「42」のプロジェクタピクセル j （すなわち (6×7) 要素）となるように図示される。同様に説明の便宜上、実カメラピクセルアレイ 29 は 12 行と 14 列（すなわち $m = 12$ で $n = 14$ ）のみからなり合計解像度が「168」のカメラピクセル i （すなわち (12×14) 要素）となるように図示される。実カメラピクセルアレイ 29 の解像度はプロジェクタピク

50

セルアレイ 27 の解像度より 4 倍高く、プロジェクタピクセルアレイ 27 から実カメラピクセルアレイ 29 への光パス内に光拡散材料は存在せず、従って、本第 1 実施形態の理想的な光線分布条件が満足されていることが特記される。

【0103】

図 3 においてプロジェクタピクセルアレイ 27 内の各正方形は異なるプロジェクタピクセル j を表す。本例において、第 3 行と第 5 列に位置するプロジェクタピクセル j の光輸送係数を得ることが求められる（すなわちプロジェクタピクセル $j = (5, 3)$ がテストされるピクセルである）。上記に説明されるように光拡散、または光散乱、は無視される。このように、プロジェクタピクセル $j = (5, 3)$ はオンにされている、すなわち点灯されている、ことが白い正方形で示されるように図示され、他のすべてのプロジェクタピクセルは濃い灰色の正方形で示されるようにオフのままとして図示される。プロジェクタピクセル $j = (5, 3)$ は光線 1 を放射し、これはプロジェクタレンズ 3 を通過した後映写表面 5（これは基本的に実プロジェクタ 21 および実カメラ 25 が位置する環境で決定される一般的な映写場面である）に当たる。光拡散は本例において無視されるので、光線 1 は映写表面 5 から反射し、カメラレンズ 7 を通過した後、カメラピクセル i のグループに当たり、円で示される第 1 光線フットプリント F_{t1} で示されるように光線フットプリントを形成する。各プロジェクタピクセル j の解像度はカメラピクセル i のそれよりはるかに粗いので、実カメラピクセルアレイ 29 に作成された第 1 光線フットプリント F_{t1} は数個のカメラピクセル i を覆い、第 1 光線フットプリント F_{t1} の中央にあるカメラピクセル i （白い正方形で示される）がもっとも高い光強度を受け、第 1 光線フットプリント F_{t1} の周辺部にあるカメラピクセル i （薄い灰色の正方形で示される）はより少ない光強度を受け、第 1 光線フットプリント F_{t1} 内にないカメラピクセル i （濃い灰色の正方形で示される）は光を受けない。次に、実カメラピクセルアレイ 29 内のすべてのカメラピクセルの光強度測定を判定することによりテストされるプロジェクタピクセル $j = (5, 3)$ に対応する各カメラピクセル i の光輸送係数を決定することができる。

【0104】

上記に説明されたように、プロジェクタピクセルアレイ 27 を第 1 ベクトル（実取り込みベクトルとしての第 1 ベクトル $R_{p r j c t}$ ）と見て、得られる実カメラピクセルアレイ 29 にわたる取り込み画像を第 2 ベクトル（実取り込みベクトルとしての第 2 ベクトル $R_{c p t r}$ ）と見ることが役に立つ。実取り込みベクトルとしての第 1 ベクトル $R_{p r j c t}$ をどのように構築できるかを図示したものが図 4 に示される。まずプロジェクタピクセルアレイ 27 を 6 つの行、行 $row_1 \sim row_6$ に分け、次に各行を 90 度回転させ 6 つの列セグメント $col_seg_r1 \sim col_seg_r6$ を形成する。6 つの列セグメント $col_seg_r1 \sim col_seg_r6$ を次に端と端をつないで（ $p \times q$ ）要素（すなわち (6×7) または 42 要素）からなる 1 つの合成列を形成する。本例において、要素 $col_1_row_19$ （すなわち列 col_1 の上から 19 番目の要素）はプロジェクタピクセルアレイ 27 のアレイ位置 $(5, 3)$ にあるプロジェクタピクセル j に対応する。要素 $col_1_row_19$ は白い正方形として示され、他の要素はすべて暗くされた正方形で、要素 $col_1_row_19$ がオンになっている唯一のプロジェクタピクセルであることを示している。

【0105】

同様のプロセスが図 5 において実カメラピクセルアレイ 29 に適用され、実取り込みベクトルとしての第 2 ベクトル $R_{c p t r}$ が構築される。本例の場合、実カメラピクセルアレイ 29 は m 行（すなわち 12 行）に分けられ、各々 90 度回転させられ図示されていない 12 の列セグメントを形成する。図 4 でプロジェクタピクセルアレイ 27 の配置を変えたのと同じように、12 個の列セグメントは端と端をつないで（ $m \times n$ ）要素（すなわち (12×14) または 168 要素）からなる 1 つの合成列 $cmra_col_1$ （すなわち実取り込みベクトルとしての第 2 ベクトル $R_{c p t r}$ ）を形成する。

【0106】

本例において、アレイ位置 $(5, 3)$ にあるプロジェクタピクセル j から放射される光

10

20

30

40

50

線はカメラピクセルアレイ 2 9 上に第 1 光線フットプリント F_{t1} を形成すると仮定される。第 1 光線フットプリント F_{t1} の範囲は円で示され、実カメラピクセルアレイ 2 9 の 4 つの異なる行の 12 個のカメラピクセル i を包含する。第 1 光線フットプリント F_{t1} 内のカメラピクセル i は図 3 または図 4 のプロジェクトピクセル $j = (5, 3)$ (すなわち行列要素 $c_{01_1_row_19}$) により放射された光線からの光強度の少なくとも一部を受ける。第 1 光線フットプリント F_{t1} の中央にあるカメラピクセルはもっとも多く光強度を受け、白い正方形で識別され、第 1 光線フットプリント F_{t1} の周辺部にあるカメラピクセルはより少なく光強度を受け、薄い灰色の正方形として識別される。第 1 光線フットプリント F_{t1} 内にないカメラピクセルは濃い灰色の正方形で示される。図 5 に図示されるように、これらの白および薄い灰色の正方形は取り込み画像内の非ゼロ要素をなし、実取り込みベクトルとしての第 2 ベクトル R_{cptr} (すなわち列 $c_{mr_a_c_01_1}$) 内で多数のゼロ値要素 (すなわち濃い灰色の正方形) の間に散在する白および薄い灰色の正方形 (すなわち非ゼロ値の NZ 要素) として示される。

10

【0107】

プロジェクトピクセル j からの 1 つの光線の光フットプリントがどのように数個のカメラピクセル i を覆うかの拡大図を提供する第 2 の例が図 6 および図 7 に示される。図 6 において、別の典型的な実カメラピクセルアレイ 2 9 の部分図は第 1 行において 1 ~ n まで、第 2 行において $(n + 1) \sim (2n)$ まで、および第 3 行において $(2n + 1) \sim (3n)$ まで等々と水平に番号が付けられた個々のカメラピクセル i を示す。このシーケンスを続けると、最下段の行に沿ったカメラピクセル i は $(m - 1)n + 1 \sim (mn)$ まで番号が付けられることが理解されよう。

20

【0108】

別の典型的な 1 つのプロジェクトピクセル j からの 1 つの光線で、実カメラピクセルアレイ 2 9 に作用する第 2 光線フットプリント F_{t2} が円として示される。例示の目的から、第 2 光線フットプリント F_{t2} 内にないカメラピクセル i [すなわち j 番目のプロジェクトピクセルから放射された 1 つの光線が当たらないカメラピクセル i] は非常に暗く示され、第 2 光線フットプリント F_{t2} により部分的に覆われるカメラピクセル i は多少暗く示され、完全に第 2 光線フットプリント F_{t2} 内にあるカメラピクセル i は暗くされずに示される。技術的に周知のように、第 2 光線フットプリント F_{t2} に少なくとも部分的に覆われる各カメラピクセル i は受けた光の量に比例する光強度値を記録する。この光強度値はその個別的なカメラピクセル i の光輸送係数として割り当てることができる。あるいは、各カメラピクセル i の光輸送係数は個々のカメラピクセル i により記録された光強度値に比例するようにすることもできる。しかし、 j 番目のプロジェクトピクセルからの投射ビームが直接当たらないカメラピクセル i は光強度値がゼロ (またはゼロに近く、または所定の閾値光強度値未満) になるはずで、これらに対応する光輸送係数も同様に光強度値がゼロ (またはゼロに近く) になるはずである。

30

【0109】

図 7 を参照すると、図 6 の第 2 光線フットプリント F_{t2} に対応するような取り込み画像ベクトル R_{cptr_j} [または光輸送行列 T の j 番目の列] が示される。この光輸送行列 T の j 番目の列は図 6 の実カメラピクセルアレイ 2 9 において番号の付けられたカメラピクセル i に各々対応する、光輸送係数の番号付けされた垂直シーケンスとして図示される。取り込み画像ベクトル R_{cptr_j} の数のシーケンスは図 6 に示される実カメラピクセルアレイ 2 9 における個々のカメラピクセル i の水平に番号付けされたシーケンスに追従することが好ましい。図示されるように、取り込み画像化ベクトル内で第 2 光線フットプリント F_{t2} により覆われたカメラピクセル i に対応する要素のみ光輸送係数として非ゼロ値、すなわち「NZ」、を有する。他のセンサピクセルはすべて光輸送係数として「ZERO」値を有する。「NZ」は任意の非ゼロ光輸送値を表し、この値は対応するカメラピクセル i が受けた光強度の量に関連することが理解されよう。第 2 光線フットプリント F_{t2} は実カメラピクセルアレイ 2 9 の数行にまたがっているので、各行は取り込み画像ベクトル R_{cptr_j} に順に記載されており、ゼロ値光輸送係数の数個の長いシ

40

50

－ケンスが 3, 4 個の非ゼロ、NZ、値光輸送変数の間に散在する。

【0 1 1 0】

光輸送行列 T を決定する本新規方法に戻ると、まずプロジェクタピクセル j の個別の光輸送係数は光輸送行列 T の j 列にマップすることが特記される。プロジェクタピクセル間の最小限の重なりを仮定すると、撮像されたプロジェクタピクセルアレイ 27 内のプロジェクタピクセルの第 1 組 S1 は光輸送行列 T の対応する列の組（プロジェクタピクセル毎に 1 つ）にマップすることになる。

【0 1 1 1】

【数 3】

$$S1 \subset \{1, \dots, (p \times q)\}$$

10

さらにプロジェクタピクセルの第 1 組 S1 は目標のプロジェクタピクセル j、すなわちテストされる目標プロジェクタピクセル、を含むと仮定される。

【0 1 1 2】

実カメラ 25 により取り込まれたプロジェクタピクセルの第 1 組を同時に活性化することにより作成される投影画像の取り込み実画像としての第 1 実画像を R cpt r_S1 とする。

次にプロジェクタピクセルの第 1 組 S1 とはテストされるプロジェクタピクセル j のみ共有するプロジェクタピクセルの第 2 組 S2、

20

【0 1 1 3】

【数 4】

$$S1 \cap S2 = \{j\}$$

を考える。実カメラ 25 により取り込まれたプロジェクタピクセルの第 2 組を同時に活性化することにより作成される投影画像の取り込み実画像としての第 2 実画像を R cpt r_S2 とする。光輸送行列 T の j 番目の列の（テストされる目標プロジェクタピクセルに対応する、すなわち j に対応する）光輸送係数は、取り込まれた取り込み実画像としての第 1 実画像 R cpt r_S1 および第 2 実画像 R cpt r_S2 より、これらが共有する唯一の光線フットプリント（すなわち図 3、図 5、または図 6 の第 1 光線フットプリント F_t1 または第 2 光線フットプリント F_t2 に類似）を特定することにより直接得られる。この共通の光線フットプリントは第 1 組 S1 および第 2 組 S2 が共有する唯一の点灯されたプロジェクタピクセル j である、目標のプロジェクタピクセル j から放射された光線に対応することになる。

30

【0 1 1 4】

従って、次のステップは取り込み実画像としての第 1 実画像 R cpt r_S1 および取り込み実画像としての第 2 実画像 R cpt r_S2 が共有する 1 つの光線フットプリントをどのように特定するかを判定することである。この共通の光線のフットプリントを特定するには取り込み実画像としての第 1 実画像 R cpt r_S1 および取り込み実画像としての第 2 実画像 R cpt r_S2 双方をピクセル毎に比較して、比較した 2 つのピクセルの暗い方を特定する方法がある。例えば第 1 実画像 R cpt r_S1 において、個々の光線フットプリント内の、各々プロジェクタピクセルの第 1 組 S1 の同時点灯に対応する、センサピクセルのみが非ゼロ（NZ）の光強度値を有し、取り込み実画像としての第 1 実画像 R cpt r_S1 内の他のすべてのピクセルはゼロ値を有する、すなわち比較的暗くなる。同様に第 2 実画像 R cpt r_S2 において、光線フットプリント内の、同時点灯されたプロジェクタピクセルの第 2 組 S2 に対応するセンサピクセルのみが非ゼロ（NZ）の光強度値を有し、他のすべてのピクセルはゼロ値（または暗い）を有する（すなわち所定の閾値未満）。2 つの第 1 組 S1 および第 2 組 S2 は目標のプロジェクタピクセル j のみを共有しているので、取り込み画像双方の直接比較で取り込み実画像としての第 1 実

40

50

画像 R_{cptr_S1} および取り込み実画像としての第 2 実画像 R_{cptr_S2} 双方に共通する唯一の非ゼロ領域（すなわち暗くない領域）を特定することにより光線フットプリント内でプロジェクタピクセル j に対応するカメラピクセルが迅速に特定される。言い換えれば、取り込み実画像としての第 1 実画像 R_{cptr_S1} および取り込み実画像としての第 2 実画像 R_{cptr_S2} における点灯領域（すなわち光線フットプリント）の交差点が特定され、この特定された光線フットプリントはテストされる目標のプロジェクタピクセル j に対応する。

【0115】

これを達成する一方法は取り込み実画像としての第 1 実画像 R_{cptr_S1} および取り込み実画像としての第 2 実画像 R_{cptr_S2} 双方をピクセル毎に比較し、比較された 2 つのピクセルの暗い方を留保することである。このプロセスは、

10

【0116】

【数 5】

$$Tj \approx \text{MIN}(R_{cptr_S1}, R_{cptr_S2})$$

と表すことができ、列 T_j は光輸送行列 T の j 番目の列で、「MIN」は取り込み実画像としての第 1 実画像 R_{cptr_S1} および取り込み実画像としての第 2 実画像 R_{cptr_S2} においてより低い値のカメラピクセル（すなわちより低い取り込み光強度値を有する、より暗いピクセル）が留保され、より高い値（すなわちより明るい）カメラピクセルは切り捨てられる。このように、留保される唯一の高光強度値は第 1 組 $S1$ および第 2 組 $S2$ 双方に共通する光線フットプリントに対応する。

20

【0117】

言い換えれば、各個別のプロジェクタピクセル j の寄与は実カメラピクセルアレイ 29 のはっきりと区別された部分にマップされるため、カメラピクセル $i = 1$ から $i = (m \times n)$ までの中に取り込み実画像としての第 1 実画像 R_{cptr_S1} および取り込み実画像としての第 2 実画像 R_{cptr_S2} に共通で目標のプロジェクタピクセル j に対応するカメラピクセル i の組 L がある。

【0118】

【数 6】

$$L \subset \{1, \dots, (m \times n)\}$$

30

目標のプロジェクタピクセル j はプロジェクタピクセルの第 1 組 $S1$ および第 2 組 $S2$ の交差点で（すなわち目標のプロジェクタピクセル j は第 1 組 $S1$ および第 2 組 $S2$ 双方に共通する唯一のプロジェクタピクセルである）、

【0119】

【数 7】

$$S1 \cap S2 = \{j\}$$

40

となることが再度特記される。従って、目標のプロジェクタピクセル j に対応しない（すなわち組 L に含まれないピクセル）取り込み画像ピクセル中（取り込み実画像としての第 1 実画像 R_{cptr_S1} および取り込み実画像としての第 2 実画像 R_{cptr_S2} 双方にいて）、取り込み実画像としての第 1 実画像 R_{cptr_S1} または取り込み実画像としての第 2 実画像 R_{cptr_S2} のいずれかで比較されたカメラピクセルの少なくとも 1 つは光を受けていない。光を受けるカメラピクセルは光を受けないカメラピクセルより明るくなるため、演算 $\text{MIN}(R_{cptr_S1}, R_{cptr_S2})$ は組 L に含まれるピクセル [すなわち L] のみ点灯され、これは列 T_j 、すなわち光輸送行列 T における j 番目の列、の良好な概算となる。

【0120】

50

これはプロジェクタピクセルアレイ 27 における隣接プロジェクタピクセルの組が列および行（または 1 点のみで交差する、すなわち 1 つのプロジェクタピクセル j のみを共有する、任意のパターンの対）で点灯され、プロジェクタピクセルの点灯された列に対し取り込み画像の第 1 集合 R_{cprtrS_y} が作られ、プロジェクタピクセルの点灯された行に対し取り込み画像の第 2 集合 R_{cprtrS_x} が作られた場合、任意の個別のプロジェクタピクセルに対する光輸送係数は双方の集合を比較し点灯された列の取り込み画像が点灯された行の取り込み画像と交差する組 L を特定することにより得られ、交差点はプロジェクタピクセル j を単独に活性化することにより映写される光線に対応する。

【0121】

このように、光輸送行列 T を決定する一方法は点灯された q 列のプロジェクタピクセルの q 個の取り込み画像に対応する画像 $R_{cprtrS_y_1} \sim R_{cprtrS_y_q}$ の第 1 組を集め、点灯されたプロジェクタピクセルの点灯された p 行の p 個の取り込み画像に対応する画像 $R_{cprtrS_x_1} \sim R_{cprtrS_x_p}$ の第 2 組を構築することである。次にプロジェクタピクセルアレイ 27 におけるすべてのプロジェクタピクセル $j = 1 \sim j = (p \times q)$ に対し、行および列の取り込み画像の対、画像 $R_{cprtrS_y_a}$ および画像 $R_{cprtrS_x_b}$ が存在し、組としての画像 $R_{cprtrS_y_a}$ および画像 $R_{cprtrS_x_b}$ の交差点は目標のプロジェクタピクセル j の活性化により作成される光線フットプリントに対応する。

【0122】

従って、投影画像のセット、画像 $R_{prjctS_y_1} \sim R_{prjctS_y_q}$ および画像 $R_{prjctS_x_1} \sim R_{prjctS_x_p}$ を構築する必要があり、ここで各投影された画像 $R_{prjctS_y_1} \sim R_{prjctS_y_q}$ は任意の投影された画像 $R_{prjctS_x_1} \sim R_{prjctS_x_p}$ と対にされ、投影された画像の各対は 1 つの光線フットプリントのみを共有する。すなわち、

【0123】

【数 8】

$$\forall j \in \{1, \dots, (p \times q)\} \exists R_{prjctSx_a}, R_{prjctSx_b}, \dots | \dots, R_{prjctSx_a} \cap R_{prjctSx_b} = \{j\}$$

となる。上記の式は $\{1 \dots (p \times q)\}$ におけるすべてのプロジェクタピクセル j について投影画像の対が存在し、各々は異なって構成されたパターンを有し、構成されたパターンの交差点は共通のプロジェクタピクセル j に対応する 1 点（すなわちパターンの区分、または 1 つの光フットプリント領域）において交差することを意味すると解釈される。構成されたパターンのこのような対の基本例として垂直な光の線および水平な光の線の投影対があげられる。この場合、垂直な光の線の取り込み画像と、水平な光の線の取り込み画像との交差点はテストされる目標のプロジェクタピクセル j に対応するすべてのカメラピクセル i （すなわちテストされるプロジェクタピクセル j から放射された光線により作成される光線フットプリント内に入るすべてのカメラピクセル i ）を含む。

【0124】

従って、光輸送行列 T 内の任意の列 T_j [ここで $j = 1 \sim (p \times q)$] は画像 $R_{cprtrS_y_1} \sim R_{cprtrS_y_q}$ および画像 $R_{cprtrS_x_1} \sim R_{cprtrS_x_p}$ により合成することができる。この特性を満足する方法はピクセル座標を使用することである。画像 $R_{prjctSx_j}$ はプロジェクタピクセル j に等しい x 座標を有するピクセルのみオンになる第 1 投影画像とし、画像 $R_{prjctSx_k}$ は k に等しい y 座標を有するピクセルのみオンになる第 2 投影画像とする。そうすると $\text{MIN}(R_{prjctSx_j}, R_{prjctSx_k})$ はプロジェクタピクセル (j, k) がオンにされたことに対応する画像を与える。このプロセスは図 8 (a)、図 8 (b)、図 9 (a)、図 9 (b)、図 10 (a)、および図 10 (b) を参照するとより良く理解できる。

【0125】

図 8 (a)において、場面または表示環境は市松格子パターンを有する平たい映写表面

10

20

30

40

50

4 1 として例示される。市松格子パターンは投影光線の対比を提供することにより単に本説明を分かり易くするために示され、表示環境は全くパターンがなくても良く、不規則な形状をしていても良いことが理解されよう。各垂直な光の線および水平な光の線の相対位置は、これらを作成する時にどのプロジェクタピクセルがオンにされたか分かっているため、既知のもので、以下により詳細に説明するように、それらの既知の相対移動を用いて実プロジェクタ 2 1 を較正することができる。

【 0 1 2 6 】

まず、明るい垂直線、または垂直な光ビーム 4 7 (すなわち投影プロジェクタピクセルの列から同時に放射された光線の列) が実プロジェクタ 2 1 により映写表面 4 1 に投影される。この場合、第 1 の明るい垂直線 4 7 _ k はプロジェクタピクセルアレイ 2 7 内の、
10 k に等しい y 座標を有するすべてのプロジェクタピクセルをオンにすることにより生成される。実カメラ 2 5 が次にこの画像 $R_{c p t r_S y_k}$ をプロジェクタピクセルの点灯された列の一例として取り込む。

【 0 1 2 7 】

図 8 (b) では図 8 (a) に類似したすべての要素が同様の参照番号を有するが、実プロジェクタ 2 1 が光線の第 2 の明るい垂直線 4 7 _ t を映写表面 4 1 に投影する。この場合、第 2 の明るい垂直線 4 7 _ t は t に等しい y 座標を有するすべてのプロジェクタピクセルをオンにすることにより生成される。実カメラ 2 5 が次にこの画像 $R_{c p t r_S y_t}$ を点灯された映写ピクセル列の別例として取り込む。実プロジェクタ 2 1 はプロジェクタピクセルアレイ 2 7 の q 列各々に対し光線の別の明るい垂直線を投影でき、実カメラ 2 5 は投影された明るい垂直線各々の別の画像を取り込めることが理解されよう。
20

【 0 1 2 8 】

図 9 (a) に関し、図 8 (a) および図 8 (b) に類似したすべての要素は同様の参照番号を有し、上記に説明される。この場合、実プロジェクタ 2 1 は明るい水平線、すなわち一行の投影 (プロジェクタ) ピクセルから同時に映写表面 4 1 に放射された水平な光線からなる光ビーム 4 9 を映写表面 4 1 に投影することが好ましい。第 1 の明るい水平線 4 9 _ j は j に等しい x 座標を有するすべてのプロジェクタピクセルをオンにすることにより生成される。実カメラ 2 5 は次にこの画像 $R_{c p t r_S x_j}$ をプロジェクタピクセルの点灯された行の一例として取り込む。

【 0 1 2 9 】

図 9 (b) において、実プロジェクタ 2 1 は光線の第 2 の明るい水平線 4 9 _ r (同時に点灯された個別の光線からなる) を映写表面 4 1 に映写する。前と同じように、第 2 の明るい水平線 4 9 _ r は r に等しい x 座標を有するすべてのプロジェクタピクセルをオンにすることにより生成され得る。実カメラ 2 5 は次にこの画像 $R_{c p t r_S x_r}$ をプロジェクタピクセルの点灯された行の別例として取り込む。実プロジェクタ 2 1 はプロジェクタピクセルアレイ 2 7 の p 行各々に対し光線の別の明るい水平線を投影でき、実カメラ 2 5 は投影された明るい水平線各々の別の画像を取り込めることが理解されよう。
30

【 0 1 3 0 】

図 8 (a) からの取り込まれた画像 $R_{c p t r_S y_k}$ および図 9 (a) からの取り込まれた画像 $R_{c p t r_S x_j}$ をピクセル毎に比較し (あるいは各々の明るい垂直線および明るい水平線のみを比較して) 、演算 $MIN(R_{c p t r_S x_j}, R_{c p t r_S y_k})$ を用いて比較された 2 画像のピクセルの内より暗い方のみを留保し 2 つの内明るい方を切り捨てると、図 10 (a) に示される画像 4 3 ' が生成される。図 10 (a) および図 10 (b) において図 8 (a) 、図 8 (b) 、図 9 (a) 、および図 9 (b) に類似したすべての要素は同様の参照番号を有しているがアポストロフィが加えられ、上記に説明される。

【 0 1 3 1 】

画像 $R_{c p t r_S x_j}$ のほとんどは画像 $R_{c p t r_S y_k}$ と同じであるため (すなわち、両者ともほとんど平たい映写表面 4 1 上の投影光線による単純な市松格子パターンからなる) 、比較された 2 つのピクセルの内より暗い方を留保しても結果として得られる
40
50

画像の大半は変わらない。つまり、比較された 2 つのピクセルが比較的同じである場合、いずれのピクセルを他方の代わりに選んでも結果として得られる画像 4 3' はあまり影響されない。しかし、取り込まれた画像 R c p t r_S y_k の第 1 の明るい垂直線 4 7_k を取り込まれた画像 R c p t r_S x_j の第 1 の明るい水平線 4 9_j 上にない対応ピクセルと比較した場合、2 つの画像ピクセルのより暗い方を留保することは画像 R c p t r_S y_k からの明るい垂直線のピクセルを切り捨て、画像 R c p t r_S x_j からの、プロジェクタピクセルではなく周辺光で照明されたそのままのピクセルを留保することになる。従って、明るい垂直線は生成された画像 4 3' から取り除かれる。同様に画像 R c p t r_S x_j の第 1 の明るい水平線 4 9_j 上のピクセルを画像 R c p t r_S y_k の第 1 の明るい垂直線 4 7_k 上にない対応ピクセルと比較した場合、2 つの画像ピクセルのより暗い方を留保することは画像 R c p t r_S x_j からの明るい水平線上のピクセルを切り捨て、画像 R c p t r_S y_k からの、プロジェクタピクセルではなく周辺光で照明されたそのままのピクセルを留保することになる。これにより第 1 の明るい水平線 4 9_j も生成された画像 4 3' から取り除かれる。しかし、第 1 の明るい垂直線 4 7_k が第 1 の明るい水平線 4 9_j と交差する領域内では、比較される画像ピクセルは双方とも光線による影響を示す明るく点灯されたピクセルである。この交差点領域内のこれら 2 つの画像ピクセルを比較すると、2 つの明るいビームのピクセルのいずれかが画像 4 1' 用に選択されることになる。その結果、画像 4 1' はプロジェクタピクセルアレイ 2 7 の座標 (j, k) から放射される投影光線に対応する、明るく点灯した領域 5 3 を示す。このように、座標 (j, k) を有するプロジェクタピクセルに対する光輸送係数は (j, k) におけるプロジェクタピクセルから個別的に投射される光線の画像を物理的に取り込む必要なくして、領域 5 3 から抽出することができる。

【0132】

図 10 (b) に第 2 の例が示され、図 8 (b) および図 9 (b) に対応する実取り込み画像（これは上述の指名方法に従い各々実取り込まれた画像 R c p t r_S y_t および画像 R c p t r_S x_r に対応する）の組み合わせがプロジェクタピクセルアレイ 2 7 の座標 (r, t) から放射された投影光線に対応する第 2 の明るく点灯した領域 5 5 をもたらす。

【0133】

同様のプロセスに従い、各プロジェクタピクセル j を 1 つずつ個別的にオンにし投影することなく、プロジェクタピクセルアレイ 2 7 におけるすべての投影ピクセルの光輸送係数を特定することができる。仮定的かつ単独的に活性化されたプロジェクタピクセルを生成してプロジェクタピクセルの光輸送係数を得るこの方法は実プロジェクタ 2 1 のプロジェクタピクセルアレイ 2 7 におけるプロジェクタピクセルの各行および列に対する (p + q) 個の取り込み画像しかせいぜい必要としない。さらに、一旦、すべてのピクセル映写位置が特定されると、(p + q) 個の取り込み画像は廃棄することができ、保存する必要があるのはインデックスおよび対応フットプリント情報のみである。

【0134】

この方法の例が図 11 に示され、プロジェクタピクセルアレイ 2 7（網目パターンとして図示）の部分図が実カメラピクセルアレイ 2 9（同様に網目パターンとして図示）と比較される。本例において実カメラ 2 5 のピクセル密度（すなわち解像度）が実プロジェクタ 2 1 の解像度より高いことが好ましく、従って、単一のプロジェクタピクセル j から放射された光ビームは数個のカメラピクセル i にかかる光フットプリント（F 1 などの）を作り出す可能性があることをよりよく示すために図解上、部分的な実カメラピクセルアレイ 2 9 を図示する網目パターンはプロジェクタピクセルアレイ 2 7 をあらわす網目パターンより濃くなっている。

【0135】

図 11 の本例において、プロジェクタピクセルアレイ 2 7 のインデックスは円 1, 2, 3, … (q + 1) … (2q + 1) … 等々が個々のプロジェクタピクセル j を表す部分アレイで表される。同様に実カメラピクセルアレイ 2 9 は個々に活性化されたプロジェクタピク

10

20

30

40

50

セル j に対するフットプリント情報を表した円形のフットプリント $F_1, F_2, F_3, \dots, F_{(q+1)}$, … 等々の対応部分アレイが重ねられて示される（光散乱は無視すると仮定して）。このように、フットプリント $F_1, F_2, F_3, \dots, F_{(q+1)}, \dots$ 等々は各々プロジェクトピクセル $1, 2, 3, \dots, (q+1)$ 等々に対応する。

【0136】

この方法に従うと、2組の情報しか格納する必要がない。第1組の情報はプロジェクトピクセル j のインデクスに対応し、第2組はカメラピクセル i のグループを各プロジェクトピクセル j に結び付けたフットプリント情報に対応する。言い換えれば、ゼロ値の係数は格納する必要がなく、メモリ要件がかなり削減される。

【0137】

この情報を体系付ける第2の例が図11に示され、プロジェクトピクセル j のインデクス 6_1 はグレースケール（すなわち非ゼロ値、または「NZグレースケール」）のカメラピクセル i の情報のグループとしてのフットプリント情報 6_3 （すなわちもたらされた光ビームのフットプリントに対応する）を指す、またはそれに対応するように図示される。

【0138】

デュアル・フォトグラフィに向けて必要な光輸送係数を生成するために格納する必要のある画像数をどのように削減するか、および格納し操作する必要のあるデータの量をどのように削減するかを示した上、以下次の部分ではデュアル・フォトグラフィを実施する際の実際的な問題のいくつかを扱う。上記で説明したとおり、光輸送行列 T は非常に大きい可能性があり、その使用（またはその数学的転置のデュアル光輸送行列 T^T の使用）には大量のアクティブメモリ（例えばDRAM）および法外な演算処理能力 / 時間を必要とする。従って、デュアル画像の一般的な使用は今まで実用的でなかった。

【0139】

効率的にデュアル画像を構築するにはまず、

【0140】

【数9】

$$V_{cptr}'' = T^T * V_{prjct}''$$

10

20

30

であることが特記される。仮想カメラ 2_1 における仮想カメラピクセルアレイ 2_7 は実際には実プロジェクト 2_1 におけるプロジェクトピクセルアレイ 2_7 に対応するため、仮想カメラ 2_1 で得た仮想取り込みベクトル V_{cptr} 内の任意の仮想カメラピクセルに対し同じインデクスとして「 j 」を用いることが便利である。従って、仮想取り込み画像における各仮想プロジェクトピクセルに対するデュアル光輸送行列 T^T において対応する要素の行の関係は、

【0141】

【数10】

$$V_{cptr}''(j) = T^T_j * V_{prjct}''$$

40

と示すことができ、行 T^T_j はデュアル光輸送行列 T^T における j 番目の行を指す。

【0142】

上述のように、デュアル光輸送行列 T^T は光輸送行列の数学的転置（すなわち光輸送行列 T を対角線で回転させる）で、行 T^T_j の値（ここで j は $1 \sim (p \times q)$ の任意の値）は従って、光輸送行列 T の j 番目の列に対応する（すなわち T_{col_j} ）。光輸送行列 T の各列は $(m \times n)$ 個の要素を有するので（すなわち実カメラ 2_5 のピクセル解像度に等しい）、これは大量の要素のようにみえる。しかし、本実施において光輸送行列 T の各列において限定数の要素しか非ゼロでない（すなわち垂直および水平光ビームの交差点が当たっ

50

たカメラピクセル i 、すなわち光のフットプリント、に対応するもの) ことを鑑みると、光輸送行列 T の各列「 j 」内(さらには各行 T^T_j 内)における($m \times n$)個の要素の内数個のみ非ゼロであることは自明である。従って、 $Vcptr''(j)$ を演算する際に T_{COL_j} のすべての要素を調べる必要はない。実際、図 11 および図 12 に関連して上述のように、すべての個別に活性化されるプロジェクトピクセル j およびそれらに対応する光ビームのフットプリント情報 63 を示す単一のインデックス 61 が格納されることが好ましい。

【0143】

図 12 に示すように、インデックス 61 は投影された各光線(実プロジェクトタ 21 の個別のプロジェクトピクセル j からの)についてフットプリント情報 63 のリストを結び付ける。フットプリント情報 63 の各グループは光輸送行列 T の各列内の非ゼロ入力に対応しており、光輸送行列 T のある列の各行列演算の際にはこれら非ゼロのグレースケール値のみ調べる必要があることが理解されよう。このように、 $Vcptr''(j) = T^T_j * Vprjct''(j)$ の各値を決定するのに必要な計算の数はかなり削減される。

10

【0144】

言い換えれば、処理する必要のある光輸送行列 T の各列 G 内における要素のこのサブセット $SVcptr''(G)$ は $T_{COL_G}(a)$ と定義することができ、ここで a はサブセット $Vcptr''(G)$ における任意の仮想取り込みピクセルである[すなわち $a \in \{1, \dots, (p \times q)\}$]。従って、各サブセット $SVcptr''(G)$ に対し調べるべき要素のセットは、

20

【0145】

【数 11】

$$SVcptr''(G) = \{a \mid \forall z \in \{1, \dots, (p \times q)\} * T_{COL_G}(a) \leq T * T_{COL_z}(a)\}$$

と定義することができる。一般的に $SVcptr''(G) \subset (p \times q)$ であるので

【0146】

【数 12】

$$Vcptr''(G) = \sum_{a \in SVcptr''(G)} T^T_G(a) * Vprjct''(a)$$

30

を演算する方が、

【0147】

【数 13】

$$Vcptr''(j) = T^T_j * Vprjct''$$

40

を演算するよりはるかに時間がかかる。

【0148】

この方法を用いて生成されたデュアル画像の例を図 13(a) および図 13(b) に示す。図 13(a) は実プロジェクトタにより投影された根本画像を示す。図 13(b) は本方法の実施による演算の結果もたらされるデュアル画像を示す。図 13(b) のデュアル画像は実プロジェクトタ 27(すなわち仮想カメラ 21)により仮想的に取り込まれた画像を表しており、または言葉を変えると実プロジェクトタ 21 が「見た」画像である。

【0149】

上記の考察はより少ない画像の組から効率的にデュアル画像を演算する方法を示し、こ

50

れは演算時間と併せて画像取り込み時間も節約する。上述のように、実取り込み画像およびデュアル取り込み画像は各々実カメラ25および実プロジェクタ21双方を較正するのに用いることができる。

【0150】

つまり、プロジェクタ視点からの画像を直接取り込むことはできないため、簡単な解決はデュアル・フォトグラフィ手法を用いて対応カメラ画像からプロジェクタ視点画像（すなわちデュアル画像）を構成し、次に構成された画像を用いてプロジェクタを較正することである。例えば、入念に測定した特徴を有する既知の被写体の実画像をさまざまな角度からいくつか撮った後、被写体の既知の寸法を用いて異なった角度の視点により生じた歪みを補正することにより実カメラを較正することができる。同じ取り込み画像からデュアル・フォトグラフィを用いて実プロジェクタが見た仮想画像を次に上述のように生成し、実カメラに類似した方法で実プロジェクタを較正することができる。

10

【0151】

しかしこの簡単な方法と結び付いた一つの問題は光輸送行列Tを生成し操作し、大量のカメラおよびプロジェクタ画像ピクセルによりもたらされる大きな画像ベクトルを演算する難しさである。この労働集約的で高価なプロセスは上述のデュアル・フォトグラフィ手法を用いることにより相当緩和されるものの、図2(a)に示すようなプロジェクタ-カメラシステムにおける実プロジェクタ21を較正する目的から出願者は完全な光輸送行列Tを生成し完全なデュアル画像を作成する必要を避けながらもデュアル画像（すなわち実プロジェクタ21が「見た」画像）を用いる利点のいくつかを利用して実プロジェクタ21の較正を簡単にする新規な方法を開発した。

20

【0152】

映写場面が比較的平たいと仮定すると、光輸送係数を判定しなければならない光ファットプリントの数を大幅に減らすことができる。言い換えれば、プロジェクタを較正するために完全なデュアル画像を構築することなく、特に映写表面にあまり凹凸がない場合戦略的な点の組の位置を判定するだけで良いことに注目することにより完全な光輸送行列Tの生成を全く避けることができる。例えば、戦略的な点の組は図8(a)におけるプロジェクタから見た平たい映写表面41の市松格子パターン内における正方形の角ができる。映写場面の表面凹凸が大きいほど、戦略的に選ばれた点の組における点の数が多くなる。逆に映写場面が平たいほど、戦略的な点の組における点の数が少なくなる。

30

【0153】

この戦略的な組における点の数を減らすという目標を達成し、それにより完全な光輸送行列Tを生成し操作することを避け、さらにデュアル画像生成プロセスのいくつかの特徴を較正プロセスそのものに組み入れるために出願者はホモグラフィに基づく方法を適合させた。本発明のこの別の実施形態はプロジェクタ視点画像にわたり市松の角の特徴（または他の既知の特徴）の座標を、デュアル画像の構築および構築されたデュアル画像からの角の検出を必要とすることなく、直接的に演算する。

【0154】

この新規な方法において、実カメラを用いて平面状の市松格子の画像を取り込み、取り込み画像にわたり市松ブロックの角を検出することができる。市松模様は全く例示的な目的で用いられ、どのような場面を取り込むこともできることが理解されよう。本件の場合、投影された垂直および水平線の配向は知られているので、場面にかかわらず投影線の変動を特定し補正することができる。このように、プロジェクタ-カメラシステムの構成要素はすべて同時に較正される。

40

【0155】

次に、プロジェクタ画像は2つ（またはそれ以上）の別個の視点から見た単一場面の2つ（またはそれ以上）の眺めを関連付ける、いわゆる透視映写モデルに追随していることが観察される。つまり、異なった眺めは場面に対し異なった角度に位置しているので、同じ場面の異なった眺め（または画像）を「見る」ことになる。しかし、実場面は1つしかないので（場面の眺めの数に関係なく）、異なった眺め間の数学的関係を実現し、任意の

50

視点における任意の点を実場面において対応する点に結び付ける（かつ他の視点各々において対応する点に結び付ける）ことができる。

【0156】

これら別個の眺めの1つが実プロジェクタの「見た」仮想画像であると仮定し、第2の別個の眺めが実カメラにより取り込まれた実画像であるとみなすと、完全なデュアル画像を生成することなく透視映写モデル（2つの眺めを共通の実場面に関連付ける）により取り込み実画像から仮想画像に関係する何らかの情報を抽出できることになる。

【0157】

この方法を用い、出願者は完全なデュアル画像を必要とせずに実プロジェクタを較正するために充分な情報を抽出する方法を考案した。従って、デュアル画像が完全に作成されなくても、迂回的な方法ではあるがプロジェクタにカメラ較正手法を適用することができる。

10

【0158】

透視映写モデルの下で、平たい被写体を異なった眺めから映写した2つの画像間の関係は単純な線形投影変換またはホモグラフィである。この変換は平たい被写体上の任意点の座標（すなわち同次座標）をその平たい被写体の具体的な眺め上で対応する点の座標に関連付ける。本実施形態で、平たい市松格子のプロジェクタ視点画像は対応するカメラ画像のホモグラフィである。具体的には、実場面（すなわち市松格子）の任意点 P について、そのプロジェクタ視点画像 $Up = (u_p, v_p, 1)$ における同次座標およびカメラ画像 $Uc = (u_c, v_c, 1)$ における座標は次の方程式を満足する。

20

【0159】

【数14】

$$Up = \lambda * H * Uc$$

ここで U はスカラーで、 H は 3×3 のホモグラフィ変換行列（技術的に知られているよう）で、右下の入力は 1 に設定される。対応座標対は 3 つの線形方程式を提供し、方程式の内 1 つがスカラーを決定し、他の 2 つはホモグラフィ変換行列 H を決定するために用いられる。 3×3 のホモグラフィ変換行列 H には 8 つの未知の入力があるので、市松模様上の N の座標点（ここで $N = 4$ ）の対応が与えられると、プロジェクタ視点画像およびカメラ画像間のホモグラフィは $2N$ の線形方程式を解くことにより再生することができる。 N の数が大きければ大きいほど、プロジェクタ視点およびカメラ画像間の座標点を関連付けるエラーが低くなる。

30

【0160】

対応する座標を得るために、10 個の白点が市松格子パターンなどの場面上に投影されることが好ましい。投影された白点を有する市松格子パターンの画像が実カメラ 25 などの実カメラを用いて取り込まれ、カメラ画像における 10 個の点の座標が演算される。本プロセスで、10 個の点は取り込み画像において対応する座標の演算中区別されていることのみが必要である。これは 10 個の点を順に投影し、対応する座標を順に判定することによって達成できる。あるいは、異なった色の点を同時に投影して、異なった点を色で識別することができる。

40

【0161】

プロジェクタは好ましくは既知の相互関係において点を投影したので、投影画像における点の座標は分かっている。これは 10 対の対応座標をもたらし、1 組は実カメラで取り込まれたもの、第 2 組は実プロジェクタで投影されたものである。一旦、ホモグラフィが再生されると、カメラ画像において検出された市松格子の角の座標を直接変換してプロジェクタ視点画像において対応する角座標を演算することができる。次に前述のようなカメラ較正方法を用いてプロジェクタパラメータを較正することができる。

【0162】

この方法の例を図 14 に示す。図 14 において、特徴取り込み結果は次のとおりである

50

。輪郭で示されない円、または点（例えば点81）はホモグラフィを推定するために用いられ、輪郭を描かれた円、または点（例えば点83）は角の点特徴である。見られるように、輪郭を描かれた点83は実際の角上にあり、検出された各角のプロジェクタ座標が正しく取り込まれたことを示している。

【0163】

しかし点を投影する代わりに、1点で交差する複数対の線（すなわち上述の水平および垂直線）を投影する上述の方法を使用し、線の各対の1交差点から点情報を抽出することができることが理解されよう。さらに、投影された線の対の空間関係は既に分かっているので、線を上述の市松格子パターンのような既知のパターン上に投影する必要はない。つまり、プロジェクタのピクセルアレイのどの列と行が同時に点灯されたか分かっているので、その空間関係も分かっている。さらに、カメラはその特定の場面に対し予め較正されている必要はないことが特記される。逆にプロジェクタとカメラは同時に較正されることができる。

【0164】

前記に説明したとおり、点情報を抽出するためにパターンを使用するこの方法は一連のパターン投影および画像取り込み工程を含む。従って、較正されていないカメラが各投影パターン（すなわち垂直または水平線）を取り込む際、映写表面の凹凸による投影パターンの歪みをその時点では無視することができる。すべてのパターンが取り込まれると、垂直および水平線間の空間関係が分かっているのでプロジェクタとカメラとの間の空間関係（または変換）を設定することができる。つまり、投影された垂直および水平線間の空間関係は、それらを生成する際、どのプロジェクタピクセルが点灯されたか分かっているので、分かる。さらに、垂直および水平線の真の配向が分かっているので、映写場面上の凹凸および眺めの角度を補正することができる。

【0165】

垂直線の取り込み画像を水平線の取り込み画像と組み合わせて点情報を抽出する際、どのプロジェクタピクセルが抽出された点に関連しているかが分かっており、さらにカメラピクセルのどのグループが抽出された点（垂直および水平線の交差点で判定される）に対応するかを判定することができる。このようにプロジェクタとカメラとの間のホモグラフィ関係（すなわち一方の眺めを他方に変換）はいすれかの装置を個別に特定の映写場面（または映写表面）に対し較正する必要なく達成することができる。

【0166】

この方法は上述の光輸送行列Tを生成する簡略化された方法を取り入れているが、必要な画像投影取り込み工程の数を（p+q）から、望ましい点の数で決定される、（p+q）のほんの一部にまで削減する。場面環境が平たい映写表面であるプロジェクタ-カメラシステムを較正するのに最小限4個の点が用いられるが、映写表面の凹凸、画像取り込み工程におけるエラーの可能性、および投影された光の線またはパターンの周辺光ノイズの存在による識別エラーを考慮に入れるため、7つの水平線および7つの垂直線を投影して多ければ14~49個の点を生成すればほとんどのエラーを克服するのに十分であることが見出されている。映写場面の表面凹凸が多いほど、交差する線の対の必要数が多いことが理解されるであろうが、この7つの垂直線と7つの水平線という数は多くの実在する状況に十分であることが見出されている。（p+q）個よりはるかに少ない点の数しか必要でない1つの理由は、デュアル・フォトグラフィで使用するために完全な輸送行列を生成しようとしているのではなく、むしろ2つの視点、すなわち実カメラの実視点と実プロジェクタからの仮想視点、の間のホモグラフィ関係を見出すために既知の関係の点を数個特定することに关心があるからである。しかし、完全な光輸送行列Tの推定は点の間に円滑な遷移を仮定すれば数個の関係する点から生成できることが強調される。つまり、一旦、ホモグラフィ関係が得られると、 $U_p = *H*U_c$ の同じ関係を用いて推定する光輸送行列Tを構築する際に隙間を埋めることができる。

【0167】

上述の光輸送行列Tを生成する方法のさらに別の用途は以下の問題、つまりプロジェク

10

20

30

40

50

タとカメラの対があるとして、カメラが特定場面（すなわち投影環境）上で望ましい眺めを「見る」ようにしたい場合、望ましい眺めを実現するためにプロジェクタは何を場面に照らすべきか、を解決するために適用することができる。この課題は以下「ビュー投影」と呼ぶ。

【0168】

投影に必要な画像を見つけることは明らかに場面の幾何学的形状およびプロジェクタ装置の投影特性とともに、プロジェクタからの光の反射が正確な望ましい眺めを実現するよう場面の測光特性の詳細かつ正確な知識を必要とする。均一なアルベド、すなわち反射性、およびパラメトリック形状を有する表示表面の場合、これら必要な測定は明示的になすことができる。

10

【0169】

しかし恣意的に複雑で、不均一な表面反射率特性を有するノンパラメトリック形状上で高画質の画像を実現するのはまだ困難な課題で、幅広い種類のツールに精通していることと人間による相当な介入を必要とする本発明の一目的は完全に自動的で、平常通りのハードウェア設定以上のユーザ介入またはパラメータの微調整を必要としない測定を考案することである。言い換えれば、「ビュー投影」能力を達成したプロジェクタ・カメラシステムはボタンを押すだけで完全に自ら較正すべきである。

【0170】

困難な周囲環境の下で複雑な表面に正しく表示できる映写システムは多くの実社会の用途を有する。これを完全に自動化すると、専門的な表示設定が必ずしもないような周囲環境において使用することができる。本発明の別の目的は非標準的な機器の必要性を減らし、普通の消費者が入手できるような市販部品で良質な較正を達成することである。この目標はこれまでの実施形態の特徴および利点を自動化され簡略化されたプロセスに組み合わせることにより達成することができる。

20

【0171】

図2(a)における、1台の実プロジェクタ21および1台の実カメラ25からなる画像設定に関連して前記に説明したように、実プロジェクタ21はp行×q列のプロジェクタピクセルアレイ27を有し、実カメラはm行×n列の実カメラピクセルアレイ29を有する。実プロジェクタ21から放射される光線は場面（すなわち映写表面）から反射し、ある部分は最終的に実カメラピクセルアレイ29に到達する。一般的に、各光線は場面において分散、反射、屈折され、実カメラピクセルアレイ29のいくつか異なった箇所に当たる。このように、プロジェクタピクセルjから放射される光線は実カメラ25に到達し、実カメラピクセルアレイ29全体にわたりm×nの画像を形成し、各カメラピクセルiはある程度の光の量を受ける。投影された画像を(p×q)要素の実取り込みベクトルとしての第1ベクトルR_{p r j c t}と表し、取り込まれた画像を(m×n)要素の実取り込みベクトルとしての第2ベクトルR_{c p t r}と表すと、実プロジェクタ21および実カメラ25間の光輸送は以下のように記述することができる。

30

【0172】

【数15】

$$R_{cptr} = T * R_{prjct}$$

40

ここでTは光輸送行列である。図2(b)に図示されるように、ヘルムホルツの相反定理によると、方程式

【0173】

【数16】

$$V_{cptr''} = T^T * V_{prjct''}$$

50

を用い、実プロジェクト 2 1 は p 行 \times q 列の仮想カメラ 2 1 ”とみなされ、実カメラ 2 5 は m 行 \times n 列の仮想プロジェクト 2 5 ”とみなされる「デュアル」な設定をモデル化できることが示されている。これにより「デュアル・フォトグラフィ」が可能になり、実プロジェクト 2 1 により取り込まれたように見える画像を合成することができ、場面は実カメラ 2 5 から発せられた光で照らされるように見える。

【 0 1 7 4 】

本実施形態の目的である「ビュー投影」は別の問題に取り組んでいる。「ビュー投影」における関心は場面を照らすために用いた場合、カメラが所定の望ましい画像を取り込めのようなプロジェクト画像を見出すことである。光輸送の見地からすると、光輸送行列 T および望ましい実取り込みベクトルとしての第 2 ベクトル R_{cptr} が与えられ、実取り込みベクトルとしての第 1 ベクトル R_{projc} を再生、すなわち生成、したいわけである。光輸送行列 T の逆光輸送行列 T^{-1} を判定できれば実取り込みベクトルとしての第 1 ベクトル R_{projc} は明らかに次の方程式で見出すことができる。

【 0 1 7 5 】

【 数 1 7 】

$$R_{projc} = T^{-1} * R_{cptr}$$

【 0 1 7 6 】

光輸送行列 T の逆は過去において他の目的に用いられている。例えば、Seitz 他による論文「逆光輸送の理論」(「A Theory of Inverse Light Transport」)、(IEEE International Conference on Communications、ICC V05) は引用により全体として本明細書に組み入れられるが、これにおいて、光輸送行列 T の逆を用いて任意の場面における光の跳ね方を分析している。この方法において、場面は 1 個の跳ね画像の総計に分解され、各画像はカメラに到達する前に 1 回跳ねる光の寄与を記録する。「インパルス画像」の行列を用い、各 1 個の跳ね画像を演算して光がどのように場面を通して伝播するかを推測する。

【 0 1 7 7 】

Seitz 他が取り組んでいる問題はもちろん「ビュー投影」が課すものとは異なり、後者の場合は望ましい画像(例えば実取り込みベクトルとしての第 2 ベクトル R_{cptr})が提供または想定され、望ましい画像を実現できる未知の投影源を推測しようとする。この場合は、 $R_{projc} = T^{-1} * R_{cptr}$ を演算することが有益である。デュアル・フォトグラフィの $T^T * R_{projc}$ は仮想取り込みベクトル V_{cptr} を具現し、これは実取り込みベクトルとしての第 2 ベクトル R_{cptr} の異なったビューであるが、実取り込みベクトルとしての第 1 ベクトル R_{projc} の実際の源は示されないことが特記される。

【 0 1 7 8 】

しかし、光輸送行列 T の逆光輸送行列 T^{-1} は転置より演算が難しく、演算リソースをはるかに多く必要とする。事実、光輸送行列 T の全体の大きさが故に逆光輸送行列 T^{-1} の演算は甚大な演算リソースを必要とする極めて困難なタスクである。尚悪いことには、任意の行列の逆を見出すことは常に可能ではない。つまり、行列によっては逆がないこともある。

【 0 1 7 9 】

技術的に周知のように、恒等行列、または単位行列は I で示され、所定の行列 A に対し次の関係が適用できる。

【 0 1 8 0 】

【 数 1 8 】

$$AI = IA = A$$

【 0 1 8 1 】

行列 A が配列 $m \times n$ の行列である場合、乗法前の恒等行列 I は配列 $m \times m$ となり、一方

10

20

30

40

50

乗法後の恒等行列 I は配列 $n \times n$ となる。

【0182】

行列の乗法的な逆は通常単位行列 I の点から定義される。行列 A の左乗法的な逆は $B A = I$ となるような行列 B で、行列 A の右乗法的な逆は $A C = I$ となるような行列 C である。行列 A の左および右乗法的な逆が等しい場合、左および右乗法的な逆は単に「乗法的な逆」、または逆と呼ばれ、 A^{-1} で示される。

【0183】

幸いに、出願者は多くの表示設定において、光輸送行列 T の逆の近似はより簡単に演算され得、充分かもしれないことを見出した。しかし行列の逆に対する適当な近似を作成することは一般的に光輸送行列 T にない特定の基準を満足することが必要となる。従って、逆光輸送行列 T^{-1} の近似を作成する前に、適当に修正された光輸送行列 T を作成する必要がある。

10

【0184】

一般的に $T^{-1} = T^T$ である。しかし上記デュアル光輸送行列 T^T の推定を生成するのに用いられた制約の 1 つは同様に逆光輸送行列 T^{-1} の構築にも有用である。上述のように、ほとんどのプロジェクタ - カメラの表示用途において、プロジェクタから発射された任意の区別可能な 2 つの光線「 j 」および光線「 k 」は通常カメラセンサの区別可能な部分にあたる、すなわち光線「 j 」および光線「 k 」各々からの光に当たるカメラセンサのピクセルはあまり重複していないという限定を一般的に用いることができる。この特徴は以下「表示制約」と呼ばれる。

20

【0185】

場面が牛乳の入ったコップからなり、光が牛乳により拡散されかなりの重複をもたらす上述の例のように、一般的に場面においてかなりの光散乱がある場合「表示制約」に違反する可能性がある。「表示制約」に違反する場面の例が図 15 に示され、ここでは 2 個のワイングラスがプロジェクタと映写場面との間に含まれかなりの光散乱をもたらす。本例において文字からなる投影画像が場面に投影され、ワイングラスのために多くの光学的歪みを受けることが示される。以下に「表示制約」が破られた場合にこのような光散乱を補正する方法が提供されるが、本考査に関しては、「表示制約」が満足され、投影される各ピクセルが次のものと区別可能であるとまず仮定される。

30

【0186】

さらに前記に説明したとおり、光輸送行列 T の各列はプロジェクタからの 1 ピクセルによりもたらされる投影画像である。従って、列入力のすべては対応するプロジェクタピクセルから放射される光が当たったカメラピクセルに対応するものを除きゼロ値を有する。

【0187】

隣接する光フットプリント（隣接して点灯されたプロジェクタピクセルにより作成）に重複がない場合、光輸送行列 T の列は直交することに出願者たちは気が付いた。実世界の用途においては、表示制約により、隣接する光フットプリント間の重複は最小限で、これは異なったプロジェクタピクセルからの光は主に異なったカメラピクセルに当たることを意味している。この結果、2 列の光輸送行列 T が隣り合わせに配置されると、非ゼロ入力は殆どの場合並ばず、点の積はゼロに近くなる。これは光輸送行列 T の列が互いにほぼ直交していることを意味している。本手法において、これらの最小限に重なりあるピクセルは光輸送行列 T の作成から取り除かれ、その結果定義上直交する近似された光輸送行列 T がもたらされる。より詳細に後述されるように、この手法により光輸送行列 T の作成および操作をさらに簡略化することが可能になる。

40

【0188】

「表示制約」により、異なったプロジェクタピクセルによる光は主に異なったカメラセンサピクセルに当たる、すなわち異なったプロジェクタピクセルに対応するカメラピクセルは互いにあまり重複しない。この例は図 16 に示され、ここで第 1 プロジェクタピクセルからの第 1 光フットプリントは第 2 プロジェクタピクセルからの第 2 光フットプリントと 2 つの周辺カメラピクセル 2_i を共有する。これは光輸送行列 T の列が互いの隣に配置

50

されると、異なった列の間で非ゼロ入力は殆どの場合並ばず（すなわちより暗い非ゼロ入力がある程度の重複がある）、点の積はゼロに近くなる。

【0189】

この例が図17に示され、ここで列C1～C4の4列の光輸送行列Tが互いに隣接して図示される。列C1において、円で表される部分的光フットプリント $P_{i \times e1_1c}$ および部分的光フットプリント $P_{i \times e1_1c}$ は列C1内に画像として記録されている第1ピクセルの光フットプリントの最も明るい部分（すなわち中央部分）を特定する。図示されるように、完全な光フットプリントは他のそれほど明るくない部分的光フットプリント $P_{i \times e1_1b}$ 、および部分的光フットプリント $P_{i \times e1_1e}$ を含む。しかし説明の便宜上、各列C1～C4内の各光フットプリントの最も明るい部分のみ円で特定され、円は修正（すなわち推定）された光輸送行列Tの構築の際、留保されるピクセル係数入力を特定する。図16に示されたように、任意の光フットプリントの最も明るい部分は他のいずれの光フットプリントの最も明るい部分とも重なり合わない。従って、光フットプリントのより暗い、周辺部分（すなわち部分的光フットプリント $P_{i \times e1_1b}$ および部分的光フットプリント $P_{i \times e1_1c}$ ）を無視してサイズを変更した光フットプリントを形成すると、サイズを変更した光フットプリント間には重複が存在しないことが保証される。図17に示すように、隣接する列に沿って円で囲まれたピクセルのグループを比較すると、隣接する列の円に囲まれたピクセルはいずれも重複しないことが特記される。例えば、列C1における部分的光フットプリント $P_{i \times e1_1c}$ は列2における部分的光フットプリント $P_{i \times e1_2c}$ とは水平方向にオフセットされ、後者は順に列3の部分的光フットプリント $P_{i \times e1_3c}$ とオフセットされており、同様にこれは列4における部分的光フットプリント $P_{i \times e1_4c}$ とオフセットされており、以下同様である。同様に各列内の他の明るい部分的光フットプリントの円で囲まれた部分（すなわち部分的光フットプリント $P_{i \times e1_1d}$, $P_{i \times e1_2b}$, $P_{i \times e1_3b}$, $P_{i \times e1_4b}$ ）のいずれも互いに水平方向に並ばない。

10

20

30

【0190】

「表示制約」により異なったプロジェクトピクセルからの光は主に異なってカメラピクセルに当たるので、対応プロジェクトピクセルから放射された光に当たるカメラピクセルを定義する光フットプリント以外、列入力の殆どはゼロ値入力である。これは2列の光輸送行列Tが互いに隣接して配置された場合非ゼロの入力は殆どの場合並ばず、点の積はゼロに近くなることを意味する。これは光輸送行列Tの列が互いに直交することを意味する。

【0191】

これは光フットプリントの暗い周辺部を無視し、光フットプリントの最も明るい部分からなる、サイズを変更した光フットプリントを扱う場合は特に当てはまる。この場合、光輸送行列Tの列は互いに真に直交することになり、特定の列に自己以外の任意の列を掛けたものの変換はゼロ結果になることを意味する。

【0192】

このように、光輸送行列Tの直交的性質は改変光輸送行列T^{*}を生成する上述のプロセスにおいて光輸送行列Tの構築され方により確保されることができる。上記に説明するおり、各列内の非ゼロ値のみを格納することにより光輸送行列Tを格納するのに必要なメモリの量も同様に削減することができる。改変光輸送行列T^{*}においてこのプロセスは、格納される非ゼロ値は光輸送行列Tの各行内の最も明るい値のみを格納することにより迅速に特定でき、これによりサイズを変更した光フットプリントが自動的に作成されるので簡略化される。例えば図17において、円で示される部分的光フットプリント $P_{i \times e1_1c}$ ～ $P_{i \times e1_4c}$ 、円で示される部分的光フットプリント $P_{i \times e1_2b}$ ～ $P_{i \times e1_4b}$ 、および円で示される部分的光フットプリント $P_{i \times e1_1d}$ で示される、光輸送行列Tの行に沿った（すなわち別個のプロジェクトピクセルに対応し光輸送行列Tの隣接する列に配列される隣接画像情報の中）最も明るいピクセルを特定することができる。各行内の最も明るい値の入力のみが留保されると、図18の構造が得られる。特定さ

40

50

れた明るい部分は次に改変光輸送行列 T^* (以下、光輸送行列 T および改変光輸送行列 T^* は別途記述されない限り互換的に使用される) のベクトル表示 40 に組み合わせることができる。図示されるように、ベクトル表示 40 は光輸送行列 T の各行内の最も明るいピクセル値のみに自らを限定することにより光輸送行列 T の隣接する列の間の重複を取り除き、任意の場面に対し「表示制約」を有効に課す。従って、光輸送行列 T のこの構成は意図的に直交している。改変光輸送行列 T^* を完全に定義するには、どの部分的光フットプリント部分のグループがサイズの変更された同じ光フットプリントに属するかを示すインデックス情報を保持する第 2 の行列の列、または第 2 のアレイのみが必要であることが理解されよう。例えば本例の場合、グループを成す部分的光フットプリント $P_{i \times e1_1c}$ と部分的光フットプリント $P_{i \times e1_1d}$ は第 1 ピクセルに対応する、サイズが変更された 1 つの光フットプリントの一部である。同様に部分的光フットプリント $P_{i \times e1_2b}$ と部分的光フットプリント $P_{i \times e1_2c}$ は合わせて第 2 ピクセルに対する第 2 のサイズが変更された光フットプリントを形成し、以下同様である。

10

【0193】

光輸送行列 T の逆の近似を決定 (すなわち逆光輸送行列 T^{-1} を決定) するため、まず $A_{A^{-1}} = I$ で、恒等行列 I は左上の角から (行列位置 (1, 1) から始まり) 右下の角まで (行列位置 (r, g) で終わる) の対角線に沿い「1」の値に設定された入力値を有し、他の個所はすべて「ゼロ」(すなわち 0) に設定された入力値を有する行列からなることを特記することが有益である。逆光輸送行列 T^{-1} を演算するために、まず行列 T を定義し、行列 T における各列は正規化値で構成され、行列 T の各列が光輸送行列 T (または改変光輸送行列 T^*) における列に対応する。つまり、

20

【0194】

【数19】

$$\begin{aligned} \Xi r &= Tr / (\|Tr\|)^2 \\ r &= 1, 2, 3, \dots, pq \end{aligned}$$

ここで列 r は行列 T の r 番目の列である。演算 $T_{r \cdot}$ は光輸送行列 T の列 r における全値の平方した合計の平方根を定義しているので、 $T_{r \cdot}$ の平方は単に列 r における全値の平方の合計である。つまり、

30

【0195】

【数20】

$$(\|Tr\|)^2 = \left\{ \sum_{\varepsilon=1}^{\varepsilon=g} (Tr_{\varepsilon})^2 \right\}$$

列 r における各値の入力を列 r における全値の入力を平方した合計で割ることにより、演算 $\{T_{r \cdot} / (\|Tr\|)^2\}$ は光輸送行列 T の列 r における値の入力を正規化する効果を有する。ここで列 r の転置を取ると、すなわち第 1 列が最上行になり、最後列が最下行になるように横に反転させると、光輸送行列 T における要素の対応列の正規化値である要素の行がもたらされる。従って、光輸送行列 T の各列について次の結果が得られる。

40

【0196】

【数21】

$$(\Xi r^T)^* (Tr) = I$$

および、

【0197】

50

【数22】

$(\exists r^T)^*(T\omega) = 0 \quad r \neq \omega$ の場合

言い換えれば、光輸送行列 T のある列を r^T の対応行で乗じると常に数字の「1」となり、光輸送行列 T のすべての列を r^T の対応行で乗じると、対角線に沿って数字の「1」がある行列を実現し、他のすべての個所にゼロを置いて実現された行列を完全に埋めることができる。

【0198】

10

従って、列が互いに直交する（またはさせられる）光輸送行列 T の場合、かつ行列 r の特定の構成が与えられる場合、行列 r の転置は当然、光輸送行列 T の逆に等しく（すなわち $r^T = T^{-1}$ ）、または少なくとも行列 r は光輸送行列 T の左乗法的な逆であることが示されている。従って、 $R p r j c t' = r^T * R c p t r'$ となる。

【0199】

20

実際にカメラセンサに当たるプロジェクタピクセルの部分のみ再生できることが特記される。いずれのカメラピクセルにも当たらないプロジェクタピクセルに対し、光輸送行列 T の対応列はゼロを含み、上記方程式 $r = T r / (T r)^2$ は未定義である。このような場合、行列 r の対応列はゼロ列として設定されることが好ましい。このように r^T はプロジェクタおよびカメラの視野で重なる領域を覆う光輸送行列 T の部分の逆であり、 $R p r j c t'$ において重なる領域に入るプロジェクタピクセルのみ再生し、他のピクセルは抹消する。以下の考察において、ビュー投影行列 T^T は望ましい眺め c が与えられた場合 $p = (T^T * c)$ と定義される投影画像 p を見出すことができ、投影画像 p を投影すると基準カメラ位置から見た場合、望ましい眺め c と同じ外観を有する場面を実現する。ビュー投影行列 T^T は事実上、逆光輸送行列 T^{-1} の近似であり、本明細書ではそのように用いられるので、以下考察の残りの部分で逆光輸送行列 T^{-1} とビュー投影行列 T^T は別途の指示がない限り互換的に用いられる。

【0200】

30

真の逆光輸送行列 T^{-1} の近似を演算する効率的な方法が実証されたが、原形において行列は未だに膨大で、大量の記憶および言うまでもなく行列乗算を行うだけでも相当な演算能力を必要とする。ここでも、真の逆光輸送行列 T^{-1} は「表示制約」に支配されている。

【0201】

従って、デュアル光輸送行列 T^T を縮小するための上記に記述された圧縮手法に類似したものをビュー投影行列 T^T に適用することが望ましい。このように、ビュー投影行列 T^T は非ゼロ入力の第1列および対応するインデックス値の第2列からなる表示に縮小することができる。「表示制約」は光輸送行列 T の別個の列における非ゼロ入力が同じ行に並ばないことが要件であることが想起されよう。これは光輸送行列 T の各行において非ゼロ入力が1つのみ（または所定の閾値を越えた入力が1つ）存在し、光輸送行列 T を表示するにはその入力とともにその列インデックスのみを格納すれば良い（格納されない光輸送行列 T の入力位置はデフォルトでゼロ値が指定されるという了解のもとで）。従って、光輸送行列 T を格納するために必要な空間は非ゼロの光輸送行列 T 入力の1列とインデックス値の第2列に縮小することができる。

【0202】

40

しかし、実世界の状況では光輸送行列 T の点灯されていない入力の多く（すなわち光ファットプリント内にないカメラピクセル i ）はセンサノイズ（すなわちカメラピクセルが捕らえたものの1つの指定された、すなわちオンにされた、プロジェクタピクセルに由来しない光）のため厳密にゼロではない。従って、各行に複数の非ゼロ値が存在する可能性があり、単一の非ゼロ値入力は光ファットプリントに対応し、残りの非ゼロ入力は光ノイズによるものである。この場合、光ファットプリントに対応する単一の非ゼロ入力は行内で最高の値を有する（すなわち最も明るい値を有する）可能性が高い。従って、光ノイズを除去

50

するために各行の最高値の入力（すなわち行最高値）を、その行において光ファットプリントに対応する単一の非ゼロ入力として特定し指定することができる。これは光ノイズによる低い値の入力を除去する。これらの行最高入力値を特定することにより、各列の非ゼロ値が得られる。従って、各プロジェクタピクセルに対し、対応するカメラピクセルの組とともにこれらカメラピクセル間の光強度の分布が特定される。通常の対応カメラピクセルの組（任意の所定プロジェクタピクセルに結び付いた）はカメラ画像全体の非常に小さいサブセットである。行列演算の際、これらの対応カメラピクセルからの入力のみを考慮すれば良いので、この疎行列表示を用いてビュー投影画像変換を行うのは非常に効率的である。

【0203】

10

実際には、「表示制約」を活用してプロセスを速める固定パターン走査手法を用いることが好ましい。上述のように、プロジェクタピクセル j からの寄与は光輸送行列 T の列「 j 」にマップされる。従って、プロジェクタピクセルの組 $\{1, \dots, p \times q\}$ は光輸送行列 T における対応する列の組にマップされる。このような、 $S_1 \cup S_2 = \{j\}$ であるような 2 組のピクセルを考え、2 組のピクセルを点灯した時に取り込まれる 2 つの画像を C_{S_1} および C_{S_2} とする。そうすると、

【0204】

【数23】

$$Tj \approx \text{MIN}(C_{S_1}, C_{S_2})$$

20

となることが示され、ここで列 T_j は光輸送行列 T の j 番目の列である。 C_{S_1}, C_{S_2} は光輸送行列 T における各々の列の総計、すなわち $C_S = \sum_{j \in S} T_j$ である。各個別的なプロジェクタピクセル j の寄与がカメラセンサの別個の部分にマップされるとすると、取り込まれた画像 C_{S_1}, C_{S_2} においてプロジェクタピクセル j に対応するピクセルの共通の組 $1 \cup \{1, \dots, m \times n\}$ が存在する。次に、 $S_1 \cup S_2 = \{j\}$ であるので、取り込まれた残りの画像ピクセルについて、画像の少なくとも 1 つはプロジェクタピクセルの第 1 組 S_1 または第 2 組 S_2 の 1 つから光を受けなかったことになる。光を受けたピクセルは光を受けないピクセルより明るくなるので、 $\text{MIN}(C_{S_1}, C_{S_2})$ はピクセル 1 のみが点灯された画像を実現し、これは列 T_j の良好な近似である。これは、もし、

【0205】

30

【数24】

$$Y_1, \dots, Y_p \quad X_1, \dots, X_q$$

の組を構築することができ、ここで、

【0206】

【数25】

$$\forall i \in \{1, \dots, (p \times q)\}, \exists X_j, Y_k \mid X_j \cap Y_k = \{j\}$$

40

の場合、 $j = C_{X_1}, \dots, C_{X_q}$ 、および C_{Y_1}, \dots, C_{Y_p} からの 1、…、 $p \times q$ である列 T_j を合成することができる。

【0207】

上述の考察において、投影画像の各対は 1 つの交差点しか共通していない。この特性を満足する構成は次のピクセル座標を指標として用いる。 X_j は x 座標が j に等しいすべてのピクセルがオンにすることにより作成される画像とし、 Y_k は k に等しい y 座標がオンにすることにより作成される画像とする。そうすると $\text{MIN}(X_j, Y_k)$ は座標 $(j$

50

, k) がオンである画像を与える。この方法を用い、 $p + q$ 個の画像を取り込むことは光輸送行列 T の $p \times q$ 列をすべて合成することになる。

【 0 2 0 8 】

光輸送行列 T を取り込むためのこの方法が各ピクセルを個別に取り込むより速く、上述の Sen 他 2005 年に關して説明された適合的仕組みより簡単ではあるが、この方法は文献で既に提案されている他の仕組みに比べまだ遅い可能性があることが特記される。特にプロジェクタ - カメラのピクセルの対応を確立する 1 つの方法は X 座標および Y 座標を時間系列的なバイナリコードで投影することで、理論的にはすべての対応が 10g (N) 時間で取り込むことができ、ここで N は最大座標値である。実際には、この方法は特に最小有効ビットが投影される際にカメラセンサの解像度に厳しい条件を課す。必要な画像数を減らすために同時に複数の縞を投影することも提案できる。しかしこれは画像において取り込まれた各縞を特定する何らかの方法、例えば目印となる縞のサブセットを特定すること、または色分けもしくは他の区別できるパターンを活用すること、が必要である。これらの仕組みは事実取り込みプロセスを速めるかもしれないが、さらに複雑さや脆弱性をもたらし、しばしば仕組みが正しく機能するように特徴検出パラメータを微調整しなくてはならない。本明細書の好ましい実施形態において微調整するパラメータはなく、ハードウェアの故障がない限り通常一度の取り込みでビュー投影行列の良好なコピーが得られる。

10

【 0 2 0 9 】

上記に説明されたように、ビュー投影行列 T は投影光学系、表示表面、およびこれらの相対位置付けにより導入される幾何学的および測光的歪みを補正することができる。ビュー投影行列 T のビュー投影能力の有効性はいくつかの実験を通して試験された。

20

【 0 2 1 0 】

ある場面の望ましい外観を制御できることを示すために、印刷されたポスターの外観を動的に変え得ることが示された。図 19 に例を示す。左側のポスター画像は白光の下で示される。右側のポスター画像はビュー投影照明の下で示される。キューブの面は同じ色に見えるようにされている。右側のポスター画像の上に 1 枚の白い紙を置くとこの望ましい眺めを実現するために用いられた実際の投影画像が明かされる。見る人々は動画を見た後すぐポスターの元の外観を忘れることが見出された。紙上で示すのは難しいが、カメラで観察されたこの場面は人間の目で観察されたものにかなり近い。ポスターはポスター内の色をサイクルして色が均一であるがごとく見せるようにして動画化された。測光的補正がリアルタイムで行われる間ポスターの上に映画も映写して、下のポスター画像の跡をすべて取り除いた。

30

【 0 2 1 1 】

前述のように、現在好まれる光輸送行列 T の構成は隣接して点灯されたプロジェクタピクセルによりもたらされた隣接する光ファットプリント間で最小限の重複を規定する「表示制約」に依存する。「表示制約」が守られない状況の例が図 15 に關し上記で考察され、ここでは 2 つのワイングラスがプロジェクタと映写表面、または場面、の間に配置される。しかしさらに上記に説明されるように、ビュー投影行列 T は光輸送行列 T の逆の近似を作成する際、光輸送行列 T の各列内の最も明るいピクセルのみを選択することにより任意の場面に「表示制約」を執行する（すなわち強要する）ことができる。本発明のアルゴリズムの有効性を各種条件の下（その有効性と正当化の基となる「表示制約」の下だけではなく）で試験するために「表示制約」が満足されない場合にどのように挙動するかを判定する実験が行われた。図 15 に図示する実験は、1 対のワイングラスがプロジェクタとその表示表面との間に置かれた形で設定された。ガラスにより導入された相当量の歪みのため、別個のプロジェクタピクセルによるセンサの光ファットプリントに多くの重複がある。この場面にビュー投影行列 T の生成において「表示制約」を執行するのと同じ取得方法および同じアルゴリズムを適用した。結果として演算されたビュー投影行列 T は図 20 に示すように、望ましくない歪みのほとんどを取り除くことができ、プロジェクタは指定表示表面に補正された画像を表示できることが見出された。

40

【 0 2 1 2 】

50

別の言葉で述べると、「表示制約」が保証できない場合、投影環境に擬似「表示制約」を強要する、ビュー投影行列 T の生成方法を光輸送行列 T の生成に適用することができる。具体的に、図16～図18に関し上述された簡略化された光輸送行列 T を生成する際、光ファットプリントのコンポーネントを特定するのに各行の最も明るいピクセルを選ぶことができ、それにより光拡散によるエラーおよび光ノイズによるエラーを大幅に削減し、または取り除くことができる。

【0213】

光輸送行列 T の逆を近似させる方法を定義した上、次に没入型表示システムにデュアル・フォトグラフィを用いて新型かつ複雑な設定をどのように達成できるかを説明する。

【0214】

図21について、好ましい実施形態において従来の前面投射型プロジェクタP1(図2(a)の実プロジェクタ21に類似)が没入型プロジェクタP2と併せて用いられる。視野(すなわちFOV)91で示されるように、好ましい本実施形態では前面投射型プロジェクタP1のFOV91で覆われる表示表面の部分は没入型プロジェクタP2の視野のサブセットであり、視野線(以降、FOVという)93で示される。前面投射型プロジェクタP1のFOV91は没入型プロジェクタP2のFOV93の範囲内に入るため、没入型プロジェクタP2が前面投射型プロジェクタP1により生み出された投影画像をシミュレートすることが望ましい。しかし、一般的に前面投射型プロジェクタP1のFOV91は没入型プロジェクタP2のFOV93におけるいずれの部分とも重なる必要は必ずしもないことが強調される。

【0215】

没入型プロジェクタP2のFOV93が前面投射型プロジェクタP1のFOV91の一部と重なる必要はないが、実カメラCを前面投射型プロジェクタP1および没入型プロジェクタP2に別個に結び付ける2つの輸送行列が作成されることが望ましい。前面投射型プロジェクタP1および没入型プロジェクタP2のFOVが必ずしも重ならないので2つの輸送行列は別個に生成できることが理解されよう。しかし好ましい本実施形態の具体例において、実カメラCは実カメラCのFOV95が前面投射型プロジェクタP1のFOV91のスーパーセットで没入型プロジェクタP2のFOV93のサブセットであるように配置される。FOV95で示されるように、実カメラCの視野は前面投射型プロジェクタP1のFOV91を完全に包含するが、没入型プロジェクタP2のFOV93により完全に包含される。没入型プロジェクタP2を用いて前面投射型プロジェクタP1から投影された画像をシミュレートするには、前面投射型プロジェクタP1からの第1投影画像 p_1 を実カメラCからの第1取り込み画像 c_1 に $c_1 = T_1 * p_1$ となるように関連付ける第1光輸送行列 T_1 をまず決定する。次に、没入型プロジェクタP2からの第2投影画像 p_2 を実カメラCからの第2取り込み画像 c_2 に $c_2 = T_2 * p_2$ となるように関連付ける第2光輸送行列 T_2 を決定する。その結果、次の関係が成立する。

【0216】

【数26】

$$c_1 = T_1 * p_1$$

および

$$c_2 = T_2 * p_2$$

没入型プロジェクタP2を用いて前面投射型プロジェクタP1からの第1投影画像 p_1 をシミュレートするためには、第1取り込み画像 c_1 (すなわち前面投射型プロジェクタP1から投影された画像の取り込み画像)が第2取り込み画像 c_2 (すなわち没入型プロジェクタP2から投影された画像の取り込み画像)と同じであることが必要である、すなわち、

【0217】

10

20

30

40

50

【数27】

$$c_2 = c_1$$

であることが必要で、これは次の関係、

【0218】

【数28】

$$T_2 * p_2 = T_1 * p_1$$

10

につながる。第2投影画像 p_2 (すなわち没入型プロジェクタ P_2 により投影された画像) を解くと次の関係が得られる。

【0219】

【数29】

$$p_2 = (T_2^{-1}) * (T_1 p_1)$$

20

これは第1投影画像 p_1 を作成するために前面投射型プロジェクタ P_1 を用いて画像を直接投影することができ、または変換された画像 $[(T_2^{-1}) * (T_1 p_1)]$ と定義] を没入型プロジェクタ P_2 で投影することにより同じ効果が得られることを意味する。ビュー投影行列は自然にプロジェクタ - カメラ対応をプロジェクタ - プロジェクタ対応に移すことが特記される。

【0220】

このような投影は図22に示され、没入型プロジェクタ P_2 を用いて図21の前面投射型プロジェクタ P_1 のような前面投射型プロジェクタがシミュレートされる。図22において、没入型プロジェクタ P_2 によりシミュレートされる仮想プロジェクタ P_1 " は点線で示される。従って、図21の前面投射型プロジェクタ P_1 により投影される第1投影画像 p_1 は図22の没入型プロジェクタ P_2 で変換画像 $(T_2^{-1}) * (T_1 p_1)$ を投影することにより再現することができる。これにより、見る人 $100A, 100B, 100C$ は前面投射型プロジェクタ、すなわち前面投射型プロジェクタ P_1 または仮想プロジェクタ P_1 " 、を遮蔽することを心配しなくて済む。画像は明らかに歪みがなく、仮想プロジェクタによる映画の再生はリアルタイムで流すことができる。上述のように T_2^{-1} は近似行列

T_2 で置き換えることが理解されよう。上記で述べたように、逆光輸送行列 T^{-1} を近似するビュー投影行列 T は以下の考察において別途記述されない限り自由に逆光輸送行列 T^{-1} を置き換えることができる。

【0221】

図23(a) ~ 図23(c) は前面投射の実画像 (図23(a)) および前面投射のシミュレートされた画像 (図23(b)) を並べてシームレスに連結したもの (図23(c)) を示すことによりシミュレーションの画質を図示する。図23(a) は図21の P_1 のような実前面投射型プロジェクタにより投影された画像の右側を示す。図23(b) は図23(a) の前面投射画像の左側を示すが、図23(b) において画像の左側は図22の P_2 のような没入型プロジェクタにより図22の仮想プロジェクタ P_1 " のような仮想前面投射型プロジェクタをシミュレートするように投影される。没入型プロジェクタにより作成される、左側のシミュレートされた前面投射画像の画質は図23(c) でより良く図示され、ここで図23(a) の右側の前面投射画像は図23(b) の左側の前面投射画像に繋げられ、並んで示され、実前面投射型プロジェクタおよびシミュレートされた仮想前面投射型プロジェクタにより各自作成された画像のシームレスな位置合わせがもたらされる。

40

【0222】

50

実前面投射型プロジェクタにより作成された実前面投射画像と没入型プロジェクタにより作成された、シミュレートされた前面投射画像を並べた比較を示す2つの追加例が図24に示される。図24において、表示画像の左半分は没入型プロジェクタにより仮想前面投射型プロジェクタからの表示をシミュレートするように作成され、表示画像の右側半分は実前面投射型プロジェクタにより作成されている。

【0223】

本手法の別の用途は図25(a)～図25(c)を参照するとより良く理解される。本例において、図25(c)の没入型プロジェクタP2を用いて各種の周囲照明効果(すなわち仮想環境)が作成される。カメラを、そのFOVが表示部屋のかなりの部分を覆うように配置すれば、ビュー投影を用いて壁が仮想モデルに従い照明されるような没入型環境を作り出すことができる。これを達成するために、図25(a)に示すように実カメラCはそのFOVが実部屋111の相当な部分を覆うように配置される。図25(a)において、実カメラCおよび没入型プロジェクタP2は実カメラCのFOVが没入型プロジェクタP2のFOVの全部(好ましくはより多く)と言わないまでも殆どを包含するように配置される。本例において、P2は没入型プロジェクタとして示されるが、没入型プロジェクタP2は前面投射型プロジェクタなど任意の種類のプロジェクタであって良い。実カメラCおよび没入型プロジェクタP2間に関係を確立するために、実カメラCを没入型プロジェクタP2に関連付ける第3光輸送行列 T_3 が前記に説明されたいずれかの方法を用いて取り込まれる、すなわち決定される。一旦、これがなされると、実カメラCにより見られる(すなわち取り込まれる)第3取り込み画像 c_3 は没入型プロジェクタP2により投影される第3投影画像 p_3 に次の関係に従って関連付けられる。

10

20

30

【0224】

【数30】

$$c_3 = T_3 * p_3$$

これは次をもたらす。

【0225】

【数31】

$$p_3 = (T_3^{-1}) * (c_3)$$

40

【0226】

このように、実部屋111における表示表面の仮想モデルを構築することができる。図25(b)に示す構築された仮想モデル部屋111"は例えばコンピュータ・シミュレーションであって良い。一旦、仮想モデル部屋111"が作成されると、各種のシミュレートされた照明効果(または投影画像または浮遊画像)を仮想モデル部屋111"に加えることができる。例えば、図25(b)は仮想モデル部屋111"が大きなロウソク113からのロウソクの光により照明されるところを示す。コンピュータ・モデルはさらに実カメラCの位置および解像度(図25(a)における)をモデル化し、図25(b)で点線の枠Cとして示される。コンピュータ・モデルは次に実カメラCの視点から仮想モデル部屋111"の合成的眺め c_3 "を「取り込み」(すなわち作成し)、図25(a)の実カメラCにより取り込まれたかのようにの仮想モデル部屋111"の実画像をシミュレートする。図25(b)のシミュレートされた照明効果は次に没入型プロジェクタP2を用いて変換画像($T_3^{-1} * (c_3)$)を投影することにより図25(c)の実部屋111において再現することができる。

【0227】

実部屋111の壁より大きい特大画像を映写することが望ましい。上記で考察したように、映写壁に対する実部屋の壁および天井の角度を補正するために各種の方法を用いて実プロジェクタを較正することができるが、本発明は異なった方法を用いてこの問題を解決

50

する。本例において、仮想モデル部屋 111”（図 25（b）の）は実部屋 111（図 25（c）の）に類似した寸法を有し、特大画像は歪みのない形で仮想モデル部屋 111”に重ね合わせられる。次に実カメラ C の視点から仮想モデル部屋 111”上の歪みなしの特大画像の合成的眺め c_3'' （すなわち合成取り込み画像）が作成される。次に没入型プロジェクタ P 2 に変換画像 $(T_3^{-1})^* (c_3'')$ を投影させ、実部屋 111 の壁に歪みのない特大画像を再現させる。結果は映写表面上の曲面（または他の凹凸）を補正するために没入型プロジェクタ P 2 を較正する必要がなかった歪みのない投影が得られる。

【0228】

示されたように、仮想プロジェクタおよび環境を組み合わせ、没入型映画ピュアを作成することができる。仮想環境はアクティブな視界もあるので、より大きい視野の表示を動画化し、より興味の湧く体験を作り出すことができる。

10

【0229】

上記に説明した手法は大視野（すなわち大きい F O V）表示の作成に適用することができる。大きい F O V は没入感を作り出し、見る人にとってより興味の湧く体験を提供する。本方法は非常に大きい F O V を有する没入型映写システムを説明する。システムはさらに映写の F O V が覆う領域からの光を取り込むことができる内蔵型大 F O V カメラ / 光センサを有して設計されている。センサによりシステムは画質を最適化し、より全般的にシステムがその環境とインタラクトできるように映写光を適合させることができる。本システムは主にサラウンド・ビデオ・コンテンツを表示したいことから主に動機付けられているが、この新映写システムは従来のビデオ・コンテンツを見るためにも用いることができる特記することが重要である。

20

【0230】

図 26 に関し、本発明による代表的な映写システムの最小限の形態は次のコンポーネント、前面投射型プロジェクタ 121、デジタルスチルカメラまたはデジタルビデオカメラであり得るカメラ 123、球形または別途の曲面ミラー 125、および上記コンポーネントの据付機構からなる。前面投射型プロジェクタ 121 からの光は曲面ミラー 125 から反射されてから、建物の壁、床、天井、および専用映写スクリーンを含む任意の表面であり得る映写表面 127 に到達する。映写表面 127 はさらに任意の形状であって良い。曲面ミラーで投射光を反射させるとプロジェクタの F O V が拡大される。プロジェクタを元とする場合もしない場合もある、環境からの光線も曲面ミラー 125 から反射されてからカメラに到達する。これも同様にカメラ F O V を拡大する。

30

【0231】

図 27 は図 26 の設計に基づいたプロトタイプを示し、図 27 の中で図 26 のものに類似した要素は類似した参照番号を有し、上記に説明される。本構成は、マルチウォールの没入型仮想環境を構築するという、スマートプロジェクタ - カメラシステムの主要な用途を際立たせている。従って、本例は従来の前面投射型プロジェクタ 121、高解像度のデジタルスチル型のカメラ 123、および半球形の曲面ミラー 125 からなる簡単なパノラマ映写設定を用いている。本設定において、曲面ミラー 125（「ドーム型プロジェクタ」と呼ぶことができる）は地域のコンビニ店で使用されるような低価格のプラスチック製半球型警備用曲面ドームである。このような曲面ドームは専門的に設計され製作された光学グレードのミラーに比べ少なくとも 3 枚は安い。さらに、据付機構も通常の金物・建材店で入手できる安価な部品で作られた。この構成で実験した際、曲面ドーム 125 は真的半球面からは（さらに言えばいずれの単純なパラメトリック形状からも）程遠いことが見出された。

40

【0232】

図 28 は図 27 における設定の別のビューで、カメラ 123 の観点から見た（非常に大雑把に）曲面ミラー 125 のビューを示す。図 28 において曲面ミラー 125 の反射で見られるように、カメラ 123 は曲面ミラー 125 の反射により床、少なくとも 3 つの垂直な壁、および天井を「見る」ことができる。

【0233】

50

図 29 (a)において、本映写システムを設置した部屋が周辺光の下で示される。図 30 (a)は較正されていないドーム型プロジェクタが市松格子の画像を表示するところを図示する。表示画像には明らかに相当な量の非線形幾何学的な歪みがある。図 30 (b)は同じ設定を示すが、ビュー投影行列を用いて幾何学的補正がなされている。画像は基準カメラの位置から撮られたので、ビューにおける直線は複数の壁にわたり直線のままである。

【 0 2 3 4 】

ビュー投影行列 T はさらに色および強度の変動とともに表示表面のアルベド / 反射率特性の不均一に対しても測光的に補正する。図 31 (a)は較正されていないドーム型プロジェクタが均一強度の画像を表示するところを図示する。見られるように、もたらされた画像は前方壁の左上および右上方向においてかなり暗い。図 31 (b)では同じ均一強度の画像が較正されたドーム型プロジェクタで投影され、より均一な強度を実現することが示される。

10

【 0 2 3 5 】

ビュー投影に必要なものすべてについて小型な代表があると、他の方法で較正するのが極めて困難な可能性のある表示システムを分析するのが容易になる。また、これらのシステムを構築して表示および撮像ハードウェアの限度まで精密に較正させるのに役に立つ。ここでは、前面投射型プロジェクタと組み合わせて上述のドーム型プロジェクタを用いたいくつかの例により逆光輸送行列 T^{-1} の有用性を示す。本例で使用されるような低予算のパノラマプロジェクタを用いた場合でも、ビュー投影行列は人を引き付ける没入型表示設定を組み立て得ることが見出された。

20

【 0 2 3 6 】

さらに、図 26 ~ 図 31 における本例のドーム型プロジェクタ設定は図 21 および図 22 の没入型プロジェクタ P_2 の代わりに用いて上述のように前面投射型プロジェクタのミュレーションを実現し、または図 25 (a) ~ 図 25 (c)における没入型プロジェクタ P_2 の代わりに用いて上述のように仮想環境を作り出すために使用できることが理解されよう。

20

【 0 2 3 7 】

本プロジェクタ - カメラシステムを用いた没入型投影照明の例を図 29 (b)に示す。見られるように、本プロジェクタ - カメラシステムは天井に加え 2 つの壁にも画像を投影することができる。この効果がビュー投影行列 T を用いてどのように達成されるかの例を図 32 ~ 図 34 に示す。

30

【 0 2 3 8 】

ビュー投影行列 T がまず上述のいずれかの方法を用いて生成される。上記に説明される通り、映写表面が主に平たい表面（または結合された平たい表面）、からなる場合、7 本の光の垂直線および交差する 7 本の光の水平線を用いて完全なビュー投影行列 T を近似することにより 49 以下の基準点を生成することができ、かつプロジェクタ対カメラの良好なレベルの較正を実現することができる。この場合、映写表面は平たいと仮定されるので、抜けている行列入力値は 49 の基準点から外挿することができる。

40

【 0 2 3 9 】

しかし、図 26 ~ 図 31 の例において映写表面は曲面ミラーからなるため、プロジェクタの解像度で完全なビュー投影行列 T が生成されることが好ましい。本例において前面投射型プロジェクタ 121 は $p \times q$ 個のプロジェクタピクセルの解像度を有するので、前面投射型プロジェクタ 121 とカメラ 123 との間の較正は $p \times q$ 個の光輸送基準点を生成することにより達成できるはずである。

【 0 2 4 0 】

図 32 に関し、図 27 の前面投射型プロジェクタ 121 は q 本の一連の垂直線 V_L_1 ~ V_L_q を個別的に映写表面（すなわちこの場合は図 27 の曲面ミラー 125）に投影し、これらは個別的かつ自動的に図 27 のカメラ 123 により取り込まれる。図 33 で、前面投射型プロジェクタ 121 は次に p 本の一連の水平線を個別的に投影し、続いてこれ

50

らも個別的かつ自動的にカメラ 1 2 3 により取り込まれる。上記に説明されるように、取り込まれた垂直および水平線は各々個別的に組み合わされそのユニークに一致する基準点 F_t (すなわち光ファットプリント) が特定される。このプロセスはすべてのユニークな交差点が特定され (図 3 4 において白い円で図示) 、その光輸送情報が抽出されるまで続けられる。前面投射型プロジェクタ 1 2 1 から放射される垂直および水平線は完全に垂直および水平であるが、もたらされる曲面ミラー 1 2 5 の投影線は曲面ミラー 1 2 5 の湾曲に追随することが理解されよう。

【 0 2 4 1 】

上述のように、ビュー投影行列 T の構築に「表示制約」が執行されることが好ましい。従って、望ましい画像 C があり、曲面ミラー 1 2 5 経由で実部屋 1 1 1 に望ましい画像 C を歪みなしに表示するためにどのように投影画像 p を変換するか判定したい場合、 $P = T \times C$ と定義される画像 C の歪んだバージョンを投影する必要がある。

10

【 0 2 4 2 】

これは図 3 5 に図示され、ここで望ましい画像 C は $m \times n$ 個の画像ピクセル入力値 $C_1 \sim C_{m \times n}$ からなるベクトル 2 0 0 として書かれている。ベクトル 2 0 0 に作成された ($p \times q$) 行および ($m \times n$) 列からなるビュー投影行列 T を掛け、($p \times q$) の変換画像 P が生み出され、これはベクトル 2 0 1 として書かれ、各々前面投射型プロジェクタ 1 2 1 の ($p \times q$) 個のプロジェクタピクセルにおいて対応するピクセルに適用される ($p \times q$) 個の画像ピクセル入力値からなる。もたらされる変換画像 P は p 行および q 列からなるように図示されている。

20

【 0 2 4 3 】

図 3 6 に関し、図 3 5 からの変換画像 P はプロジェクタ L C D 画像用の変換画像 P として前面投射型プロジェクタ 1 2 1 に送られ、曲面ミラー 1 2 5 (図 2 7) に投影される。実部屋 1 1 1 にもたらされる画像 2 0 3 は図 3 6 の望ましい画像 C の歪んでいない表示である。

【 0 2 4 4 】

本例の場合、カメラ 1 2 3 および前面投射型プロジェクタ 1 2 1 は変換画像 P を作成される前に較正されなかったことが特記される。逆に変換画像 P の歪みは、カメラ - プロジェクタ対の較正補正情報を含むビュー投影行列 T を用いて構築されたのでカメラ 1 2 3 と前面投射型プロジェクタ 1 2 1 間で較正されていないことに関連する問題を本質的に補正している。

30

【 0 2 4 5 】

望ましい画像がビデオ画像である場合、ビュー投影行列 T はビデオ画像に適用されることが理解されよう。つまり、ビデオ画像は順に並べられた複数のスチル画像を有してなるので、ビュー投影行列 T を順序付けられたスチル画像の各々に適用して変換ビデオ投影を実現することになる。

【 0 2 4 6 】

さらに強調すべきは、前面投射型プロジェクタ 1 2 1 の F O V とカメラ 1 2 3 の F O V とは一般的に異なっており、重なっても重ならなくても良い点である。2 つの F O V が相当重なる場合、カメラ 1 2 3 により取り込まれた画像は上述の方法に類似した方法で前面投射型プロジェクタ 1 2 1 からの投影画像の質を向上させるためのフィードバックとして用いることができる。例えば、カメラ 1 2 3 から前面投射型プロジェクタ 1 2 1 へのフィードバックを用いて表示表面の反射特性および形状の変化 (カメラ 1 2 3 から見た) を補正し、投影画像が平たく白い表面に投影されているが如く見えるようにすることができる。

40

【 0 2 4 7 】

カメラ 1 2 3 の F O V は前面投射型プロジェクタ 1 2 1 の F O V で覆われない領域を含むこともできる。例えば、前面投射型プロジェクタ 1 2 1 が図 2 9 (a) および図 2 9 (b) に示すテスト部屋の前および横の壁を覆うのに対し、カメラはプロジェクタ F O V 外の、場合によって見る人たちがいる領域を含む、領域を取り込むことができる。これによ

50

リシステムは見る人達または見る人達のポインティングデバイスを検出し追跡することにより見る人達に適応しインタラクトすることができる。カメラ 1 2 3 が例えばリモコンに搭載された小型の光を追跡してユーザ・インタラクションを促進することも可能かもしれない。

【 0 2 4 8 】

図 3 7 に關し、図 2 6 の構成に基づくが天井据付操作を対象とした別の構成が図示される。図 2 6 に類似した要素はすべて類似した参照番号を有し、上記に説明される。

【 0 2 4 9 】

図 3 8 および図 3 9 において、さらに 2 つの別の構成が示される。図 3 8 および図 3 9 において図 2 1 に類似した要素はすべて類似した参照番号を有し、上記に説明される。

10

【 0 2 5 0 】

図 3 8 では平面ミラー 1 4 1 が光路を折ることにより前面投射型プロジェクタ 1 2 1 およびカメラ 1 2 3 を曲面ミラー 1 2 5 の下に配置することができ、より小さいフットプリントを達成している。図 3 9 は展示ブース操作のために前面投射型プロジェクタ 1 2 1 、カメラ 1 2 3 、曲面ミラー 1 2 5 、および平面ミラー 1 4 1 をブース 1 4 3 内に入れるブース用設計を示す。この構成を用いると、同時に 2 つの投影画像、第 1 表示表面 A 上の第 1 の前方（または後方）投影画像および第 2 表示表面 B 上の第 2 の後方投影画像を実現することができる。

【 0 2 5 1 】

上述の設計のいくつかにおいて、プロジェクタおよびカメラは共通の光心を有しない。しかし、光心が同配列されたプロジェクタ - カメラ対を設計することは可能である。同配列光心を有するシステムはプロジェクタとカメラが全く同一の視野を有することができるが、同配列でない光心を有するシステムは環境の 3 D 再現ができる可能性がある。

20

【 0 2 5 2 】

ここまででは、提供された例は 1 台のカメラと 1 台のプロジェクタで構成されていたが、上記に示唆するように、複数のカメラおよびプロジェクタの組み合わせを使用することもできる。これは異なった視野を有する 2 台以上のプロジェクタからの 2 つ以上の投影画像をどのようにシームレスに統合もしくは組み合わせるか、または複数のプロジェクタを組み合わせてどのように大型視野の表示を作成するかという問題を呈する。従って、複数のプロジェクタを用いることにより可能になる視覚効果をいくつか説明する前に、まず複数のプロジェクタをどのようにシームレスにいっしょに使えるかを考察することが有用であろう。つまり、本発明を 1 台のプロジェクタと 1 台のカメラからなるシステムに適用された場合の効果は複数のプロジェクタおよび 1 台（またはそれ以上）のカメラを有するシステムにも拡大できるのである。

30

【 0 2 5 3 】

プロジェクタ - カメラシステムを較正し、上述の「ビュー投影問題」を解決するためのノンパラメトリック方法の開発が上記で考察している。つまり、カメラが望ましいビューを見るように、プロジェクタがどのように場面を照明すべきかを定義すること（すなわちプロジェクタが場面に何を投影すべきかを定義すること）のビュー投影問題が上記で考察される。要約すると、上述の開発はまず、プロジェクタから投影された投影画像 p をカメラで取り込まれた画像 c に $c = T p$ という関係で関連付ける光輸送行列 T について複数の構成を提供した。次に中間作業の行列 Ξ が次の関係により、埋められると定義された。

40

【 0 2 5 4 】

【 数 3 2 】

$$\Xi_r = Tr / (\|Tr\|)^2$$

……方程式 (1)

但し、 $r = 1, 2, 3, \dots, pq$

ここで列 r は行列 T の r 番目の列で、 $p q$ は光輸送行列 T における列の数である。次

50

に「表示制約」の下ではビュー投影行列 T を次のように定義できることが示され、

【0255】

【数33】

$$\Xi^T = T^{-1}$$

これは次の関係をもたらす。

【0256】

【数34】

$$p = \Xi^T * c$$

10

【0257】

上記に説明されたとおり、「表示制約」は情報表示を目的とした通常のプロジェクタとカメラの設定において、別個のプロジェクタピクセルから放射された任意の2つの光線は通常、カメラピクセルアレイにおいて別々の部分に当たる、すなわち別個の光線の各々からの光に当たったカメラピクセル i にあまり重複がないという観察に由来する。これは光輸送行列 T の列が互いに直交していることを意味し、方程式(1)の正規化プロセスにより光輸送行列 T の逆をもたらすことができる。さらに上記に説明されるように、「表示制約」が自然に守られない状況において、光輸送行列 T の構成を修正し、光輸送行列 T の各行における最も明るいピクセルが別個の光線によりもたらされる光フットプリントと一部であると宣言し、光輸送行列 T の同じの行における他のピクセルすべてがゼロ値の入力であると宣言することにより「表示制約」を人工的に強要することができる。この操作により光輸送行列 T は強制的に直交にされ、方程式(1)の適用が可能になる。

20

【0258】

このようにビュー投影行列 T は、望ましい眺め c が与えられた場合、 $p = T$ であり、 p を投影すると基準カメラから見た場合、望ましい眺め c と同じ外観を有する場面を実現する投影画像 $p = T$ を見出すという、ビュー投影問題を解決する。

30

【0259】

実際上大型視野表示、壁画面に囲まれる仮想居住ステーション、および他の多くの用途は個別プロジェクタ各々の限られた視野のため1台より多いプロジェクタの使用を必要とする。一般的に、1台を越えるプロジェクタを使用する場合これらのプロジェクタの視野は互いに部分的に重なり合う。このような2台以上のプロジェクタを有する表示システムを用いてビュー投影を達成する（または実際のこと異なる視野にわたり連続的な画像を投影する）には、個々のプロジェクタをモザイク化して望ましいカメラビューを生成する必要がある。

30

【0260】

本発明を用いたモザイク化の例が2台のプロジェクタシステムを使用して提示される。より多くのプロジェクタを有するシステムへの拡張は簡単なので、モザイク表示を構築する本方法は3台以上のプロジェクタを有する表示システムにも拡張できることが理解されよう。つまり、第1プロジェクタからの第1投影画像を第2プロジェクタからの第2投影画像と組み合わせるという、以下に説明するプロセスは第2プロジェクタからの第2投影画像を第3プロジェクタからの第3投影画像を組み合わせるのに適用でき、第1投影画像、第2投影画像、および第3投影画像を組み合わせたモザイク画像が作成される。同様に同じプロセスが第3投影画像を第4プロジェクタからの第4画像と組み合わせるのに適用でき、第1投影画像、第2投影画像、第3投影画像、および第4投影画像を組み合わせたモザイク画像が作成される。

40

【0261】

重なり合う視野（FOV）および双方のプロジェクタのスーパーセットとなるFOVを有するカメラからなるマルチプロジェクタ表示システムを検討する。各第1投影画像 p_1 および第2投影画像 p_2 をカメラで取り込まれた第1取り込み画像 c_1 および第2取り込み

50

画像 c_2 に関する第 1 光輸送行列 T_1 および第 2 光輸送行列 T_2 (別個に決定) は次の方程式をもたらす。

【 0 2 6 2 】

【 数 3 5 】

$$c_1 = T_1 * p_1$$

および

$$c_2 = T_2 * p_2$$

【 0 2 6 3 】

双方のプロジェクタの視野 F O V にわたる画像を表示するには、複合された画像 c がシームレスな形で第 1 取り込み画像 c_1 および第 2 取り込み画像 c_2 を組み合わせるような第 1 取り込み画像 c_1 および第 2 取り込み画像 c_2 を見出す必要がある。つまり、 $c = c_1 + c_2$ と定義される複合された画像 c はカメラから見た望ましい画像である。より具体的に、望ましい複合された画像 c を表示するために次の線形方程式を解いて適当な第 1 投影画像 p_1 および第 2 投影画像 p_2 を演算する必要がある。

【 0 2 6 4 】

【 数 3 6 】

$$c = c_1 + c_2$$

または

$$\begin{aligned} c &= (T_1 * p_1) + (T_2 * p_2) \quad \cdots \cdots \text{方程式 (2)} \\ &= [T_1 T_2] * [p_1 p_2]^T \end{aligned}$$

【 0 2 6 5 】

このような設定において、カメラピクセルは 1 台のプロジェクタからの 1 個のプロジェクタピクセルにより、または各々のプロジェクタからの 2 個のプロジェクタピクセルにより同時に、点灯される。前者の場合、カメラピクセルは第 1 投影画像 p_1 または第 2 投影画像 p_2 における対応プロジェクタピクセルについて線形方程式を呈する。カメラピクセルが 2 台のプロジェクタの F O V で重なる部分に当たる後者の場合、第 1 投影画像 p_1 または第 2 投影画像 p_2 における 2 個の対応プロジェクタピクセルについて線形方程式を呈する。各プロジェクタピクセルはいくつかのカメラピクセルを覆うので、いくつかの線形方程式により制約される。従って、すべてのカメラピクセルからのこのようない方程式は第 1 投影画像 p_1 またはプロジェクタ画像 p_2 に対し制約過剰の線形システムを形成する。

【 0 2 6 6 】

しかし、2 台のプロジェクタシステムの場合、上述の単独プロジェクタシステムでできるようにビュー投影行列を直接演算することができなくなる (すなわち $[T_1 T_2]$ の逆を直接演算できない)。これは異なったプロジェクタからのピクセルの投影は重なり合うことができるためで、従って、「表示制約」も複数のプロジェクタ間では成立せず、すなわち第 1 光輸送行列 T_1 における列は第 2 光輸送行列 T_2 における列と必ずしも直交せず、よって $[T_1 T_2]$ に対するビュー投影行列を演算するのに方程式 (1) を用いることができない。従って、方程式 (2) の線形システムを直接解く代わりに、収束するまで次の 2 工程を反復する交互性の線形解法が開発された。

【 0 2 6 7 】

10

20

30

40

【数37】

1. p_2 の現行推定を仮定し、

$$p_1 = (\Xi_1)^T * (c - T_2 * p_2) \cdots \cdots \text{方程式 (3)} \text{ を演算する。}$$

2. p_1 の現行推定を仮定し、

$$p_2 = (\Xi_2)^T * (c - T_1 * p_1) \cdots \cdots \text{方程式 (4)} \text{ を演算する。}$$

10

【0268】

方程式 (3)、方程式 (4) において、 $(\quad_1)^T$ および $(\quad_2)^T$ は 2 台のプロジェクタの各々のビュー投影行列で、最初の工程において第 2 投影画像 p_2 はゼロに等しく設定される。ビュー投影行列はプロジェクタとカメラとの画像間の対応とマッピングを自然に伝えるので、方程式 (3) および方程式 (4) に示す第 1 投影画像 p_1 および第 2 投影画像 p_2 が各々の補完的画像に収束するまで多くの反復を必要としない。实际上、第 1 投影画像 p_1 および第 2 投影画像 p_2 が組み合わせるとモザイク画像を形成する各々の補完的画像に収束するために数回（通常 5 回以下）の反復プロセスしかかからない。つまり、第 1 投影画像 p_1 は第 1 画像に収束し、第 2 投影画像 p_2 は第 2 画像に収束し、第 1 投影画像 p_1 および第 2 投影画像 p_2 の画像が投影され重ね合わされると、組み合わせられたモザイク画像を形成する。

20

【0269】

図 40 に関し、このプロセスを実施したシステムの例は第 1 プロジェクタ - カメラシステム対 221 および第 2 プロジェクタ - カメラシステム対 223 を含む。第 1 プロジェクタ - カメラシステム対 221 は、上述方法のいずれかを用いて構築された第 1 ビュー投影行列 \quad_1 により関連付けられた第 1 プロジェクタ 21a および第 1 カメラ 25a を含む。同様に、第 2 プロジェクタ - カメラシステム対 223 は、同じく上述方法のいずれかを用いて構築された第 2 ビュー投影行列 \quad_2 により関連付けられた第 2 プロジェクタ 21b および第 2 カメラ 25b を含む。さらに第 1 ビュー投影行列 \quad_1 および第 2 ビュー投影行列 \quad_2 は各々独立して生成され、第 1 ビュー投影行列 \quad_1 が生成される間、第 2 プロジェクタ - カメラシステム対 223 は消されており、第 2 ビュー投影行列 \quad_2 が生成される間、第 1 プロジェクタ - カメラシステム対 221 は消されていることが理解されよう。

30

【0270】

第 1 プロジェクタ - カメラシステム対 221 は第 1 投影領域 $R_{e g_1}$ を定義する第 1 視野 $F_{O V_1}$ を有し、第 2 プロジェクタ - カメラシステム対 223 は第 2 投影領域 $R_{e g_2}$ を定義する第 2 視野 $F_{O V_2}$ を有する。図示されるように、第 1 投影領域 $R_{e g_1}$ と第 2 投影領域 $R_{e g_2}$ は網目印で特定される区域内で互いに重なり合う。この重複領域はさらに $R_{e g_1 + 2}$ という記号が付けられている。重複領域 $R_{e g_1 + 2}$ のサイズは説明の便宜上大きくされており、小さい重複領域の方がより典型的であるが、重複の量は本用途にとって重要ではないことが理解されよう。

40

【0271】

方程式 (3) および方程式 (4) は幾何学的（すなわち空間的）および測光的（すなわち照明）補正を考慮しており、従って、隣接する表示画像間での必要な融合を処理することが特記される。これは光輸送行列 T が幾何学的情報を組み入れており、その結果、行列もそうであるからである。さらに、代表的な光輸送行列 T の光ファットプリント情報を特定する際、光ファットプリント内の取り込まれた各ピクセルは各カメラピクセルの 3 色 (RGB) のサブコンポーネントにおける任意の光強度変動について値の入力を有するので、完全な測光的情報も得られる（例えば図 3 における第 1 光線ファットプリント F_{t1} または図 6 における第 2 光線ファットプリント F_{t2} の光ファットプリントを構成する白および各明度の灰色のブロックを参照）。

50

【0272】

しかし、光輸送行列 T がバイナリの ON / OFF の方法で構成されるとすると（すなわち光ファットプリント内のピクセルは完全に ON と分類され（すなわち「0」から「255」の代表的な光度尺度で「255」の光強度値を有すると分類）、光ファットプリント外のピクセルは完全に OFF と分類される（すなわち「0」の光強度値を有すると分類））、光輸送行列 T を構成するこのバイナリの ON / OFF の方法は測光的情報をあまり有しない（事実上黑白の画像になるので）。しかし、この光輸送行列 T を構成するこのバイナリの ON / OFF の方法は完全な幾何学的情報を有するので、上記 $p_{\rightarrow 1}$ の方程式（3）および第 2 投影画像 p_2 の方程式（4）の 2 式は複数の投影画像の幾何学的モザイク化を判定することはできる。しかし、この場合、複数の投影画像をモザイク化する際、複数の投影画像の光強度を融合させるために光融合工程（以下に説明）を追加すると有用である。

10

【0273】

複数の投影画像の光強度を融合する追加工程が有用かもしれない別の状況は光輸送行列 T が限られた数の光ファットプリント情報から作成される場合である。上述のとおり、映写表面（または場面）が平たく、光輸送行列 T は限られた数の交差パターンを用い特定された限られた数の光ファットプリントを生成して推定されるような状況に当てはまる。映写表面は平たいので、光輸送行列 T 内で欠ける幾何学的情報はホモグラフィ手法を用いて推測することができる。推定された光輸送行列 T は従って、完全な幾何学的情報を提供するが、一般的に特定された限られた数の光ファットプリント間の場所について測光的情報は提供しない。

20

【0274】

映写表面が均一に白色である（たとえば白い映写キャンバスまたは白い壁）と仮定すると、すべての特定および推測された光ファットプリント情報に対し測光的情報として単に白色情報を挿入することにより光融合工程は飛ばすことができる。あるいは、映写表面が必ずしも白ではないが、均一の色であると仮定すると、とくに栄された 1 つの光ファットプリント（または 2 つ以上の平均）の測光的情報を割り当てることにより推定測光的測定を定義することができる。つぎに推測された光ファットプリントすべての測光的情報をこのように定義された推定測光的測定で埋めることができる。

20

【0275】

追加的光融合工程の話に戻り、利用可能なピクセル情報をすべてを用いて生成された光輸送行列 T がある場合でも、複数の投影画像をモザイク化する際、光融合工程を追加することは投影画像にとって有益である。この場合、追加的光融合工程から得られる情報は方程式（3）および方程式（4）から得られた融合結果に対し追加的パラメータとなることができる。

30

【0276】

この追加的光融合工程を明らかにする複数の投影領域の簡略化されたビューを図 4-1 に示す。前と同様に、第 1 投影領域 $R_{e g_1}$ は第 1 プロジェクタ - カメラシステム対 2-2-1 が提供する投影領域で、第 2 投影領域 $R_{e g_2}$ は第 2 プロジェクタ - カメラシステム対 2-2-3 が提供する投影領域である。説明の便宜、第 1 投影領域 $R_{e g_1}$ は垂直な平行線で示され、第 2 投影領域 $R_{e g_2}$ は水平な平行線で示される。重複領域 $R_{e g_1+2}$ はしたがって垂直および水平な平行線の交差で示される。特に興味があるのは第 1 投影領域 $R_{e g_1}$ 、第 2 投影領域 $R_{e g_2}$ 、および重複領域 $R_{e g_1+2}$ の一部にかかる濃い輪郭で示された望ましい投影領域 2-2-5 の定義である。望ましい投影領域 2-2-5 は望ましい組み合わせ（すなわちモザイク）画像が表示される領域を定義する。望ましい投影領域 2-2-5 には第 1 プロジェクタ - カメラシステム対 2-2-1 および第 2 プロジェクタ - カメラシステム対 2-2-3 の双方から画像寄与がある。望ましい投影領域 2-2-5 内で領域 $R_{e g_A}$ は第 1 プロジェクタ - カメラシステム対 2-2-1 のみにより提供される望ましい投影領域 2-2-5 の部分を特定し、領域 $R_{e g_C}$ は第 2 プロジェクタ - カメラシステム対 2-2-3 のみにより提供される望ましい投影領域 2-2-5 の部分を特定し、領域 $R_{e g_B}$ は第 1 プロジェクタ - カメラシステム対 2-2-1 および第 2 プロジェクタ - カメラシステム対 2-2-3

40

50

の組み合わせにより提供される望ましい投影領域 225 の部分を特定する。領域 Reg_B は第 1 プロジェクタ - カメラシステム対 221 または第 2 プロジェクタ - カメラシステム対 223 のいずれか片方のみにより提供されることもできることが理解されようが、画像境界（1 つのプロジェクタ - カメラシステムにより提供される画像が終わり第 2 のプロジェクタ - カメラシステムにより投影される第 2 の画像が始まる個所）における視覚的アーティファクトはプロジェクタ - カメラシステム間の遷移を融合することにより緩和され、または取り除かれることが見出されている。従って、この好ましい実施形態で、第 1 プロジェクタ - カメラシステム対 221 および第 2 プロジェクタ - カメラシステム対 223 双方が領域 Reg_B 内に作成された画像に寄与している。ここで問題は、領域 Reg_B 内の画像のどれだけ（すなわちどの部分）を第 1 プロジェクタ - カメラシステム対 221 および第 2 プロジェクタ - カメラシステム対 223 が各々提供しているかである。

10

20

30

40

50

【0277】

説明を簡単にするために、図 42 は第 1 投影領域 Reg_1 および第 1 プロジェクタ - カメラシステム対 221 の FOV 内にある望ましい投影領域 225 の部分、すなわち領域 Reg_B、が図示される。垂直な平行線は望ましい投影領域 225 に寄与しないため暗くされた第 1 投影領域 Reg_1 の部分を示す。矢印 1A, 1B, 1C, 1D は第 1 投影領域 Reg_1 の境界から領域 Reg_B に向かった離れ、望ましい投影領域 225 の境界に近づき、第 2 プロジェクタ - カメラシステム対 223 により提供される望ましい投影領域 225 に近づく際、正規化された光強度がどのように変化するかを示す。領域 Reg_A は組み合わせ部分の領域 Reg_B の外にあり、第 1 プロジェクタ - カメラシステム対 221 のみによって提供される。第 1 投影領域 Reg_1 の左側の境界から矢印 1A を横切り、領域 Reg_B に近づくと、矢印 1A は暗く図示され、すべての画像成分が第 1 プロジェクタ - カメラシステム対 221 により提供されていることを示す。矢印 1A に従い、領域 Reg_B に入ると、矢印 1A は最初暗く図示され、次に明るくなり（すなわち縞が細くなるよう図示）、画像の強度が最初は強く第 1 プロジェクタ - カメラシステム対 221 により提供されるが、アレイ 1A に沿って領域 Reg_B および第 1 投影領域 Reg_1 の右側の境界に向かい左から右に横切るにつれ強度が下がることを示している。矢印 1D は最初、領域 Reg_B の右端において第 1 プロジェクタ - カメラシステム対 221 により強度は提供されないが、領域 Reg_B 内で矢印 1A の終端に向かい右から左に横切るにつれ第 1 プロジェクタ - カメラシステム対 221 からの光強度は増加することを示す。同様に矢印 1B は望ましい投影領域 225 の境界から離れるように矢印 1B を下がると光強度が下がることを示す。同じく、矢印 1C は望ましい投影領域 225 の境界から離れるように矢印 1C を上がると光強度が下がることを示す。

【0278】

言い換れば、画像における光強度の変動は定義された最高光強度値、すなわち正規化値、の係数として表現することができる。この正規化値に係数「1」を掛けると定義された最高光強度値が提供され、同じ正規化値に係数「0.5」を掛けると定義された最高光強度値の半分が提供される。一定領域内で正規化値を変えることにより、その一定領域内の画像の強度を変えることができる。本例の場合、境界においてピクセルの光強度を急に変えるのではなく、画像境界を徐々に融合することにより境界における望ましくない光のアーティファクトを避ける。

【0279】

第 2 プロジェクタ - カメラシステム対 223 の視点から見た同様の構成が図 43 に示される。ここでも水平な平行線は望ましい投影領域 225 に寄与していないので暗くされた第 2 投影領域 Reg_2 の部分を示す。前と同様、領域 Reg_B は画像が第 1 プロジェクタ - カメラシステム対 221 および第 2 プロジェクタ - カメラシステム対 223 双方の組み合わせにより提供される融合区域を示し、領域 Reg_C は第 2 プロジェクタ - カメラシステム対 223 のみによって提供される望ましい投影領域 225 の部分を示す。矢印 2A, 2B, 2C, 2D は第 2 投影領域 Reg_2 から望ましい領域の境界に向かい、また領域 Reg_B 内で第 1 プロジェクタ - カメラシステム対 221 によって提供される望ま

しい投影領域 225 の部分に向かい移動するにつれ正規化光強度が変化するかを示す。領域 Reg_C は組み合わせ部分の領域 Reg_B の外にあり、第1プロジェクタ-カメラシステム対 221 のみによって提供される望ましい投影領域 225 の部分を定義する。領域 Reg_B に入ると、矢印 2A は画像の強度が最初は強く第2プロジェクタ-カメラシステム対 223 により提供されるが、アレイ 2A に沿って領域 Reg_B の左側の境界に向かい右から左に横切るにつれ強度が下がることを示している。矢印 2D は最初、領域 Reg_B の左端において第2プロジェクタ-カメラシステム対 223 により強度は提供されないが、領域 Reg_B 内で左から右に横切るにつれ第2プロジェクタ-カメラシステム対 223 からの光強度は増加することを示す。同様に矢印 2B は領域 Reg_B 内で望ましい投影領域 225 の境界から離れるように矢印 2B を下がると光強度が下がることを示す。同じく、矢印 2C は望ましい投影領域 225 の境界から離れるように矢印 2C を上がると光強度が下がることを示す。

【0280】

第1プロジェクタ-カメラシステム対 221 により投影される各ピクセルについて、第1投影領域 Reg_1 および Reg_B の左、右、上部、および下部の境界に対するその投影ピクセルの近さが判定される。これらのパラメータは投影されるピクセルの正規化光強度に影響する。第1プロジェクタ-カメラシステム対 221 からの投影ピクセルが第1投影領域 Reg_1 のいずれかの境界に近いほどその境界のパラメータ寄与が高くなり、より明るい正規化強度となる。第2プロジェクタ-カメラシステム対 223 からの投影ピクセルの場合も同じで、第2投影領域 Reg_2 および領域 Reg_B の4つの境界に対する投影ピクセルの近さが判定される。さらに、境界において急な光強度の変化を避けるために、境界に近い投影ピクセルの光強度は境界に近づくに合わせて調節される。

【0281】

例えば図41において、領域 Reg_A で矢印 A1 に沿って左から右に進むと第1プロジェクタ-カメラシステム対 221 により完全に提供される望ましい投影領域 225 の左の境界から、第2投影領域 Reg_2、および第1プロジェクタ-カメラシステム対 221 および第2プロジェクタ-カメラシステム対 223 の双方により提供される領域 Reg_B の左の境界に移ることになる。矢印 A1 に沿って左から右に進むと第1投影領域 Reg_1 の左の境界から離れて行くが、画像は第1プロジェクタ-カメラシステム対 221 のみにより提供されるので、第1プロジェクタ-カメラシステム対 221 により出されるピクセルの正規化光強度は正規の最も高い値にある。矢印 A1 の終わりに到達し、矢印 A2 の初めに到達すると、光融合はまさに始まるところなので第1プロジェクタ-カメラシステム対 221 はまだ最も高い正規化値にある。矢印 A2 の初めにおいて第1プロジェクタ-カメラシステム対 221 は最も高い正規化光強度を有し、第2プロジェクタ-カメラシステム対 223 はまだ最も低い正規化値を有する。矢印 A2 に沿って左から右に進むと（すなわち第1プロジェクタ-カメラシステム対 221 により専ら提供される領域 Reg_A から第2プロジェクタ-カメラシステム対 223 により専ら提供される領域 Reg_C に向かって）、第1プロジェクタ-カメラシステム対 221 の正規化光強度は最も高い正規化光強度から最も低いものに下がる。逆に矢印 A2 に沿って左から右に移動すると、第2プロジェクタ-カメラシステム対 223 の正規化光強度は最も低い正規化光強度から最も高いものに上がる。この光遷移は必ずしも直線的でないことが理解されよう。例えば、第1プロジェクタ-カメラシステム対 221 における正規化光強度は矢印 A2 に沿って右側の境界または領域 Reg_B に近づくにつれより大きい変化が起こることが好ましい。

【0282】

従って、矢印 A2 の終わりで、第2プロジェクタ-カメラシステム対 223 は正規化光強度が最高であり、第1プロジェクタ-カメラシステム対 221 は正規化光強度が最低である。矢印 A3 に沿って左から右に移動すると第2プロジェクタ-カメラシステム対 223 が投影される光をすべて提供し、第1プロジェクタ-カメラシステム対 221 による光の寄与はない。このように、領域 Reg_C 内で、第2プロジェクタ-カメラシステム対 223 は最も高い正規化光強度に維持され得る。

10

20

30

40

50

【0283】

上述の方程式(3)および方程式(4)は幾何学的および測光的双方の情報を用い、図44に示すように塑像に対し2台のプロジェクタをモザイク化することによってテストされた。2台のプロジェクタはカメラから見えるすべての表面を覆う必要があるので、これは無理に作った例ではない。この設定を用い、シームレスで歪みのない画像(カメラの視点から見た)が塑像上に投影することに成功した。さまざまな照明条件の下で塑像の画像をビュー投影することにより簡単に照明を変えることもできる。

【0284】

図45(a)および図45(b)は複数のプロジェクタを組み合わせる上記方法を用いた大型FOVプロジェクタシステムの構成を示す。図45(a)では1つの曲面ミラー125が複数のプロジェクタ-カメラ対145と組み合わせて用いられる。図45(b)では1つのミラーピラミッド151が複数のプロジェクタ-カメラ対145と用いられ、大きいFOVを実現している。図45(b)の構成の場合、すべてのプロジェクタの光心をミラーピラミッド内に同配列して、1つの仮想大型FOVプロジェクタを作り出すことができる。同様にカメラの光心も同配列して1つの仮想大型FOVカメラを作り出すことができる。

10

【0285】

図46は複数の大きいFOVプロジェクタ153a, 153bを用いてより大きい全体的投影FOVを達成することができるることを示す。1つ以上の通常のプロジェクタ155も組み合わせて使用することができる。見られるように、点線157で示されるFOVプロジェクタ153aのFOVは点線159で示されるプロジェクタ154bのFOVと、重複量161だけ重なっている。FOVプロジェクタ153a, 153bからの画像は複数のプロジェクタを組み合わせるための上述の方法を用いて組み合すことができる。

20

【0286】

本発明の別の実施形態は「ビュー投影行列」を実施するための最適化方策を扱う。これらの方策は一般的に消費者レベルの演算システムで利用できる並列処理能力をいかに最適に使用するかとともに「ビュー投影行列」の較正データをどのように並列処理アプリケーションにより適した形に配列するかを中心としている。

【0287】

1つ以上のプロジェクタを自動的に位置合わせできるよう較正する方法が上述される。上述の方法は非平坦で非白色の表面に投影画像を歪みなしに表示することに理想的に適しているが、これに限定されない。取られる方法は光輸送行列Tに基づいており、これは投影画像pおよびカメラで取り込まれた画像cの関係を $c = T p$ と表し、光輸送行列Tの各列はプロジェクタからの1ピクセルの投影画像である。さらに上記に光輸送行列Tを効率的に格納する方法、および光輸送行列Tを生成するのに必要な取り込み画像数をpおよびqの乗算からpおよびqの合計またはそれ以下に減らす方法が説明される。さらに2つの取り込み画像から光輸送行列Tの各列を次の関係、

30

【0288】

【数38】

$$Tj \approx \text{MIN}(C_{S1}, C_{S2})$$

40

によりどのように合成するか示され、ここで、列 T_j は光輸送行列Tのj番目の列、第1組 S_1 および第2組 S_2 は2組のピクセルで $S_1 \cup S_2 = \{j\}$ 、および C_{S1}, C_{S2} は光輸送行列Tの各々の列の合計、すなわち $C_S = \sum_j S T_j$ である。

【0289】

関係 $c = T p$ は投影画像pが場面に投影された時にカメラがcとして何を取り込むかの予測を可能にする。特定のcに対し p を見出すには次の逆方程式、

【0290】

50

【数39】

$$p = T^{-1} * c$$

が必要となる。

【0291】

光輸送行列 T は非常に大きい行列であるので、その逆光輸送行列 T^{-1} を演算するのは一般的に非現実的である。従って、上記ではさらにほとんどの実際的な状況に適した逆光輸送行列 T^{-1} の近似を計算する方法が示される。逆光輸送行列 T^{-1} の近似はまず光輸送行列 T の各列はプロジェクタからの 1 ピクセルの活性化によりもたらされる投影画像であることを念頭に置き、逆光輸送行列 T^{-1} の近似を求めるためには、

10

【0292】

【数40】

$$\Xi_r = Tr / (\|Tr\|)^2 \quad \cdots \cdots \text{方程式 (1)}$$

但し、 $r = 1, 2, 3, \dots, pq$

あるような行列 T を構築でき、列 r は行列 T の r 番目の列である。上記により詳細に説明されたように、

20

【0293】

【数41】

$$(\Xi_r^T) * (Tr) = I$$

および、

【0294】

【数42】

$$(\Xi_r^T) * (T\omega) = 0 \quad r \neq \omega \text{ の場合}$$

30

であるので、これは、

【0295】

【数43】

$$\Xi^T = T^{-1}$$

を意味する。従って、関係 $= T p$ は $p = (T^{-1}) * c$ と整理することができ、これは次のように近似することができる。

40

【0296】

【数44】

$$p \approx (\Xi^T) * c$$

【0297】

上述のように、 T は本考察において「ビュー投影行列」と呼ばれる。ビュー投影行列 T の意味合いは、望ましい眺め c が与えられた場合、投影画像 p は基準カメラ（望ましい眺め c の画像を作成した）の位置から眺めると望ましい眺め c と同じ外観を有する場面を実現するような投影画像 $p = (T^T) c$ を見出す（すなわち作成または生成する）ということである。

【0298】

50

ここで「ビュー投影行列」の実施を最適化するという問題が持ち上がる。懸案の問題は消費者レベルのパソコンもしくはラップトップ／ノートブック・コンピュータ、またはプロジェクトシステムの統合演算能力上でさえも消費者レベル演算システムにおける実施を促進するために「ビュー投影行列」の実行をどのように最適化するかということである。本発明は消費者レベルにおける演算アーキテクチャの最近の改良を利用しようとするものである。

【0299】

並列処理能力のある機械はより普及しており、本発明はこれらの並列処理機械を利用しようとするものである。より具体的に、本発明は多くの消費者レベルの演算装置に普及している特定サブシステムの並列処理能力の改良をうまく利用しようとするものである。

10

【0300】

コンピュータ・プログラムは通常幅広く多様な命令のセットに従い多数の異なったタスクを実施できる汎用コンピュータ、すなわち非特殊コンピュータ、の中心をなす演算システムの主C P U（中央演算装置）内で実施される。

【0301】

しかし、演算システムはデータ・トラフィック制御、音声処理、ビデオ処理、メモリ・アクセス制御、等々具体的なデータ処理タスクを実施する他の特化サブシステムを有する。これらの特化サブシステムは演算システムの主ボード上の特化集積回路として、または演算システムの主ボードに取り付けられるドーター・ボードとして実装できる。本発明の一特色はビュー投影行列[†]のデータ処理の要求をすべて、または一部、演算システムの主C P Uから少なくとも演算システムの特化サブシステムに移すことである。より正確には本発明は1つの具体的な特化サブシステムの能力を利用する。

20

【0302】

演算システム内の特化サブシステムの内、ビデオ処理サブシステムは近年大きく進歩しており、演算システムの主C P Uに匹敵するような処理能力に到達している。これは並列処理の分野に特に当てはまる。C P Uは汎用処理装置として設計されており、従って、幅広い、かつ自由度のある命令セットを支援しなければならないため、いっしょにして並列処理装置を形成できる小数の複雑な処理装置、または処理コアに制限されている。このような汎用C P Uはそのアーキテクチャにより比較的少数の処理コアに制限される。つまり、並列処理の汎用C P Uは大きな命令セットを実施できるよう通常M I N D（複数命令、複数データ）アーキテクチャに従っているが、自由度の代償は処理コアの数が非常に制限されることである。例えば、現在汎用C P Uは約2つまたは4つの処理コアに制限される。逆に、ビデオ処理サブシステム（しばしばビデオ・カード、またはビデオ処理ドーター・ボードに統合される）は1つの一次タスク、すなわちビデオ処理、を実施することに特化されているので、命令セットはより特化され、限定されている。G P Uは通常並列処理装置により多数の簡略化された（すなわちより小さい）処理コアを統合できるS I M D（単一命令、複数データ）アーキテクチャに従っている。G P Uは限られた命令セットを有するが、G P Uはしばしばスレッドと呼ばれる命令を並列に多数処理する能力がある。このようなG P Uは何百もの処理コアを有することができる。

30

【0303】

本発明は並列処理に最適化されたデータ構造を提供し、他の機能を最適化して特に高度並列処理システムに適したことによりビデオ・カードにおける並列処理の改良を利用しようとしている。このように、通常C P Uの負担となるような処理要件をコンピュータシステムのビデオ・カードにオフロードすることができる。これはビュー投影行列の実施の多くは画像に関係しているので特に役立ち、従って、高度並列演算ビデオ・カードの特化アーキテクチャに良く適している。

40

【0304】

本発明は一般的に並列処理システムに適しているが、説明の容易さから本明細書では並列処理サブシステムの具体的な例に適用して説明される。つまり、本発明は演算システムのC P Uに加え、またはそれに代わり、演算システムのG P Uを用いることによりビュー

50

投影行列^Tの実施を加速させる。各種のGPUアーキテクチャが利用できるが、NVIDIA（登録商標）、8800GTXの製品群のGPUはCUDA（登録商標）（Compute Unified Device Architecture）と呼ばれるアーキテクチャを通じて並列演算に対する系統だったサポートを有しているので本例示的実施はこれらを使用している。「NVIDIA（登録商標） CUDA（登録商標），Computer Unified Device Architecture，Programming Guide」、2.0版に説明されるようにCUDA（登録商標）はCプログラミング言語に対する小さな拡張を通じてプログラミング・アクセスをユーザに提供しており、同ガイドは引用により本明細書に全体として組み入れられる。以下の考察において、プログラミング用語、キーワード、またはソフトウェア構造に関する言及はCプログラミング言語に関するとみなされる。

10

【0305】

NVIDIA（登録商標）、CUDA（登録商標）を用いて「ビュー投影」演算を実施することは多くの特殊な気遣いを必要とする。CUDA（登録商標）はマルチコア・ハードウェア・プラットフォームで、多数の処理装置を有する（例えばFX 8800、GPUは128個のハードウェア処理装置、またはコア、を有し、NVIDIA（登録商標）GT200は270個のコアを提供する）。各処理装置、またはコアは専用のプライベート・メモリを有しており、複数の処理装置が各々のグループ・メモリを共有するスレッドアレイのブロックにグループ化されるが、処理装置のすべては同じグローバル・メモリも共有する。従って、メモリ・トラフィックは性能の大きな障害となっている。

20

【0306】

各CUDA（登録商標）処理装置内において、命令のパイプライン化を用いて複数の命令スレッドの実行を加速する。複数のスレッドは同じ開始点を有することができるが、実行中に分かれ得る。さらに、正しいパイプラインをその後の実行において維持するためには、自分のスレッドのブロック内で他のスレッドの前に終了するスレッドは正しい順の実行を維持するためすべてのスレッドが実行を終了するまで待たされる。従って、命令スレッドの実行の不一致を最小限にし、パイプライン実行におけるシリアル化遅延を縮めるためにビュー投影行列のデータ構造を整理する必要がある。

【0307】

対応すべき第1の問題はアドレスの不連續性をいかに最小限にするかである。複数のカメラピクセルが1つの個別のプロジェクタピクセルに関連しているかもしれない。この基本的な例が図3および図6に示され、ここでカメラのピクセル解像度はプロジェクタのそれより高く、1つのプロジェクタピクセルに囲まれる（すなわち関連する）カメラピクセルの光ファットプリント（またはクラスタ）をもたらす。複数のカメラピクセルが単一のプロジェクタピクセルに関連する（カメラまたはプロジェクタのピクセル解像度に関係なく）別の理由は光の流れにより生じる歪みの種類と背景または場面の形状である。これらの歪みの種類によりいくつかのプロジェクタピクセルが複数のカメラピクセルに対応する可能性ができ、あるいは他のプロジェクタピクセルが単一のカメラピクセルに対応する可能性ができ、またはプロジェクタピクセルが対応する（すなわち関連する）カメラピクセルを有しないことすらあり得る。

30

【0308】

図47はビュー投影行列^Tの直観的ではあるが、非効率的なデータ構造を示す。各投影画像ピクセル（すなわちプロジェクタピクセル）に対し各具体的投影画像ピクセルに関連するカメラピクセルの数（intcount）および各関連カメラピクセルに関する具体的な情報を格納するソフトウェア・データ構造300_0～300_(mn-1)がある（すなわち「Element」の種類はCプログラミング言語における「struct」）。各関連カメラピクセルは関連カメラピクセルの各々位置座標（311）およびRGB色値（312～314）を指す4つのメモリポインタ301～304により定義される。

40

【0309】

理解されるように、ピクセルの較正データはファイルに格納され、アプリケーションが開始すると、較正データは定義されたデータ構造に従いメモリにロードされる。従って、

50

アプリケーションが開始すると、各投影画像ピクセル $300_0 \sim 300_ (mn - 1)$ に対し、その関連カメラピクセル各々について4つのメモリポインタ $301 \sim 304$ の別個のセットが、関連カメラピクセルの各々について4つの対応する動的にマロックされた（すなわちメモリ割り当てされた）バッファ $311 \sim 314$ とともに作成される。さらに、アドレス・アクセス・パターンがランダムで不連続であるためデータは非常に非効率的にロードされる。従って、このデータ構造は論理的に理解し易いが、メモリ・アクセスの操作の観点から非常に非効率的である。各投影画像ピクセルをアクセスするには、関連カメラピクセル毎に4つの異なったバッファが関わり、各バッファは物理的メモリの異なった部分にあるかもしれない。

【0310】

10

従って、この方法はDRAMメモリ（すなわちDDRDRAM、DDR SDRAM、等々）のバースト・モードをうまく活用していない。不連続な形でメモリをアクセスすることはDRAM動作においてバンクや列を変更する時間のため、バースト・モードで連続的アドレス位置のメモリをアクセスするよりはるかに遅くなり得る。

【0311】

アドレスの不連続性を最小限にする最初の工程は従って、データへのアクセスができるだけ連続的な形でアクセスできるよう図47のデータ構造を再編成することである。もたらされる、改良されたデータ構造400が図48に示されるが、動的にマロックされたバッファが関わらなくても良い。データ構造400は3つのバッファとしての第1バッファ401、第2バッファ402、第3バッファ403に格納される。「Count_Loc_Buffer」と呼ばれる第1バッファ401はプロジェクトピクセル、または要素、毎の関連カメラピクセルの総数（すなわち数）および第1カメラピクセルの位置を格納する。この方法で、プログラムは特定のプロジェクトピクセルに関連する第1カメラピクセルをどこで見つけるかが分かり、関連カメラピクセルは連続的に配列されているので、プログラムは第1カメラピクセルからいくつの後続カメラピクセルが同じくその特定のプロジェクトピクセルに関連しているかが分かる。第2バッファ402の「Coords_Buffer」は関連カメラピクセルのX-Y座標を格納し、第3バッファ403の「RGB_Buffer」は関連カメラピクセルのRGB値を格納する。

20

【0312】

30

例えば、アプリケーションがプロジェクトピクセル0から開始し、次にプロジェクトピクセル1に進み、次にプロジェクトピクセル2に、次にプロジェクトピクセル3に、以下同様であると仮定する。さらにプロジェクトピクセル0は4個の関連カメラピクセルを有すると仮定すると、これらの関連カメラピクセルはアドレス位置（0, 1, 2, 3）に格納されることになる。従って、count(0) = 4（4個の連続的に格納された関連カメラピクセルがプロジェクトピクセル0に関連していることを示す）で、loc(0) = 0（プロジェクトピクセル0に関連する関連カメラピクセルの第1のピクセルはメモリ位置0に格納されていることを示す）である。次にプロジェクトピクセル1が3個の関連カメラピクセルを有すると仮定し、これらはプロジェクトピクセル0に関連する4個のカメラピクセルに連続的に続くメモリに格納されることになる。従って、これらの3個の関連カメラピクセルはアドレス位置（4, 5, 6）に格納されることになる。従って、count(1) = 3（プロジェクトピクセル1が3個の関連カメラピクセルを有することを示す）で、loc(1) = 4（プロジェクトピクセル1に関連する第1カメラピクセルがメモリ位置4に位置することを示す）である。ここでプロジェクトピクセル2が100個の関連カメラピクセルを有すると仮定すると、これらのカメラピクセルはアドレス7, 8, 9, ..., 106に格納されることになる。従って、count(2) = 100およびloc(2) = 7で、これはプロジェクトピクセル2に関連する第1カメラピクセルの開始アドレスである。プロジェクトピクセル3, 4、以下同様についてこのパターンに従う。この配列を用い、データは大部分連続的な順序でアクセスされ、DRAMメモリのバンク/列変更の時間を最小限にする。

40

【0313】

50

関連カメラピクセルに関する情報のメモリ位置（すなわちメモリ・アドレス）は第2バッファ402のC o o r d s _ B u f f e rと第3バッファ403のR G B _ B u f f e rと同じであることも特記される。これは各関連カメラピクセルについての情報を保持するデータ単位のサイズを知る（すなわち選択する）ことにより達成される。例えば、第2バッファ402のC o o r d s _ B u f f e rにおいて、関連カメラピクセル毎の情報は2つの部分（XおよびYの座標情報）を含むが、単一のデータ単位として格納される。いくつのデータ単位を飛び越すかを知ることにより正しいカメラピクセル情報にジャンプすることができ、データ単位の数は読み取りたい関連カメラピクセルの数により第1バッファ401のi n t l o c（）により定義される開始点からのオフセットとして判定される。同様に、第3バッファ403のR G B _ B u f f e rにおいて関連カメラピクセル毎の情報は3つの部分を含むが（赤、緑、および青の情報）、1つの単位として格納されるので、ここでもいくつのピクセルデータを飛び越すかを知ることにより望まれるカメラピクセル情報にジャンプすることができる。このように第1バッファ401のC o u n t _ L o c _ B u f f e rに格納された位置情報（例えばl o c_（0）またはh l o c_（1）等々）は第2バッファ402のC o o r d s _ B u f f e rおよび第3バッファ403のR G B _ B u f f e rの双方に適用できる。基本的に、第1バッファ401のC o u n t _ L o c _ B u f f e rは各プロジェクトピクセルに対し関連カメラピクセルの総数およびプロジェクトピクセル毎の第1関連カメラ画像ピクセルの相対位置を格納し、これは他の2つバッファ、第2バッファ402のC o o r d s _ B u f f e rおよび第3バッファ403のR G B _ B u f f e r各々に対し同じである。

10

20

30

40

【0314】

このように操作において、3つの大きなバッファ（第1バッファ401、第2バッファ402、および第3バッファ403）があり、互いに分離している。しかし、「ビュー投影行列」を処理する時、これらの3つの大きなバッファへのアクセスは各々連続的である。これによりキャッシュにあまり負担をかけないシステムとなる。キャッシュ機構の場合、2つの大きなバッファ（b u f f e r _ # 1 およびb u f f e r _ # 2）があってb u f f e r _ # 1の第1バイトを読み取ると、第1バイトと併せてデータ連続的グループ全体（例えば64バイトのサイズの）が読み取られる。従って、本例において、b u f f e r _ # 1の最初の64バイトが第1バイトと併せてキャッシュにロードされる。これはしばしば「バースト・モード」と呼ばれる。次にb u f f e r _ # 2の第1バイトを読み取ると、b u f f e r _ # 2の最初の64バイトもロードされることになる。システムがb u f f e r _ # 1の第2バイトを読み取るために戻ると、それは既にキャッシュに入っているので、システムD R A Mへのアクセスは必要ない（すなわちS D R A M読み取り操作が必要とされない）。これは「キャッシュ・ヒット」と呼ばれる。これに続き、システムはb u f f e r _ # 2の第2バイトを要求し、このバイトも既にキャッシュにがあるので、再びキャッシュ・ヒットがある。従って、キャッシュが十分に大きく、何万もの別々のバッファ（図47の構造で必要となるような）を避けねば、大部分キャッシュ・ヒットになる。本例の場合、図48の構造は図47の構造に比べバッファの数を何万から3つだけのバッファに削減する。

50

【0315】

図48の別の利点は、アプリケーションが開始してビュー投影行列がファイルからロードされる度にコンピュータ・システムが較正データを1つ1つ個別の小さなデータ構造および個別のバッファに解析する必要がないことである。これによりロード時間を10分の1に削減することができる。本発明をテストするために作成されたサンプル・システムでは、ロード時間が約30秒から3秒に削減された。

50

【0316】

本発明の最適化に向けた次の工程はカーネル実行の不一致を避けることである。上記に説明のとおり、G P Uは小さな処理装置（すなわちコアまたはプロセッサ）を多数有しており、これらのプロセッサはプロセッサのブロックに分けられる。各ブロックはS I M D（単一命令、複数データ）アーキテクチャで、すべてのプロセッサが類似したタスク（す

なわちスレッド)を実行すると並列性が維持される。プロセッサの1つのブロック内におけるいすれかのスレッドが異なると、他のスレッドと分かれ、並列性は失われる。さらに、スレッドの同期化を維持するために他のスレッドは分かれたスレッドの終了まで待つ。

【0317】

本アプリケーションにおいて、タスクの多く(例えば行列の乗算)は類似しているが、異なった(プロジェクタピクセル) / (カメラピクセル)組み合わせに適用される。さらに、各プロジェクタピクセルは異なる数の関連カメラピクセルを有するかもしれないので、スレッドの長さが異なる。スレッド間のこの長さの差はプロセッサの1ブロック内で不一致および同期化遅延(すなわちシリアル化遅延)問題を起こす。この問題は似通った数の関連カメラピクセルを有するプロジェクタピクセルが同じCUDA(登録商標)スレッド・ブロックにグループ化されるよう投影行列を整理し直すことにより最小限にすることができる。基本的に、プロジェクタピクセルは各々の関連カメラピクセル総数(すなわち図48の第1バッファ401のCount_Loc_Bufferにおける入力「count()」)により共通のCUDA(登録商標)スレッド・ブロックにグループ化される。この方法で、カーネル実行の不一致は避けられ、各スレッド・ブロックの実行が最適化される。

【0318】

基本的に、CUDA(登録商標)(および他のマルチプロセッサ全般)は1つのスレッド・ブロックを同じ処理装置、すなわちコア、にパイプライン状にスケジュールする。何らかの流れ制御が実行の不一致を生じさせると、CUDA(登録商標)はこれらのスレッドが同じ実行点に再度到達するまでシリアル化する。「ビュー投影行列」の較正データは異なるプロジェクタピクセルに対応する異なる数の関連カメラピクセルを有するかもしれないため、この問題を扱うには特別な注意が払われる。

【0319】

図49(a)～図49(b)は本発明で取られる方法を図示する。図49(a)はプロジェクタ・カメラシステム内で関連カメラピクセルの数が1つのプロジェクタピクセルから次にかけ異なり得ることを示すグラフである。グラフの横軸は各プロジェクタピクセルを番号で特定し、縦軸は個々のプロジェクタピクセル毎の関連カメラピクセルの数(すなわち関連カメラピクセル総数)を示す。図49(b)において、1つのプロジェクタピクセルから次にかけ関連カメラピクセル総数が滑らかな増加の形で変化するよう図49(a)のデータが並べ替えられている。

【0320】

図49(c)において、プロジェクタピクセル毎の関連カメラピクセルの数(すなわち関連カメラピクセル総数)は定形数の量子化レベル501に量子化される。特定された量子化レベル内の関連カメラピクセル総数を有するプロジェクタピクセルはプロジェクタピクセルグループにまとめられる。本発明は共通のプロジェクタピクセルグループ内のすべてのプロジェクタピクセルについて関連カメラピクセルの数を同じにすることを求める。同じ数の関連カメラピクセルを有するプロジェクタピクセルは次に共通のCUDA(登録商標)スレッド・ブロックに割り当てられ、同じスレッド・ブロック内で不一致がないことを確実にする。共通のプロジェクタピクセルグループ内のプロジェクタピクセルすべてが同じ数の関連カメラピクセルを有することを確保するために、プロジェクタピクセルグループ内のプロジェクタピクセルである量子化レベルに引き上げるべく関連カメラピクセル総数を変える必要があるものにはゼロ値の存在しないカメラピクセルが追加される。

【0321】

このように、あるスレッド・ブロック内のすべてのスレッドが等しい数のカメラピクセルを処理することになり、この数はスレッド・ブロック内の最大関連カメラピクセル総数に等しくされる。いすれかのプロジェクタピクセルの、関連スレッド・ブロック内の最大関連カメラピクセル総数より低い関連カメラピクセル総数はゼロ値のカメラピクセル情報(すなわちダンプピクセル)で水増しされ、より低い関連カメラピクセル総数を最大関連カメラピクセル総数に等しくなるよう引き上げる。

【0322】

例えば、1つのスレッド・ブロック内で1つのプロジェクトピクセルが100個の関連カメラピクセルを有し、同じスレッド・ブロック内における他のプロジェクトピクセルはすべて30個の関連カメラピクセルしか有しない場合、70個のダンプピクセル（すなわちRGB値がゼロに設定されたカメラピクセル）が30個の関連カメラピクセルを有するプロジェクトピクセルに追加して関連カメラピクセル総数（すなわち図49におけるentry count（））を「100」に引き上げる。このように、あるスレッド・ブロック内のすべてのスレッドが同じタスクを等しい回数実行し、不一致は生じない。

【0323】

これを図的に表したものを見50に示す。本例は5個のプロジェクトピクセル601～605からなるプロジェクトピクセルグループを示す。各プロジェクトピクセル601～605について、対応する関連カメラピクセル611～615が図的に順次配列され、暗いボックスで表され、非ゼロ（NZ）のRGB値を有する。

10

【0324】

このサンプルのプロジェクトピクセルグループの最大関連カメラピクセル総数は「10」である。この関連カメラピクセル総数値はプロジェクトピクセル602, 605により共有される。しかし、残りの3個のプロジェクトピクセル601, 603, 604は10未満の関連カメラピクセル総数を有する。プロジェクトピクセル601, 603, 604は各々1、5、および3の初期関連カメラピクセル総数値を有する。プロジェクトピクセル601, 603, 604の各々の関連カメラピクセル総数を最大総数値の「10」に引き上げるために、プロジェクトピクセル601, 603, 604の各々に点線の箱で表されるように異なった数のダンプピクセルが追加され、RGB値はゼロに設定されている。プロジェクトピクセル601は9個のダンプピクセル（またはダミーピクセル、すなわちRGB値がゼロに設定された架空のカメラピクセル）が元の1個の関連カメラピクセルに追加され、総数値を「10」に引き上げられている。プロジェクトピクセル603は5個のダンプピクセルが元の5個に追加され、「10」の総数値が作成されている。プロジェクトピクセル604は7個のダンプピクセルが元の3個の関連カメラピクセルに追加されている。

20

【0325】

別の特色は装置-ホストのコピー（すなわちメモリ・アクセス）操作を削減することにより本発明の実行を加速する。これはPBOバッファ（OpenGL Pixel Buffer Object）をCUDA（登録商標）カーネルの結果バッファとして割り当てることにより達成される。これにより、CUDA（登録商標）カーネルの結果は表示用にホストPC（すなわちCPU）に転送して戻す必要がなくなる。代わりに、CUDA（登録商標）カーネルの結果はそれをCUDA（登録商標）のグラフィックス・アーキテクチャにより直接表示するOpenGLテクスチャ・メモリにマップされる。

30

【0326】

前述のように、グローバル・メモリへのアクセスは時間およびリソースの点で特に高価であるので、本発明の別の特色は利用可能なグローバル・メモリのバス幅をより良く利用することにより必要なグローバル・メモリ（すなわち主メモリ）へのアクセス要求を削減する。キャッシュされないグローバル・メモリ・アクセスは共有メモリまたはレジスタ・メモリ（すなわちプロセッサ・コアにおけるローカルなメモリ）へのメモリ・アクセスに比べ約100から200倍高価で（すなわち遅く）あり得る。本ビュー投影演算アプリケーションのようなデータ・トラフィック集約型アプリケーションでは、グローバル・メモリ・アクセスは性能上大きな障壁となる。

40

【0327】

図51に関し、本例では説明を簡単にするためグローバルメモリは主メモリ701と特定され、処理コアの共有メモリおよび/またはレジスタメモリは集合的にローカルメモリ703と特定される。CUDA（登録商標）の処理装置はグローバルメモリとしての主メモリ701を命令毎に128ビットまでの幅でアクセスでき、これはワードまたはロング

50

・ワードのビットサイズ（すなわち8ビットから32ビット）を有する典型的なデータの断片（すなわちデータム）に比べ数倍幅広い。これは個々のRGB値が通常幅1バイト、すなわち8ビット、である本件の場合特に当てはまる。しかし利用できるバスサイズを活かすために、本発明は利用可能な最大幅のデータサイズ、すなわち本例の場合は128ビット、でグローバル・メモリをアクセスする。この場合、実際のデータ・タイプに拘わらず、128ビットのバスサイズを充分に活用するため必ず利用可能なバス幅のサイズであるfloat4またはint4のデータ・タイプを使用する。このように、主メモリのデータ単位（「主データム」）は128ビット単位711～714で定義される。従って、正しいデータ収集はグローバル・メモリからレジスタ（または共有メモリ）にデータが128ビットロードされると得られる。基本的に、データはグローバル・メモリからレジスタ（または共有メモリ）に各々128ビットの塊721～724としてロードされ、これらは複数の個別的数据値を含むことができる。例えば、塊721は4個のデータム値、Local Datum 0～Local Datum 3、を含む。処理コア705が次にレジスタ（または共有メモリ）かた個々のローカル・データム値（Local Datum 0～Local Datum 15）を実際のデータム値により決まるビットサイズでアクセスできる。10

【0328】

本発明の別の特色はRGBのアドレス指定方式を変えることによりRGBアクセスを最適化する。殆どのビデオ・フォーマットはRGBピクセル・フォーマットの出力を実現し、これはピクセル当たり3バイト、または3バイトピクセル、を実現する。しかし3バイトピクセルは通常の物理的メモリの通常のアドレス指定方式と互換性がない。物理的メモリのアドレス指定方式は通常4バイトに位置合わせされる（すなわち4バイトの区分でアドレス指定される）。従って、3バイトピクセルはCUDAの最大データ・フェッチ能力が16バイト（すなわち128ビットの幅）であるので、これとも互換性がない。加えて、CUDAのテクスチャ・エンジンは3バイトピクセル・フォーマットには使用できない。20

【0329】

この問題に取り組む通常の方法は3バイトのRGB（赤、緑、青）フォーマットを4バイトRGBA（赤、緑、青、アルファ）フォーマットに変換することである。ビデオは一般的に記憶空間を節約するためにRGBフォーマット（すなわちピクセル当たり4バイトではなくピクセル当たり3バイト）で作成され、システムのCPUはビデオのRGBフォーマットを処理用にRGBAフォーマットに変換することが要求される。本発明はこのRGBからRGBAへの変換を避けるとともに3バイトのRGBフォーマットと4バイトで位置合わせの物理的メモリとの互換性を確保している。これは3バイトのデータを4バイトの空間にマップすることにより達成される。30

【0330】

図52はRGBフォーマットをRGBAフォーマットと同じように速く用いることができるることを図示する。RGBフォーマット（ピクセル当たり3バイト入力画像データには変更を加えない。むしろ、RGB画像はCUDAテクスチャ・エンジン（すなわちグローバル・メモリの特殊部分）にマップされ、これは通常ピクセル当たり16バイトであるint4タイプの画像と解釈される。しかし、グローバル・メモリに書き込まれるピクセルデータは実際には3バイトであるので、int4データム単位毎に51/3のRGBピクセルを当てはめることができる。従って、真のRGBAデータを格納するとした場合int4データム単位当たり4個のRGBAピクセルしか当てはめられないので、利用可能なメモリ空間をより効率的に使用することになる。問題は個々の3バイトのRGB情報を4intメモリ空間からアクセスするかということである。40

【0331】

これは簡単なアドレス転換により達成される。図52における1行のデータの各int4セグメント内で、ピクセル情報は3つの連続バイト（すなわちRGBデータ）のグループに集められていることが見られる。望まれるピクセル情報が存在する適切なint4セ50

グメントを特定するためには合計ピクセル・オフセットを 16 バイト (すなわち `int` 4 セグメント当たり 16 バイトがある) で割る必要がある。例えば、サイズ (`w` , `h`) (すなわち幅 `w` ピクセルで高さ `h` ピクセル) の RGB 画像はサイズ ([(3 バイト * `w`) / (16 バイト)] , `h`) の `int` 4 画像としてアドレス指定される。この方法は `w` が「 16 」の倍数である限り自然数かつ正確な結果をもたらす。利用可能な画像ソースは殆ど「 16 」に位置合わせされた (すなわち「 16 」の倍数の) フレームを提供するのでこれは問題でない。

【 0 3 3 2 】

この処置の別の利点は床および天井ピクセルが隣接する `int` 4 セグメントと位置合わせされる可能性が非常に高いことであり、これは画像のズーム操作を実行する場合などに必要となるピクセル間で画像情報を補間するのに必要なメモリ・フェッチ操作の数を非常に削減する。

10

【 0 3 3 3 】

これは 2D 線形補間の際、`int` 4 画像データからどのように正しい RGB ピクセルをフェッチできるかを説明することにより示すことができる。位置 (`float x` , `float y`) におけるピクセル値を補間したい場合、その周囲の整数位置 (`floor(x)` , `floor(y)`) , (`ceil(x)` , `floor(y)`) , (`floor(x)` , `ceil(y)`) , (`ceil(x)` , `ceil(y)`) における 4 個のピクセルからのピクセル値が必要となる。例えば位置 (`x = 35.4` , `y = 54.7`) におけるピクセルの値を知りたい場合、周囲の 4 つの整数位置 (35 , 54) 、整数位置 (36 , 54) 、整数位置 (35 , 55) 、および整数位置 (36 , 55) からこの値を (双一次補間を用い) 補間することができる。ピクセル情報は 128 ビットのグループに集められているので、周囲のピクセル情報は 2 つのフェッチ操作内でアクセスできる可能性が高い。基本的に双一次補間のためには 4 個のピクセル値をアクセスする必要がある。4 個のピクセルに対する望まれる情報を 2 回だけのメモリ・アクセス操作で得る可能性は 11 / 16 であることが見出されている。ということは 4 個の望まれるピクセル値すべてを 4 回メモリをアクセスすることにより得る可能性は 5 / 16 ということになる。この 3 バイトピクセルから `int` 4 メモリ・マッピングを行わない場合、4 個の望まれるピクセル各々に対し別々のメモリ・アクセス操作が必要となり、従って、必ず 4 回メモリをアクセスしなくてはならないことになる。このように、本方法は CPU による明白な RGB から RGBA への変換を避けると同時に双一次補間の効率も向上させる。

20

【 0 3 3 4 】

本実施の別の特色は共有メモリを 1 つのスレッド・ブロックにおける各スレッドに注意深くアロックする (メモリの空間を割り当てる) ことによりメモリ・バンク間の衝突を避けることである。CUDA (登録商標) のチップ上の共有メモリは同じスレッド・ブロックに所属するスレッド間でバンク衝突がないと非常に能率を上げ得る。従って、スレッド・ブロックに対するバッファ (すなわちバンク) が必要となる場合、各スレッドからのアクセスは注意深く割り当てられ、半ワープ (16 スレッドのグループと定義される) 内で 2 つのスレッドが同じバンクをアクセスすることができないようにする。このように、共有メモリへのアクセスは各々 2 つの CUDA (登録商標) システム・クロックしかかかるない。

30

【 0 3 3 5 】

本発明の別の特色は CUDA のテクスチャ・エンジンを充分に活用し、より良いメモリ・アクセス性能を達成することである。グラフィック・カード (すなわちビデオ・カード) のテクスチャ・エンジンはデータ・キャッシングおよびユーザが提供するピクセル・アドレスおよびピクセル・フォーマットに基づく自動メモリ・アドレス演算を利用して画像データ・アクセスを加速する専用ハードウェアである。テクスチャ・エンジンはピクセル位置が範囲外であるなどいくつかのエラーの場合を自動的に対処することもできる。従つて、ソフトウェアがこれらの問題に関わる必要がない。

40

【 図面の簡単な説明 】

50

【0336】

【図1】(a)および(b)は、デュアル・フォトグラフィを実施するための従来の構成を示す。

【図2】(a)および(b)は、本発明によるデュアル・フォトグラフィの構成を示す。

【図3】1つのプロジェクタピクセルから投射される光線が場面から反射されカメラピクセルアレイに小さな光フットプリントを形成する例を示す。

【図4】映像が情報のアレイとしてどのように表示され得るかの例を示す。

【図5】カメラピクセルアレイに取り込まれた光フットプリントが情報のアレイとしてどのように表示され得るかの例を示す。

【図6】プロジェクタにおける1つのプロジェクタピクセルの活性化によりもたらされるデジタルカメラ内の光センサアレイに投影されるフットプリントを図示した例である。

【図7】図3の例を反映した、光輸送行列T内における光輸送係数の列を図示した例である。

【図8】(a)および(b)は、市松格子パターンを有する場面に同時に投影される2列のプロジェクタピクセルの2つの例を示す。

【図9】(a)および(b)は、市松格子パターンを有する場面に同時に投影される2行のプロジェクタピクセルの2つの例を示す。

【図10】(a)は、図6および図8(a)の画像を組み合わせることにより作成される、1つのプロジェクタピクセルによりもたらされる発生光線のフットプリントを示し、(b)は、図7および図8(b)の画像を組み合わせることにより作成される、1つのプロジェクタピクセルによりもたらされる発生光線のフットプリントを示す。

【図11】プロジェクタピクセルを図9(a)または図9(b)で発生された光線で決定される非ゼロ値の光輸送係数に結び付けたインデックスの第1例である。

【図12】プロジェクタピクセルを図9(a)または図9(b)で発生された光線で決定される非ゼロ値の光輸送係数に結び付けたインデックスの第2例である。

【図13】(a)は、実カメラで撮影した実取り込み画像を示し、(b)は、実プロジェクタから見た、本発明の方法を用いて生成されたデュアル取り込み画像を示す。

【図14】プロジェクタを較正するためにホモグラフィの使用を適用した結果の例である。

【図15】プロジェクタと場面の間に置かれた2つのワイングラスにより歪められた映像を示す。

【図16】2つの別個のプロジェクタピクセルにより実現された2つの重なり合った隣接光フットプリントの例である。

【図17】本発明による、任意の場面に表示制約を課すためのプロセスの例である。

【図18】図17に示された表示制約におけるさらなる工程である。

【図19】本発明がポスター画像の色補正に適用される例である。

【図20】図15の映写場面で、本発明の光歪みを補正する方法を適用した状態を示す。

【図21】没入型表示システムの作成にデュアル・フォトグラフィを用いる典型的な映写設定である。

【図22】没入型プロジェクタP2が仮想プロジェクタP1"をシミュレートするために用いられるところを示す。

【図23】(a)は、実前面投射型プロジェクタにより投影された画像の右側を示し、(b)は、(a)に示す画像の左側を示し((b)においては画像の左側は没入型プロジェクタにより投影される)、(c)は、(a)の右側画像を(b)の左側画像に接合したところを示す。

【図24】さらに没入型プロジェクタにより生成された左側の画像を前面投射型プロジェクタにより生成された右側の画像を接合した例を示す。

【図25】(a)～(c)は、実際の部屋の中に仮想モデル部屋で作り出された仮想画像を再現した、本発明の別の適用を示す。

【図26】本発明の最小限の形態に従った典型的映写システムである。

10

20

30

40

50

【図27】図26の設計に基づくプロトタイプを示す。

【図28】図27の構成を別の角度から示す。

【図29】(a)は、図27および図28の映写システムを設置した部屋を周辺光の下で示し、(b)は、(a)の部屋を没入型映写照明の下で示す。

【図30】(a)は、較正されていない画像を投影した図29(a)の部屋を示し、(b)は、較正された画像を投影した図29(a)の部屋を示す。

【図31】(a)は、較正されていない空白の映像を投影した図29(a)の部屋を示し、(b)は、較正された空白の映像を投影した図29(a)の部屋を示す。

【図32】本発明をドーム型ミラー・プロジェクタに適用する第1工程である。

【図33】本発明をドーム型ミラー・プロジェクタに適用する第2工程である。

【図34】本発明をドーム型ミラー・プロジェクタに適用する第3工程である。

【図35】ドーム型ミラー・プロジェクタの投影用に正確に歪められた画像を実現するために本発明を望ましい画像に適用したところを示す。

【図36】図35で生成された歪んだ画像を投影した結果を示す。

【図37】天井据付運転の別の設計である。

【図38】本発明の別の構成である。

【図39】本発明のさらに別の構成である。

【図40】2つ以上のプロジェクタ-カメラシステムをモザイク化して合成画像を作成する設定を示す。

【図41】複数のプロジェクタ-カメラシステムによる画像の合成を示す。

【図42】マルチプロジェクタ-カメラシステムにおける2つのプロジェクタ-カメラシステムの内1つ目により投影された画像を示す。

【図43】マルチプロジェクタ-カメラシステムにおける2つのプロジェクタ-カメラシステムの内2つ目により投影された画像を示す。

【図44】複雑な表面上に市松格子パターンを投影することに本発明を適用した例である。

【図45】(a)は、1つの曲面ミラー125および複数のプロジェクタ-カメラ対145を使用する設計を示し、(b)は、1つのミラーピラミッド151および複数のプロジェクタ-カメラ対145を使用する設計を示す。

【図46】全体により大きい映写FOVを達成するために複数の大きいFOVプロジェクタ153a, 153bを用い得ることを示す。

【図47】ソフトウェア・アプリケーション実行の際、ビュー投影行列、すなわち較正データまたは情報、をメモリにロードするための直観的ではあるが、最適化されていないデータ構造を示す。

【図48】ソフトウェア・アプリケーション実行の際、メモリに較正情報を配列するための最適化されたデータ構造を示す。

【図49】(a)は、較正データ(例えばビュー投影行列からのなど)個別のプロジェクタピクセルに関連するカメラピクセル数を表す生データを示し、(b)は、(a)の情報を最少から最多に再配列したところを示し、(c)は、(b)の情報を固定量子化レベルに分割した、すなわち区分した、ところを示す。

【図50】例えば、図49(a)に示すような各量子化レベル内において、プロジェクタピクセル毎の関連カメラピクセルのグループはゼロRGB値のダミーピクセルで嵩上げされ各々の量子化レベルに等しくなるよう関連カメラピクセルの合計数が増やされることを示す。

【図51】データを第1バイトサイズを有するデータ・ワードで主メモリ(すなわちグローバル・メモリ)からローカル・メモリに転送できるが、処理コアはローカル・メモリからデータを第1バイトサイズより小さい第2バイトサイズを有するデータ・ワードで読み取ることができることを示す。

【図52】3バイトのデータ(すなわちRGB情報)が16バイトのデータ・ワードに書き込まれ、次に16バイトのデータから3バイトのデータ単位でアクセスできることを示す。

10

20

30

40

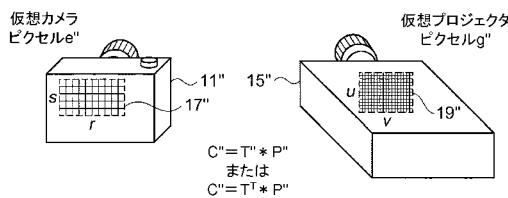
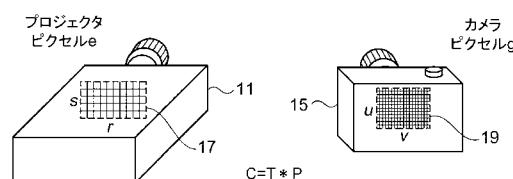
50

【符号の説明】

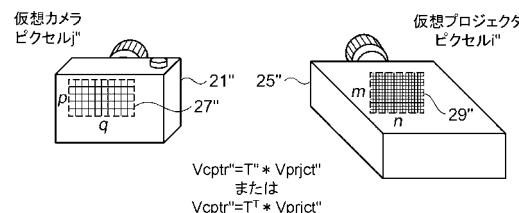
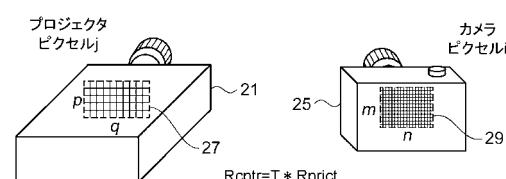
【0337】

1b, 1c...光線、3...プロジェクタレンズ、5...映写表面、7...カメラレンズ、21...実プロジェクタ、25...実カメラ、27...実プロジェクタ、29...実カメラピクセルアレイ。

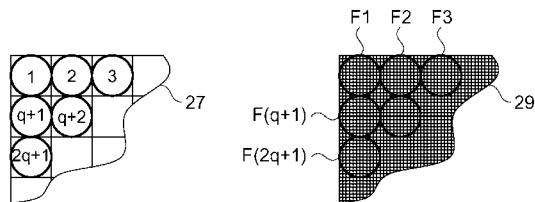
【図1】



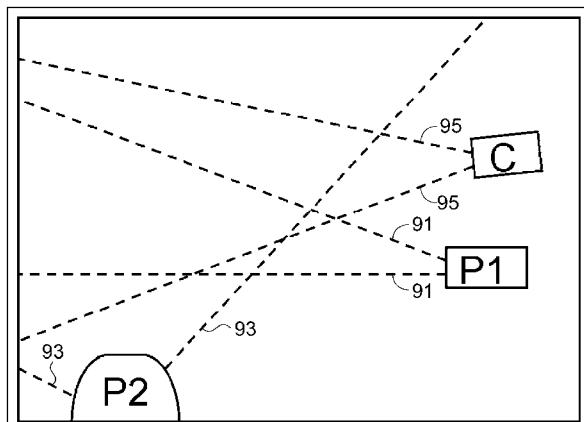
【図2】



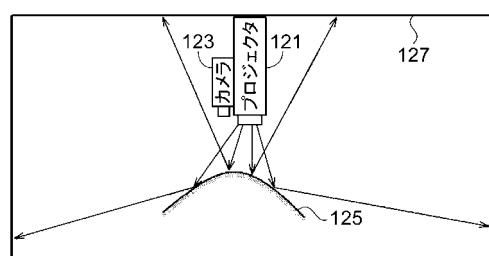
【図 1 1】



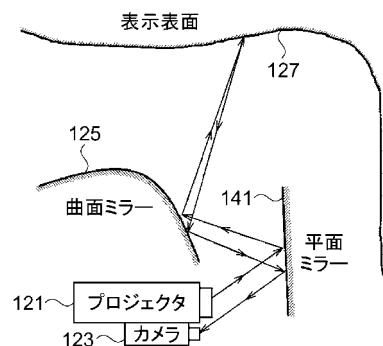
【図 2 1】



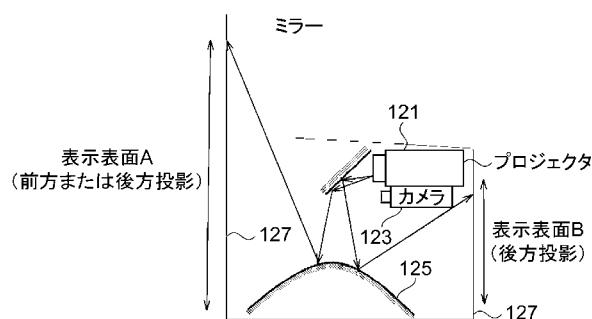
【図 3 7】



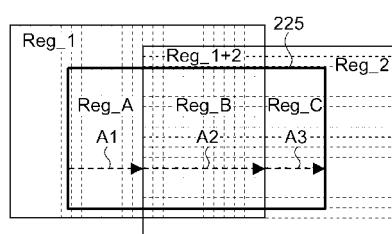
【図 3 8】



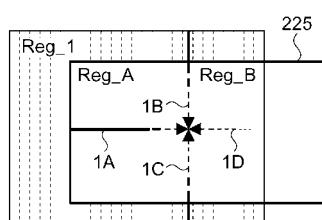
【図 3 9】



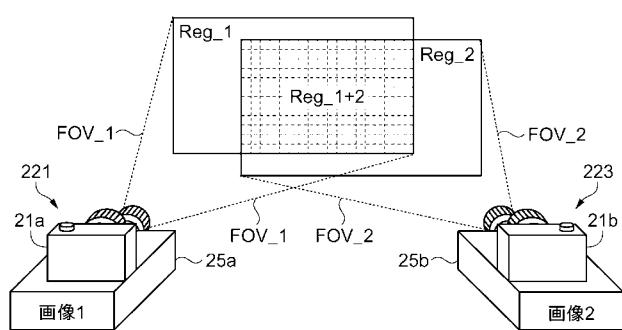
【図 4 1】



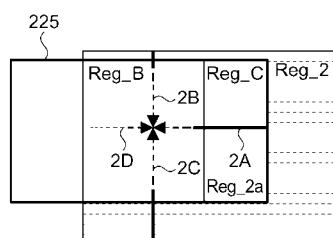
【図 4 2】



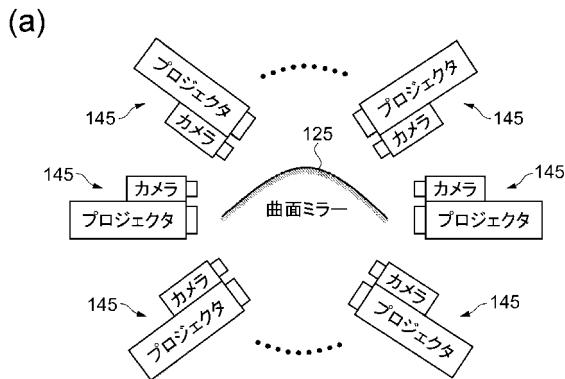
【図 4 0】



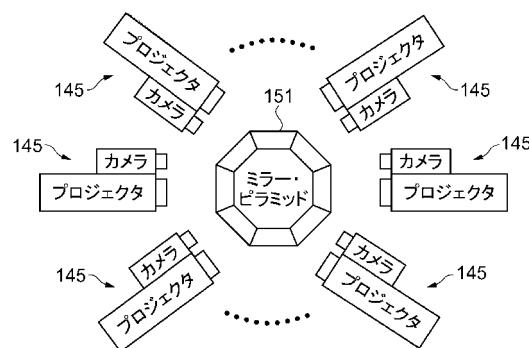
【図43】



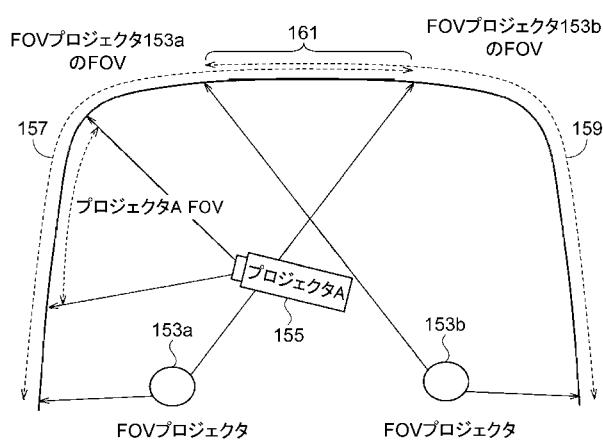
【図45】



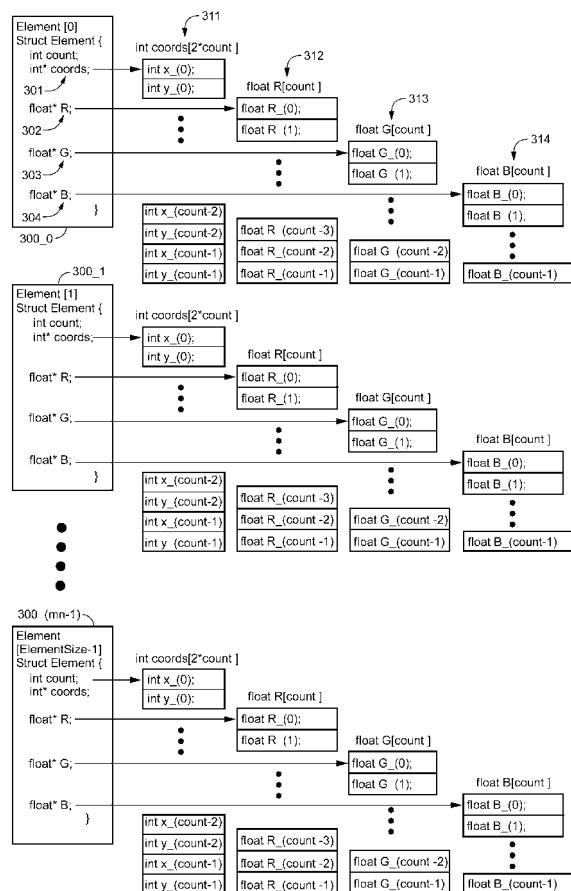
(b)



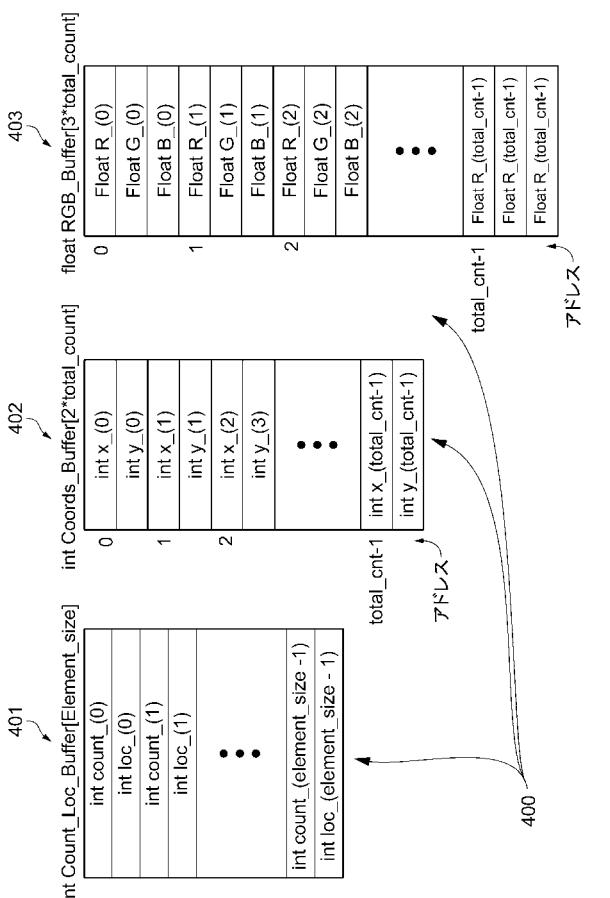
【図46】



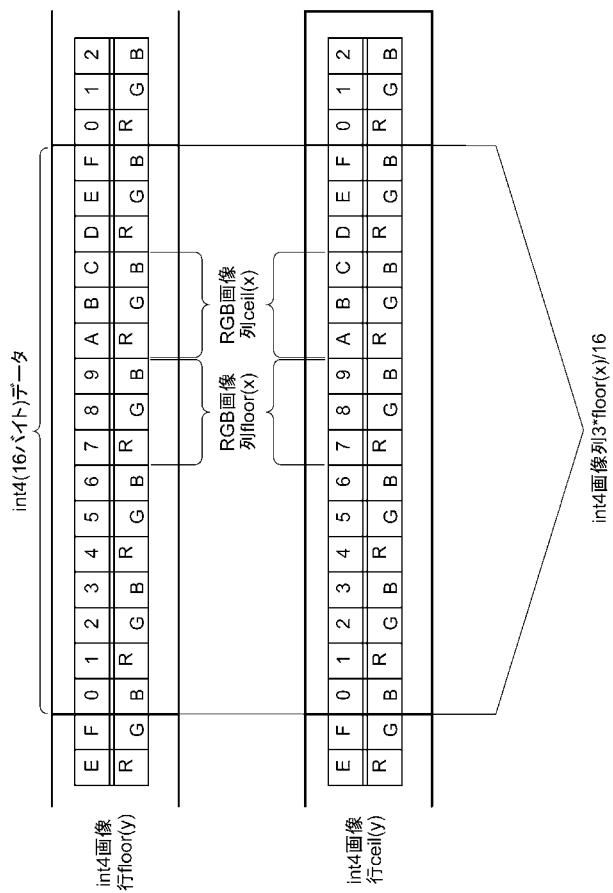
【図47】



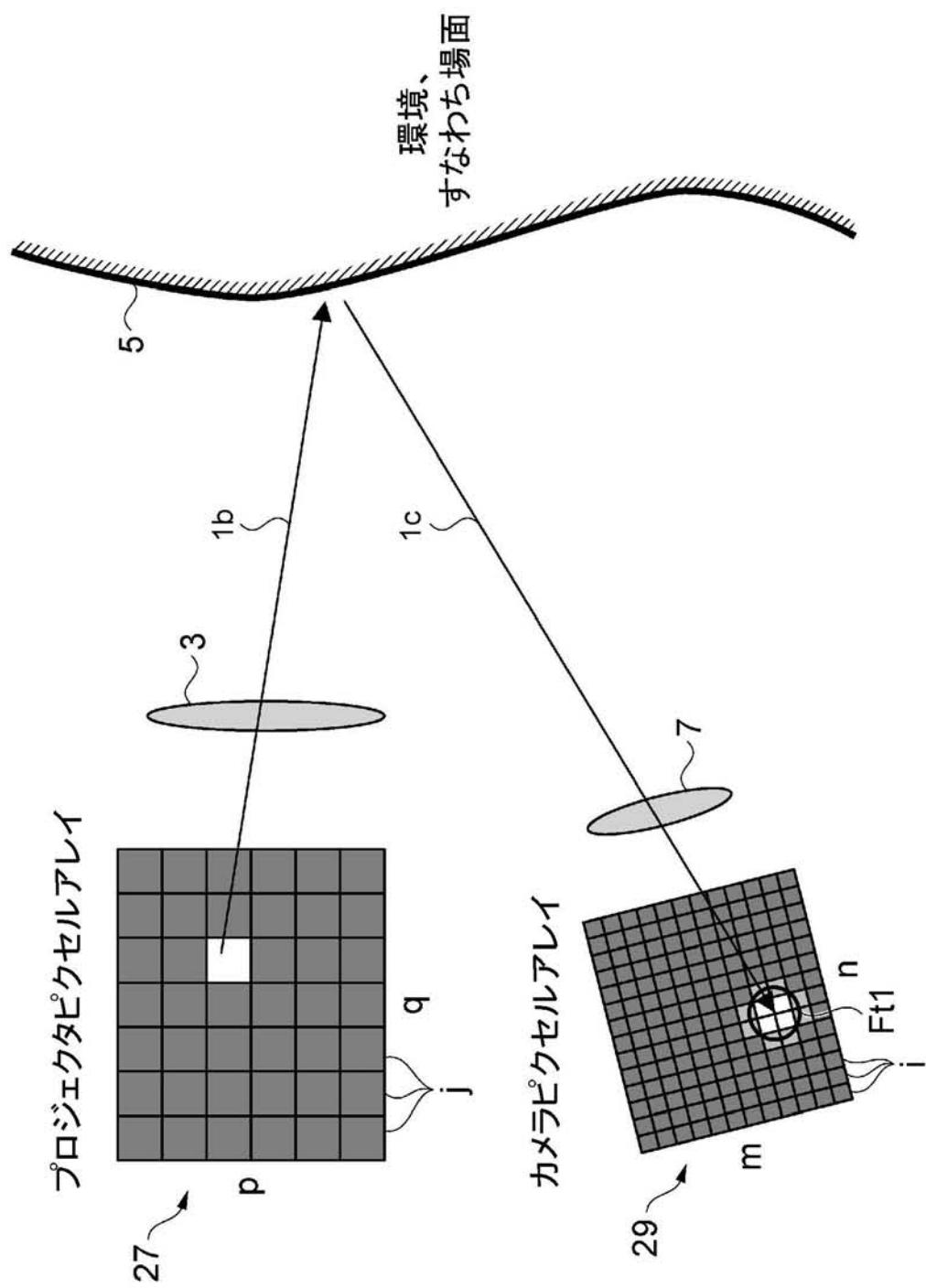
【図 4 8】



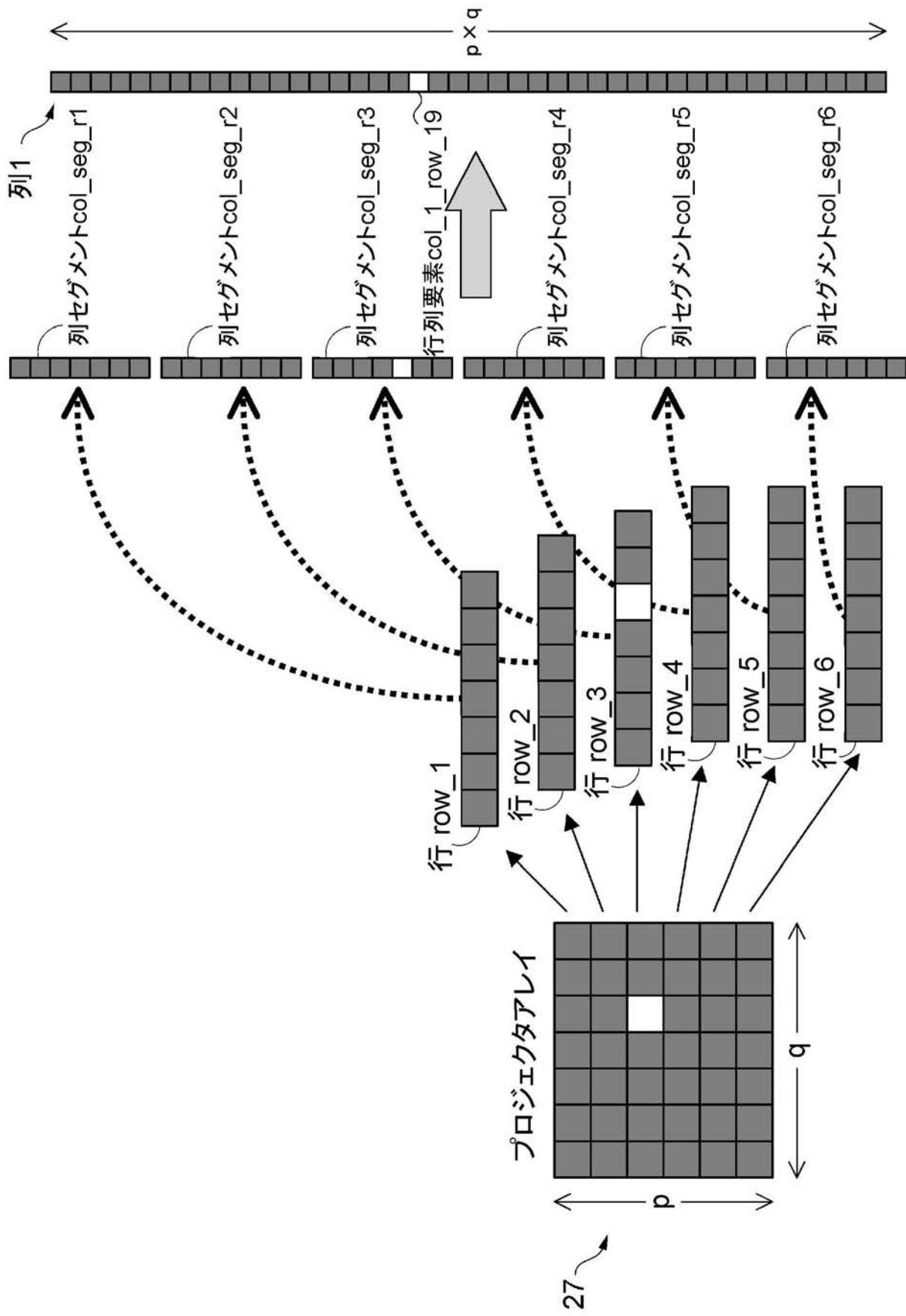
【 図 5 2 】



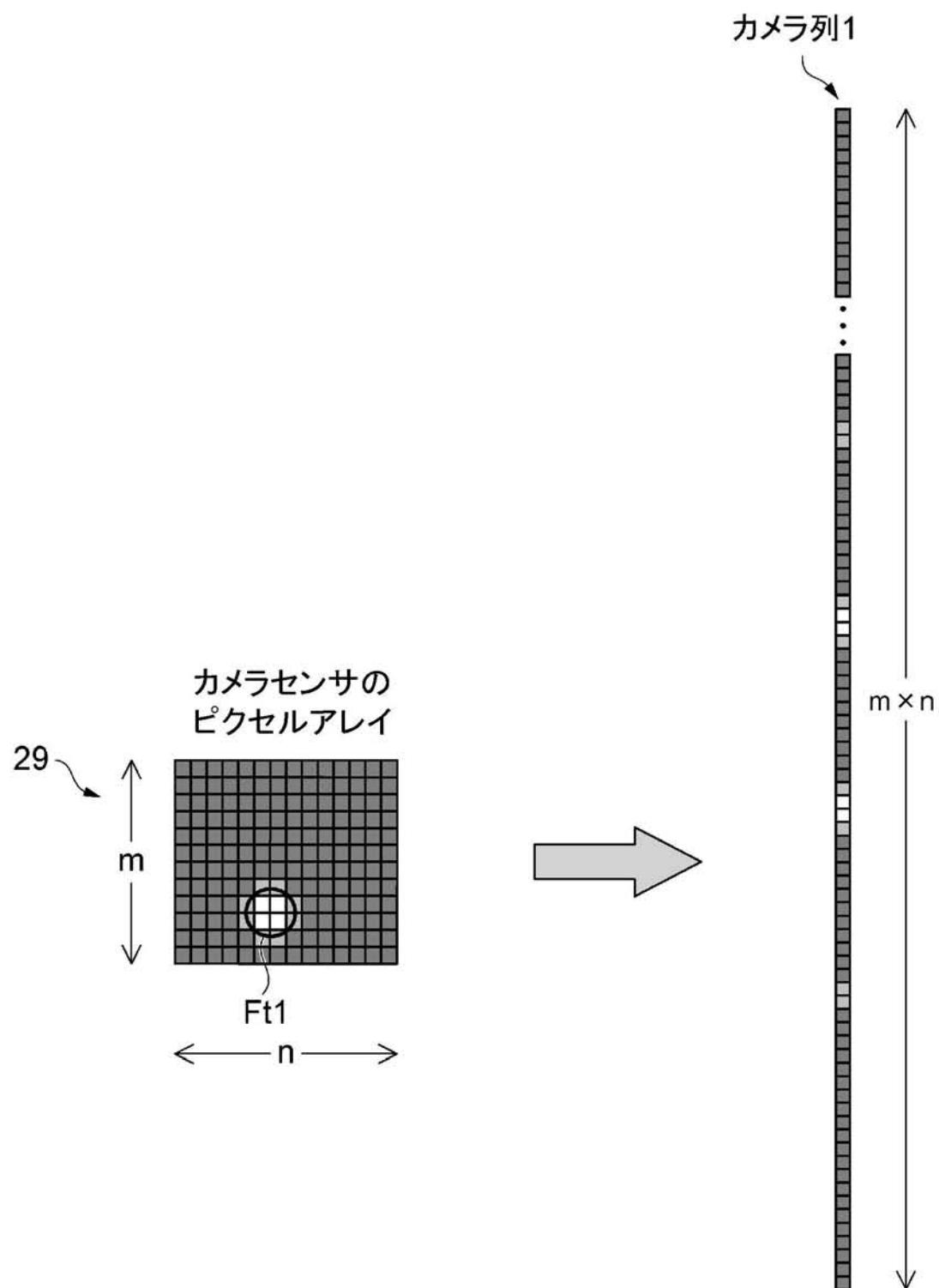
【図3】



【図4】



【図 5】



【図 6】

Diagram illustrating a 2D array structure with annotations. The array is composed of several sub-tables. The first sub-table is a 6x6 grid with columns labeled 1 to 6 and rows labeled $n+1$ to $n+6$. The second sub-table is a 6x3 grid with columns labeled $n-2$ to n and rows labeled $2n-2$ to $2n$. The third sub-table is a 6x3 grid with columns labeled $n-2$ to n and rows labeled $3n-2$ to $3n$. The fourth sub-table is a 6x3 grid with columns labeled $n-2$ to n and rows labeled $4n-2$ to $4n$. The fifth sub-table is a 6x3 grid with columns labeled $n-2$ to n and rows labeled $5n-2$ to $5n$. The sixth sub-table is a 6x3 grid with columns labeled $n-2$ to n and rows labeled $6n-2$ to $6n$. The seventh sub-table is a 3x3 grid with columns labeled $(m-1)n+1$ to $(m-1)n+3$ and rows labeled $mn-2$ to mn . Ellipses indicate the continuation of the grid structure.

Annotations include:

- A circled '29' with an arrow pointing to the cell at row $n+1$, column 2.
- A circled 'Ft2' with an arrow pointing to the first row of the first sub-table.
- Braces labeled 'i' indicating the width of the first sub-table (6 columns).
- Ellipses indicating the continuation of the grid structure.
- Ellipses indicating the continuation of the grid structure.
- Ellipses indicating the continuation of the grid structure.

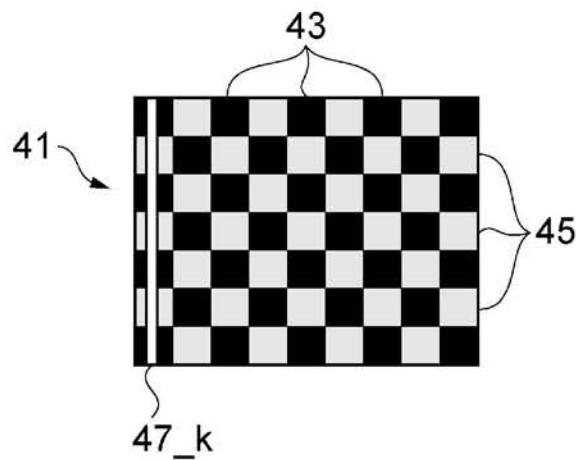
1	2	3	4	5	6
$n+1$	$n+2$	$n+3$	$n+4$	$n+5$	$n+6$
$2n+1$	$2n+2$	$2n+3$	$2n+4$	$2n+5$	$2n+6$
$3n+1$	$3n+2$	$3n+3$	$3n+4$	$3n+5$	$3n+6$
$4n+1$	$4n+2$	$4n+3$	$4n+4$	$4n+5$	$4n+6$
$5n+1$	$5n+2$	$5n+3$	$5n+4$	$5n+5$	$5n+6$
⋮					
$(m-1)n+1$	$(m-1)n+2$	$(m-1)n+3$	⋮	⋮	$mn-2$
$mn-1$	mn				

【図7】

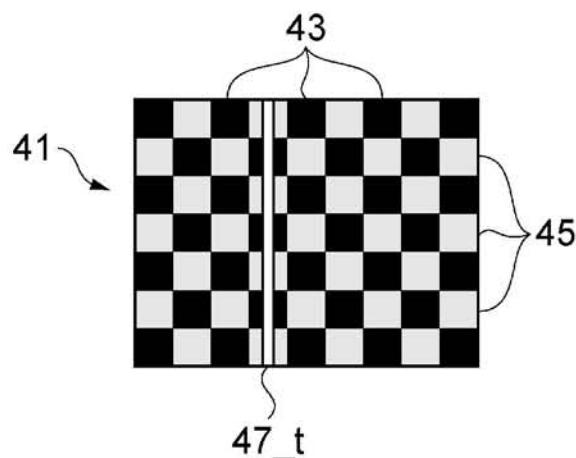
1	NZ
2	NZ
3	NZ
4	ZERO
⋮	
n	ZERO
n+1	NZ
n+2	NZ
n+3	NZ
n+4	ZERO
⋮	
2n	ZERO
2n+1	NZ
2n+2	NZ
2n+3	NZ
2n+4	ZERO
⋮	
3n	ZERO
3n+1	NZ
3n+2	NZ
3n+3	NZ
3n+4	ZERO
⋮	
4n	ZERO
4n+1	NZ
4n+2	NZ
4n+3	NZ
4n+4	ZERO
4n+5	ZERO
4n+6	ZERO
⋮	
mn	ZERO

【図 8】

(a)

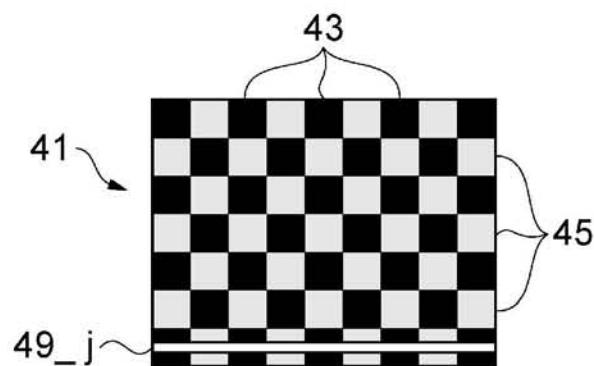


(b)

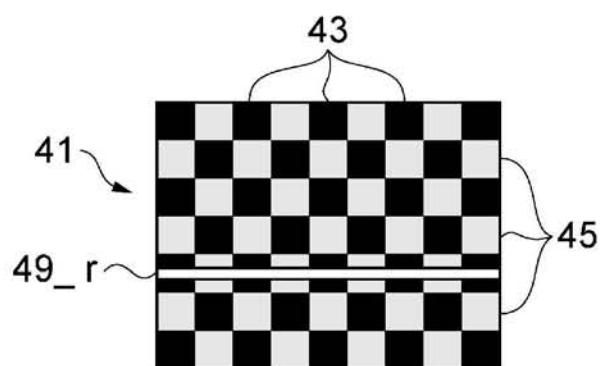


【図9】

(a)

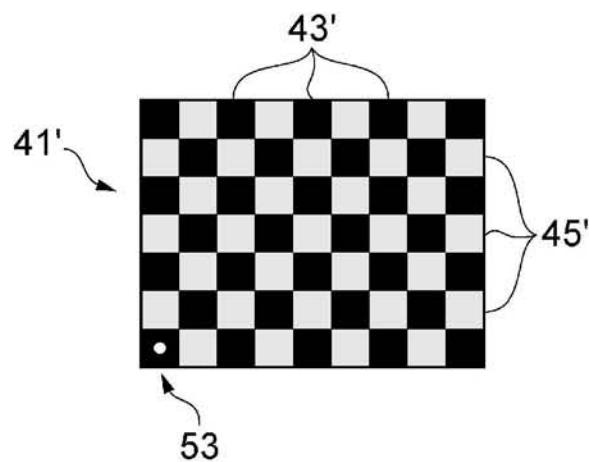


(b)

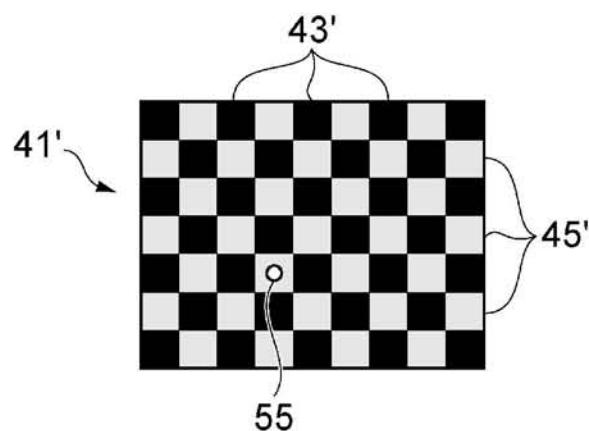


【図 10】

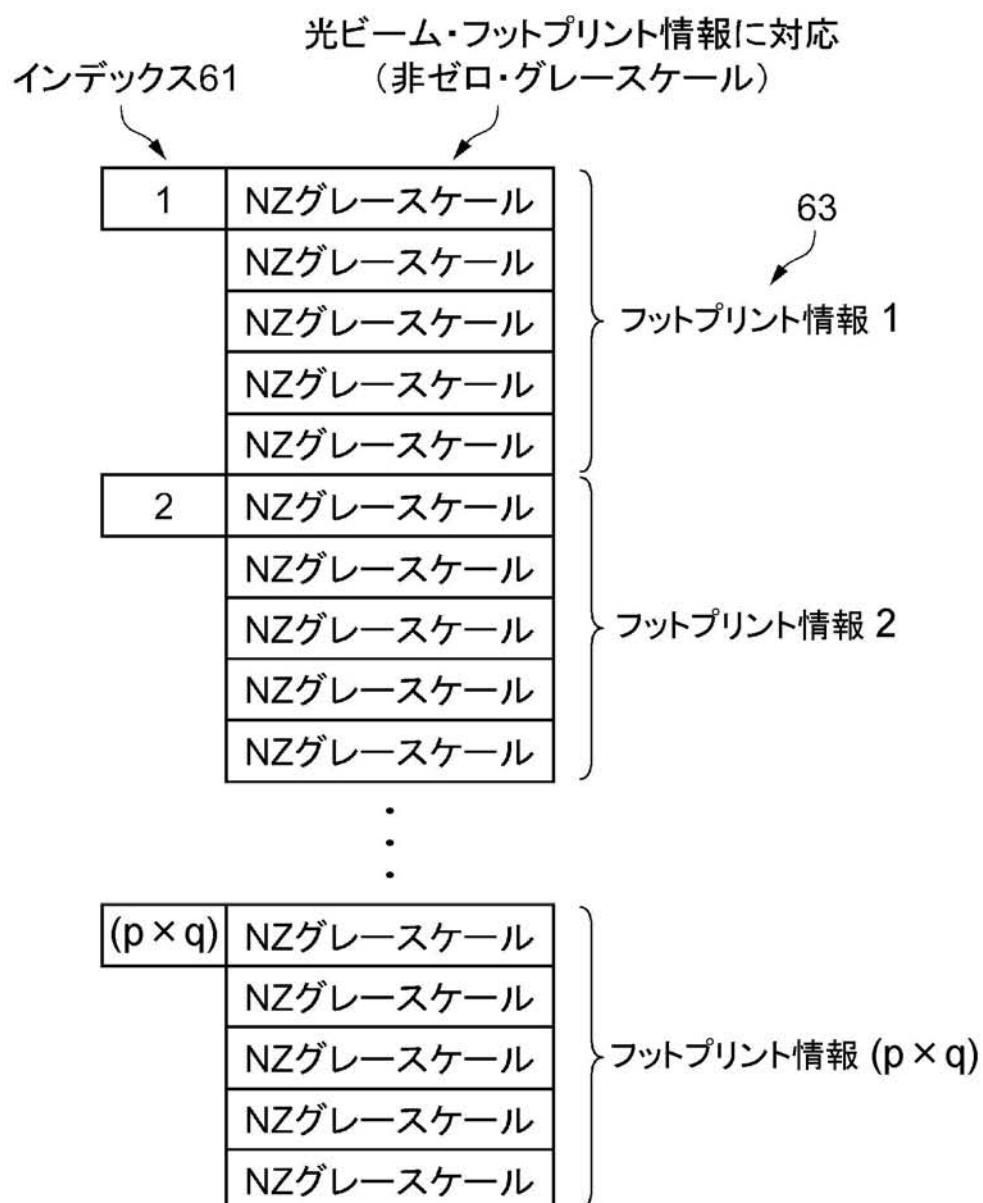
(a)



(b)

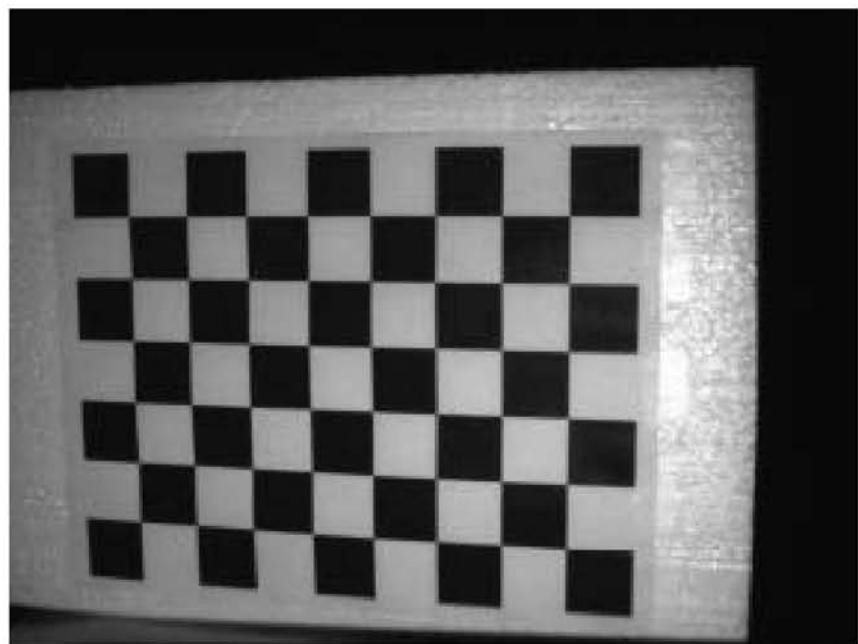


【図 12】

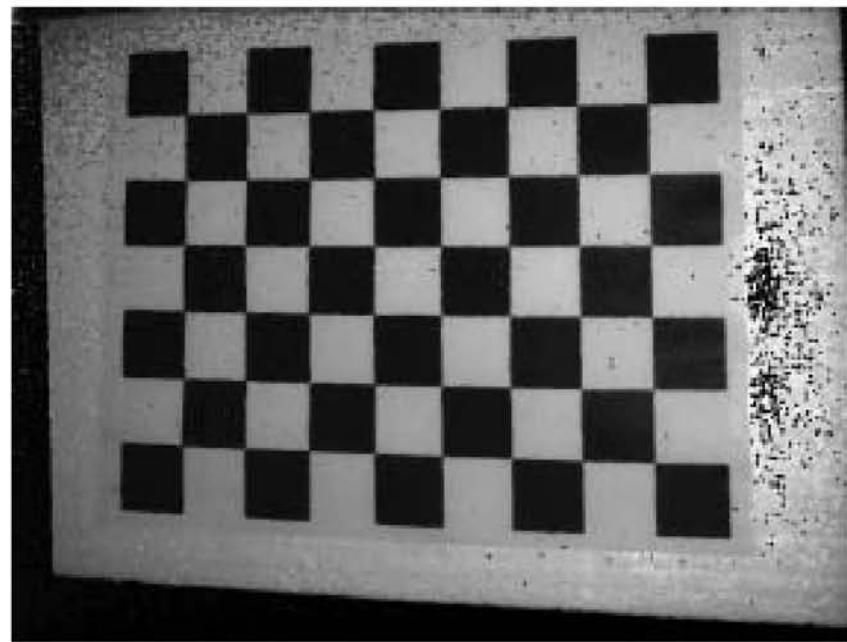


【図13】

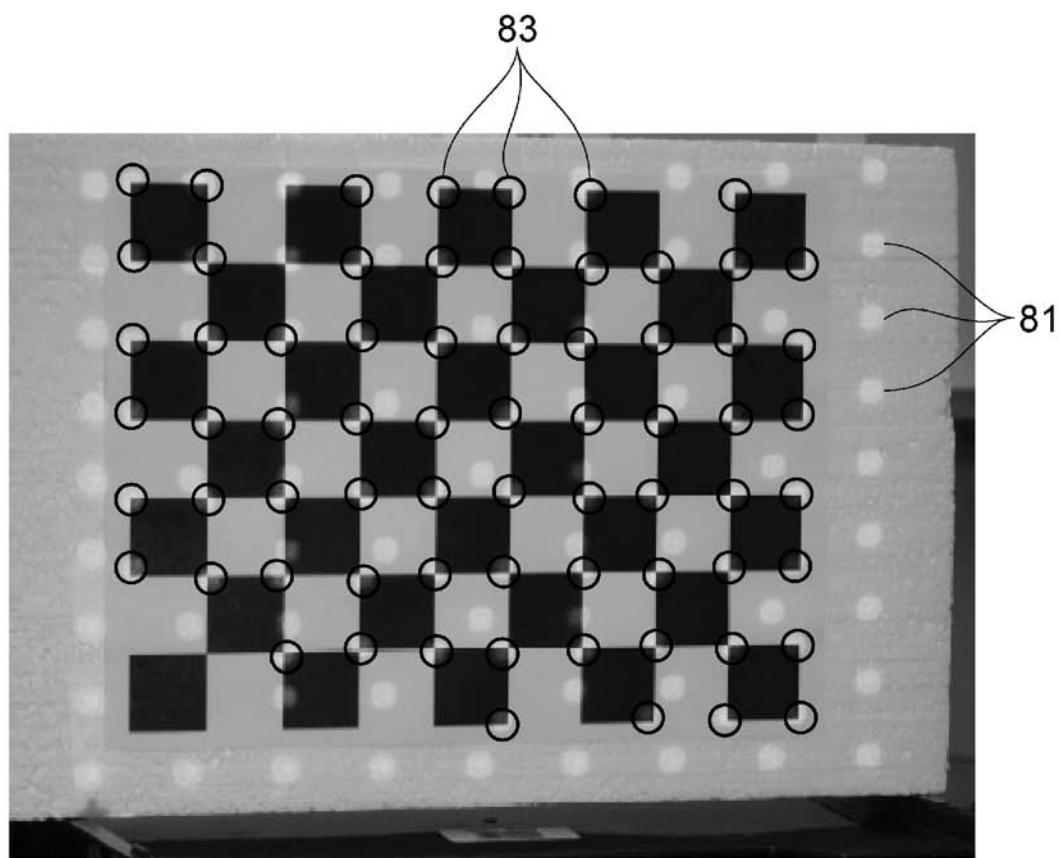
(a)



(b)



【図14】

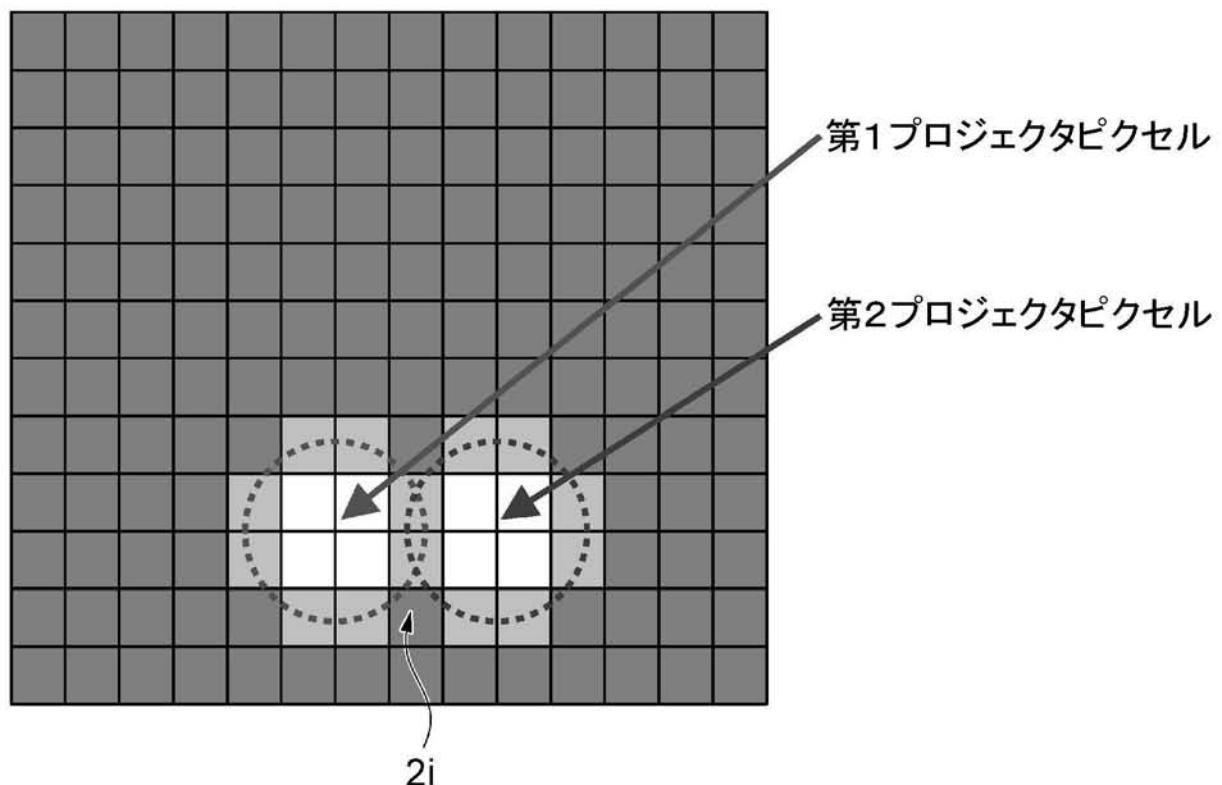


【図15】

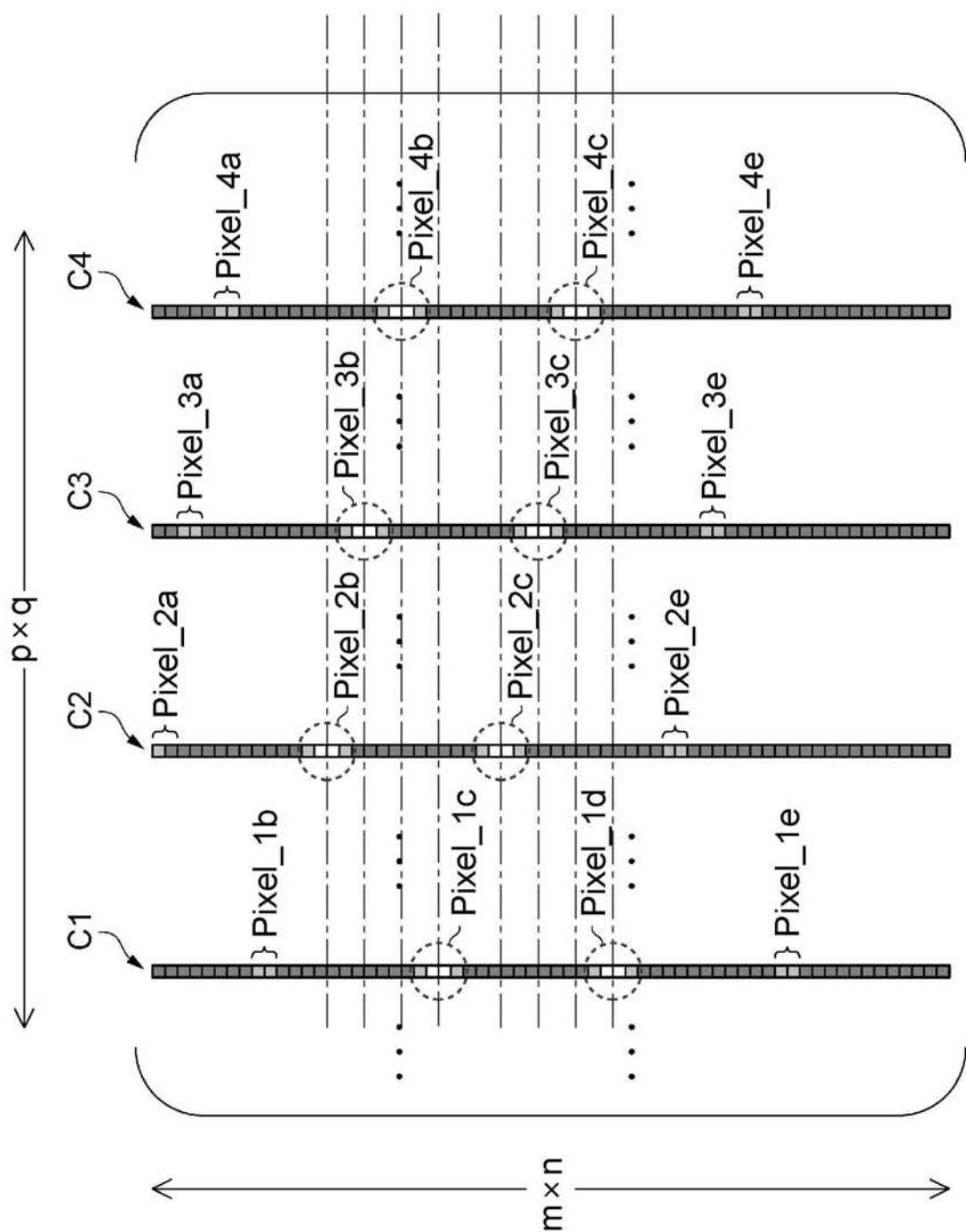


【図 16】

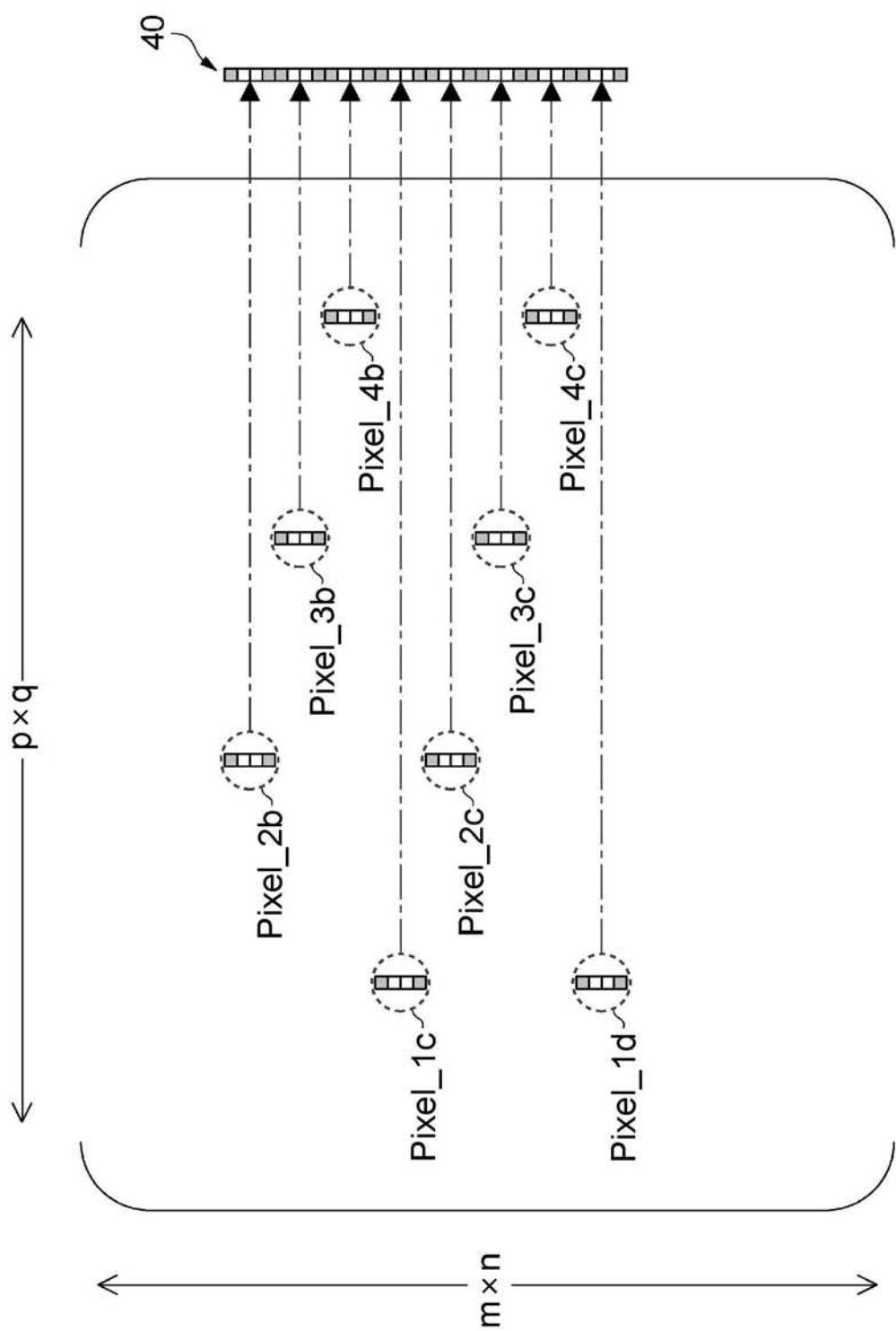
カメラピクセルアレイ



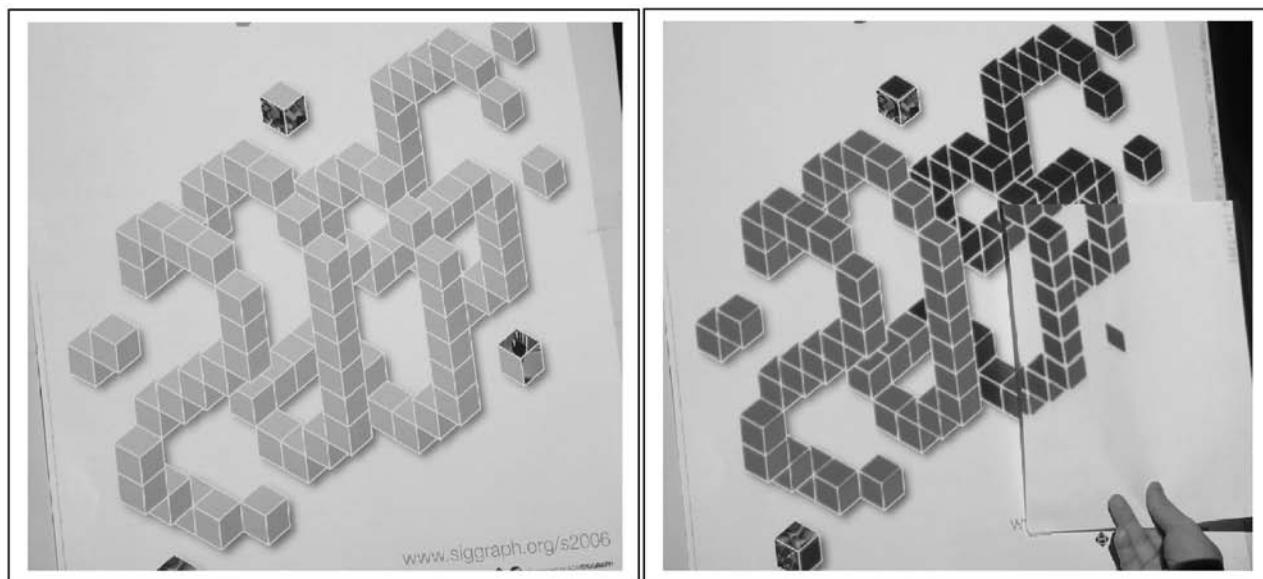
【図 17】



【図 18】



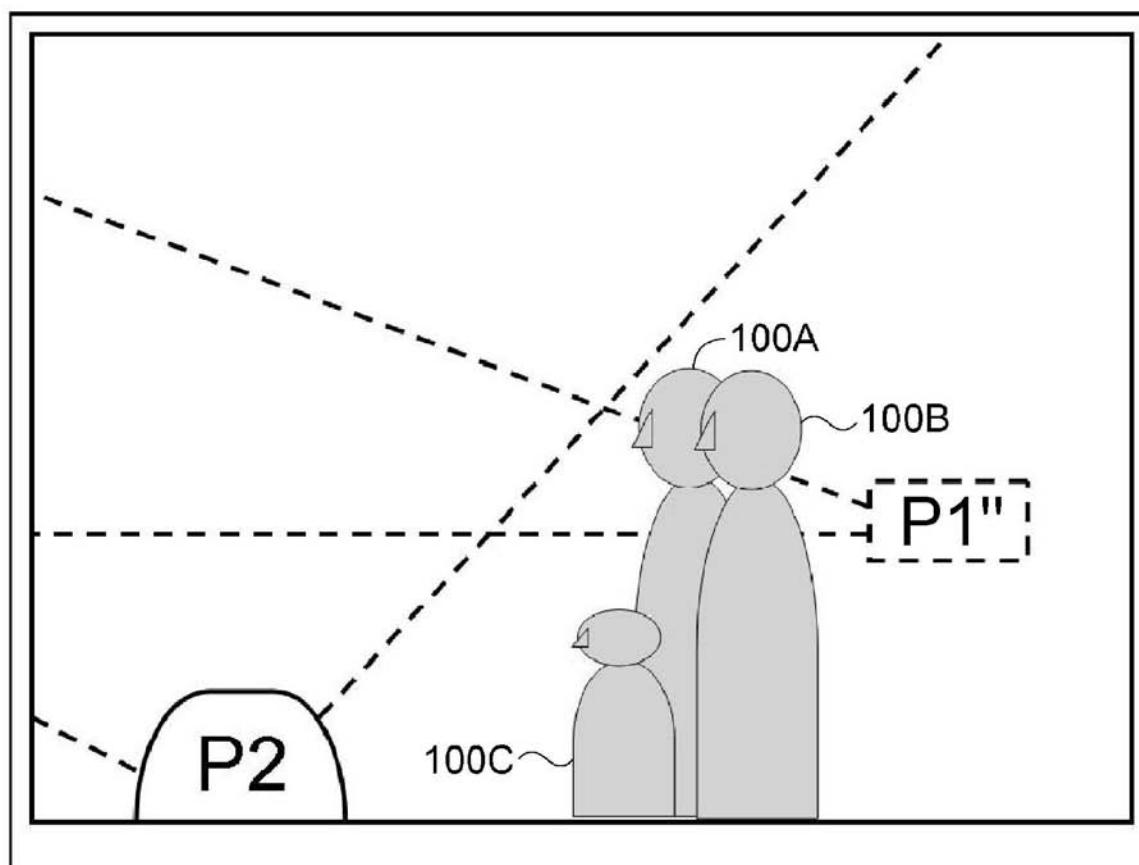
【図19】



【図20】



【図 2 2】



【図23】

(a)



(b)



(c)

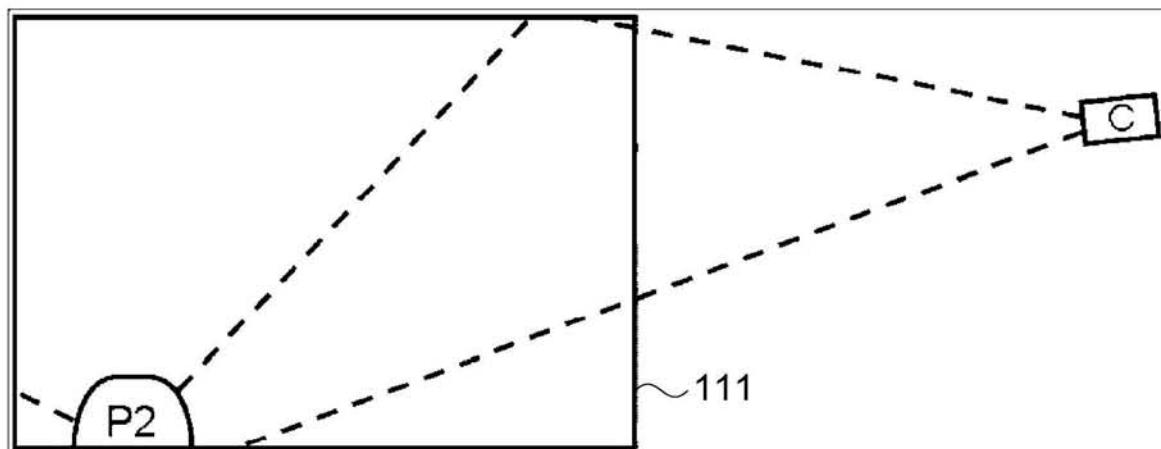


【図24】

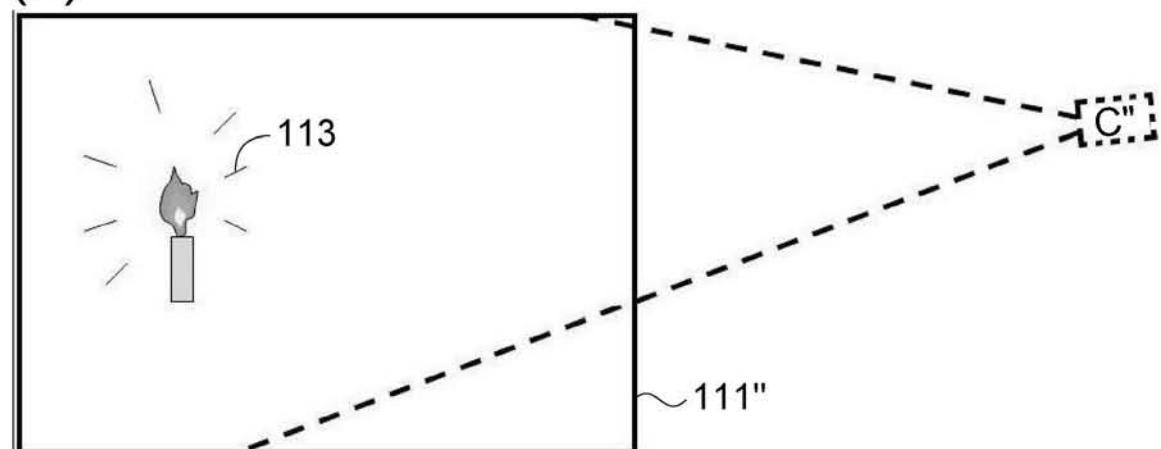


【図25】

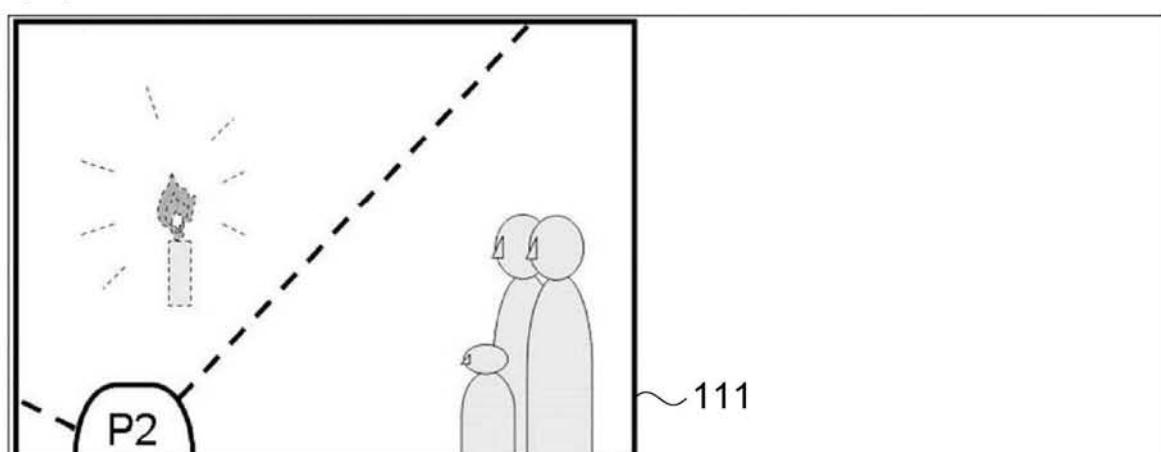
(a)



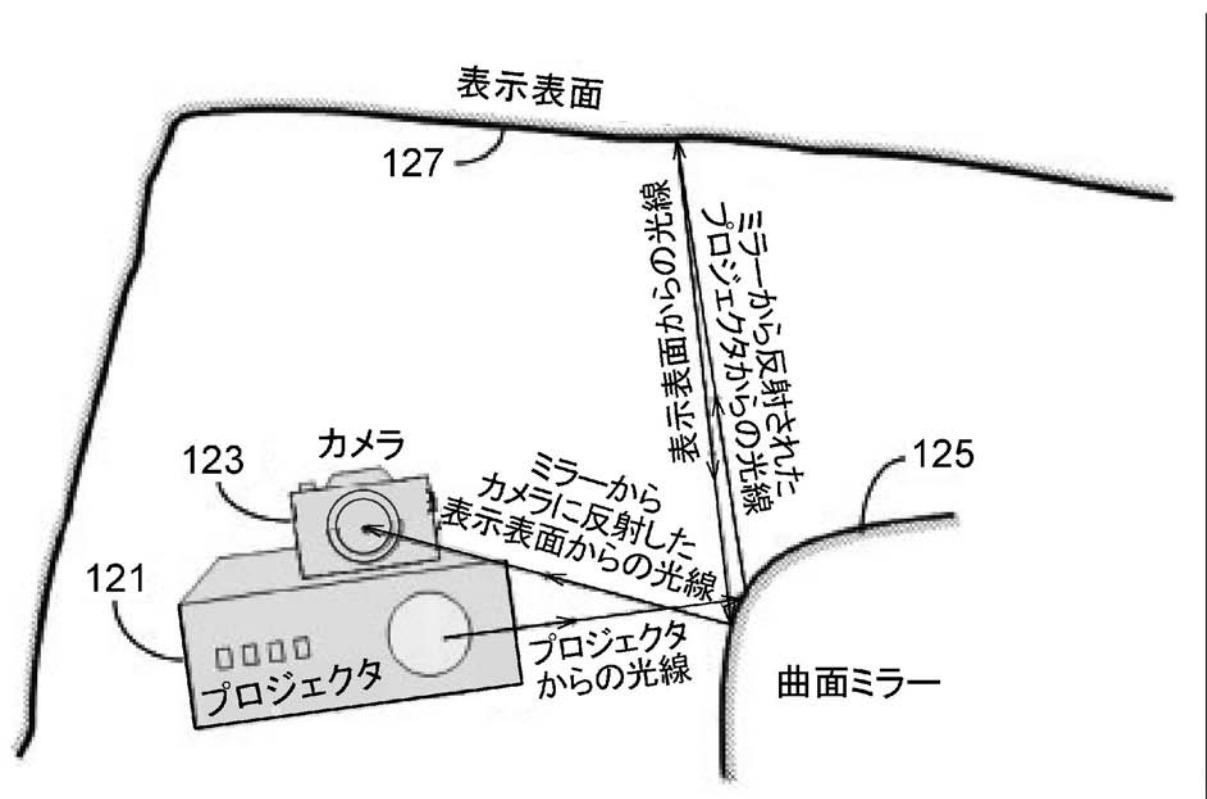
(b)



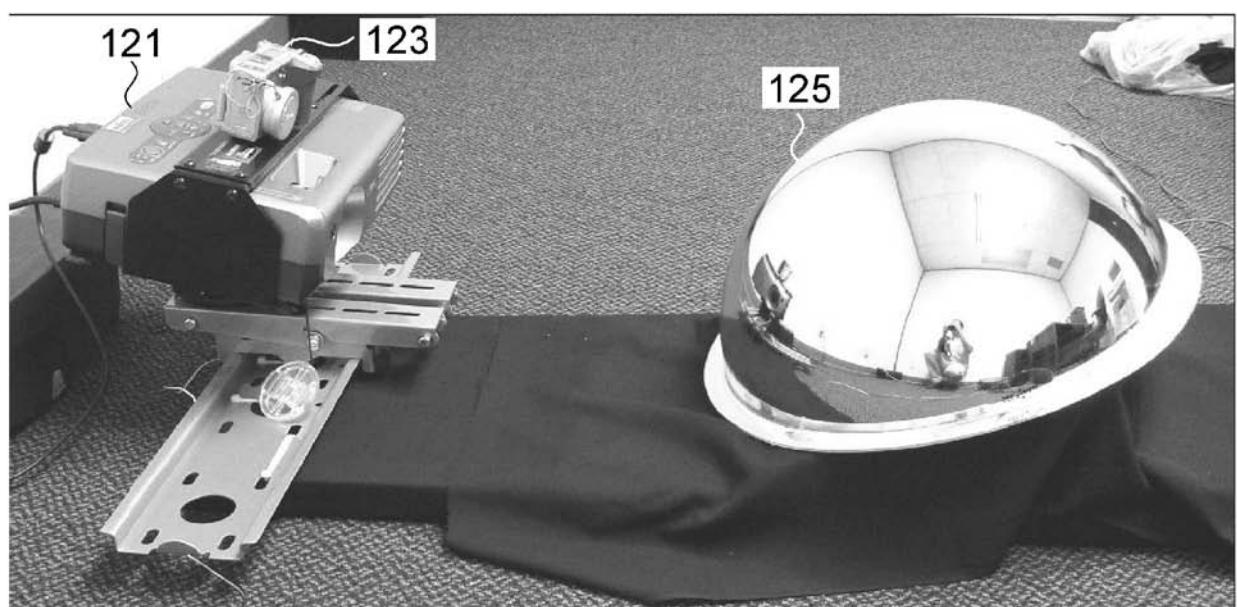
(c)



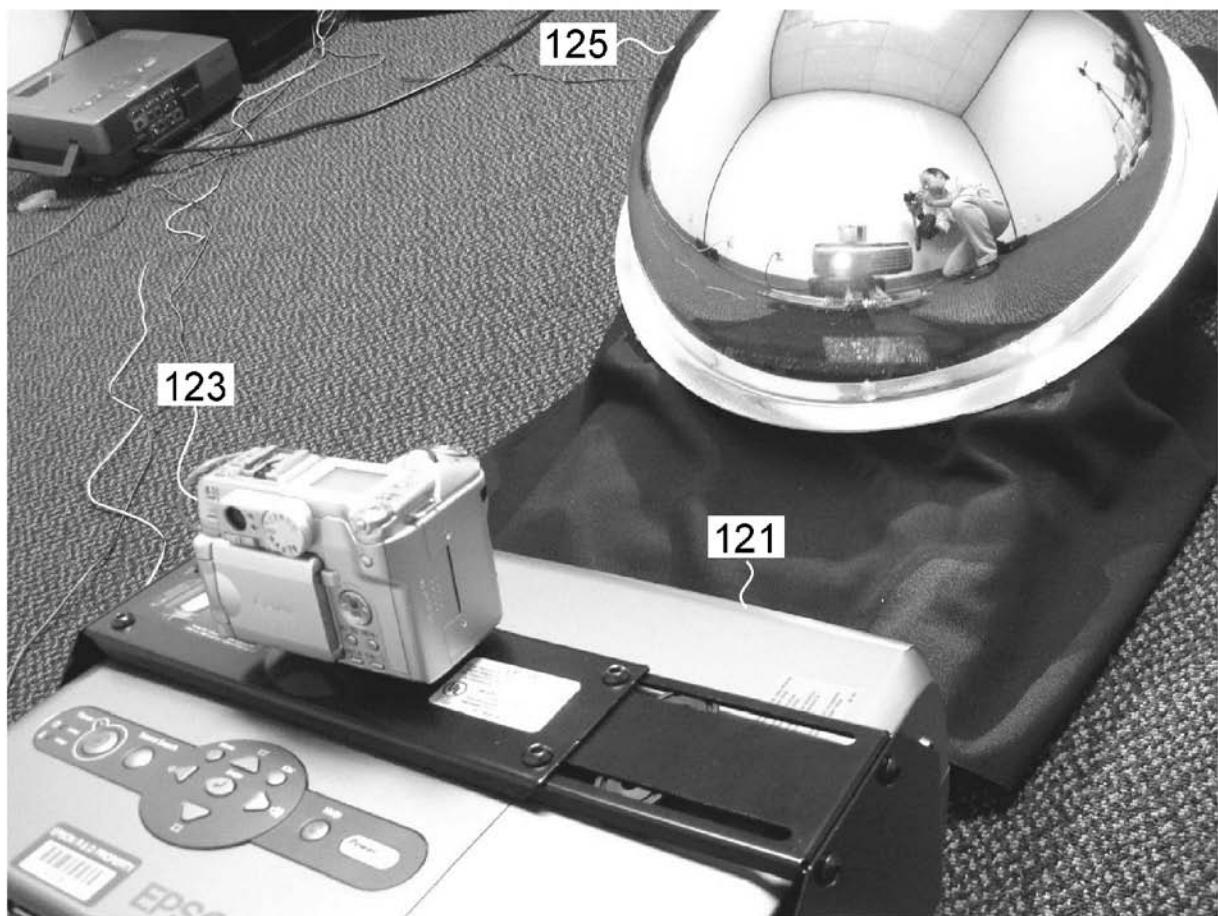
【図 2 6】



【図 2 7】

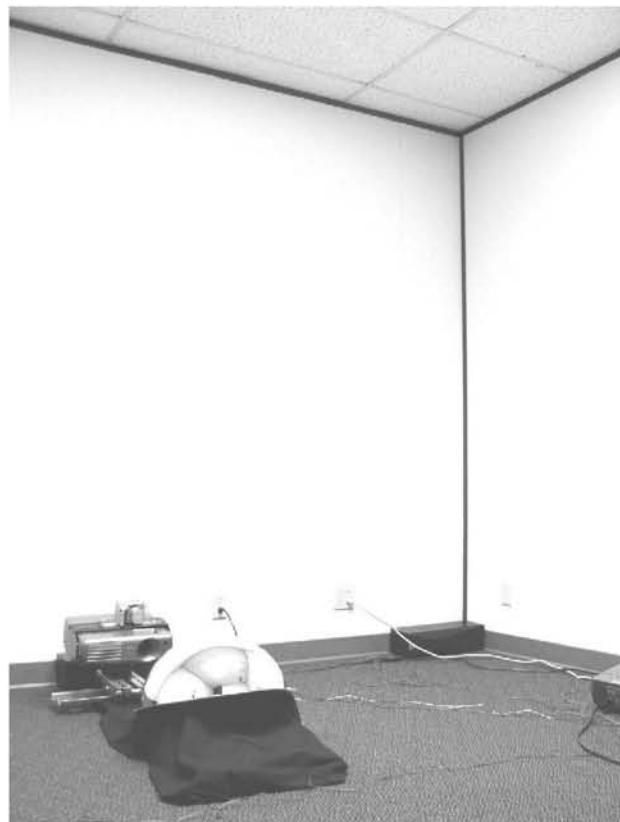


【図28】



【図29】

(a)



(b)



【図 3 0】

(a)

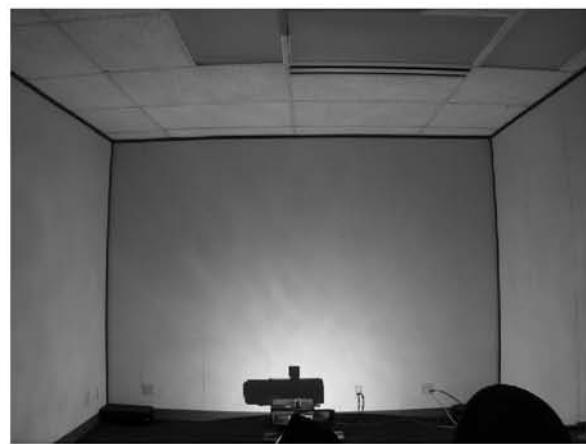


(b)



【図 3 1】

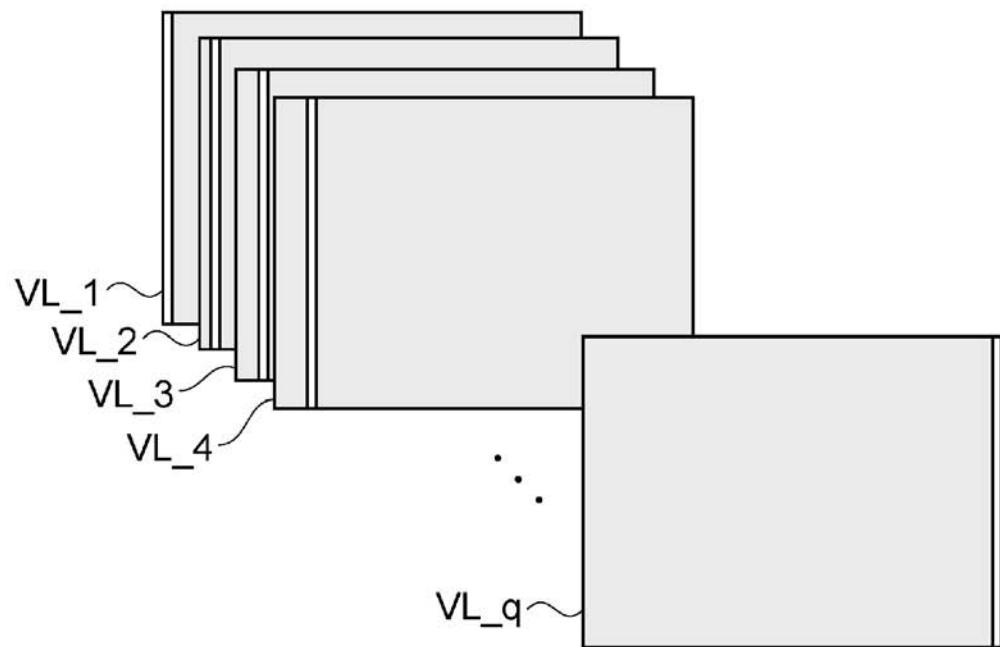
(a)



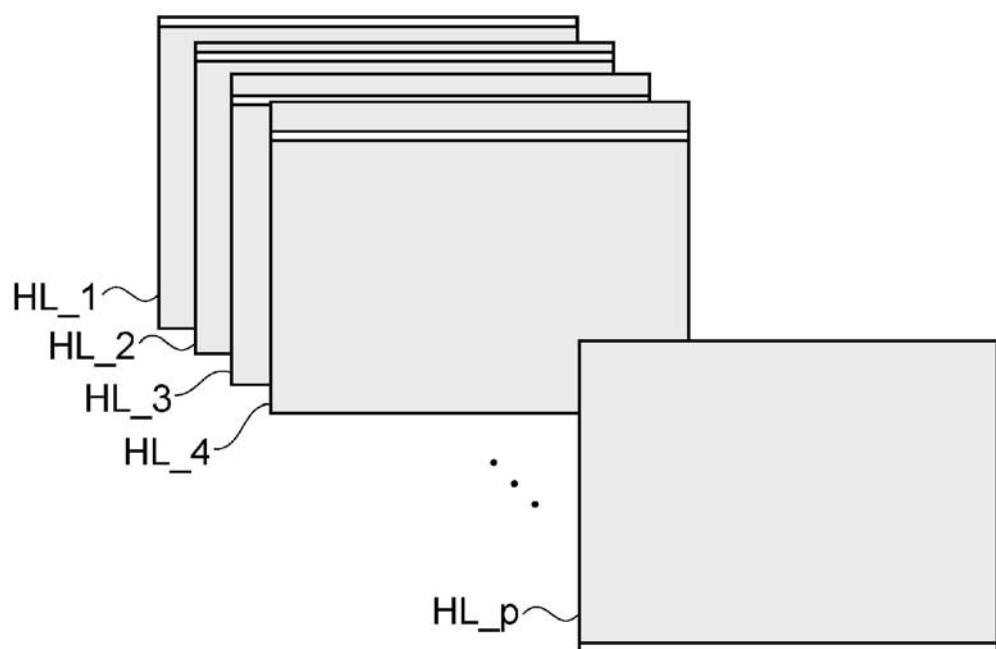
(b)



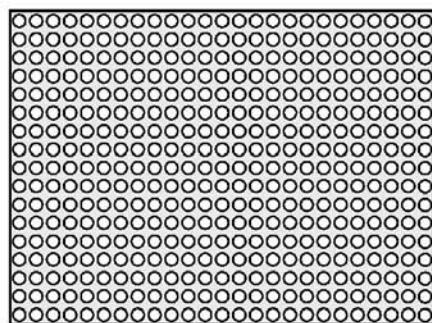
【図 3 2】



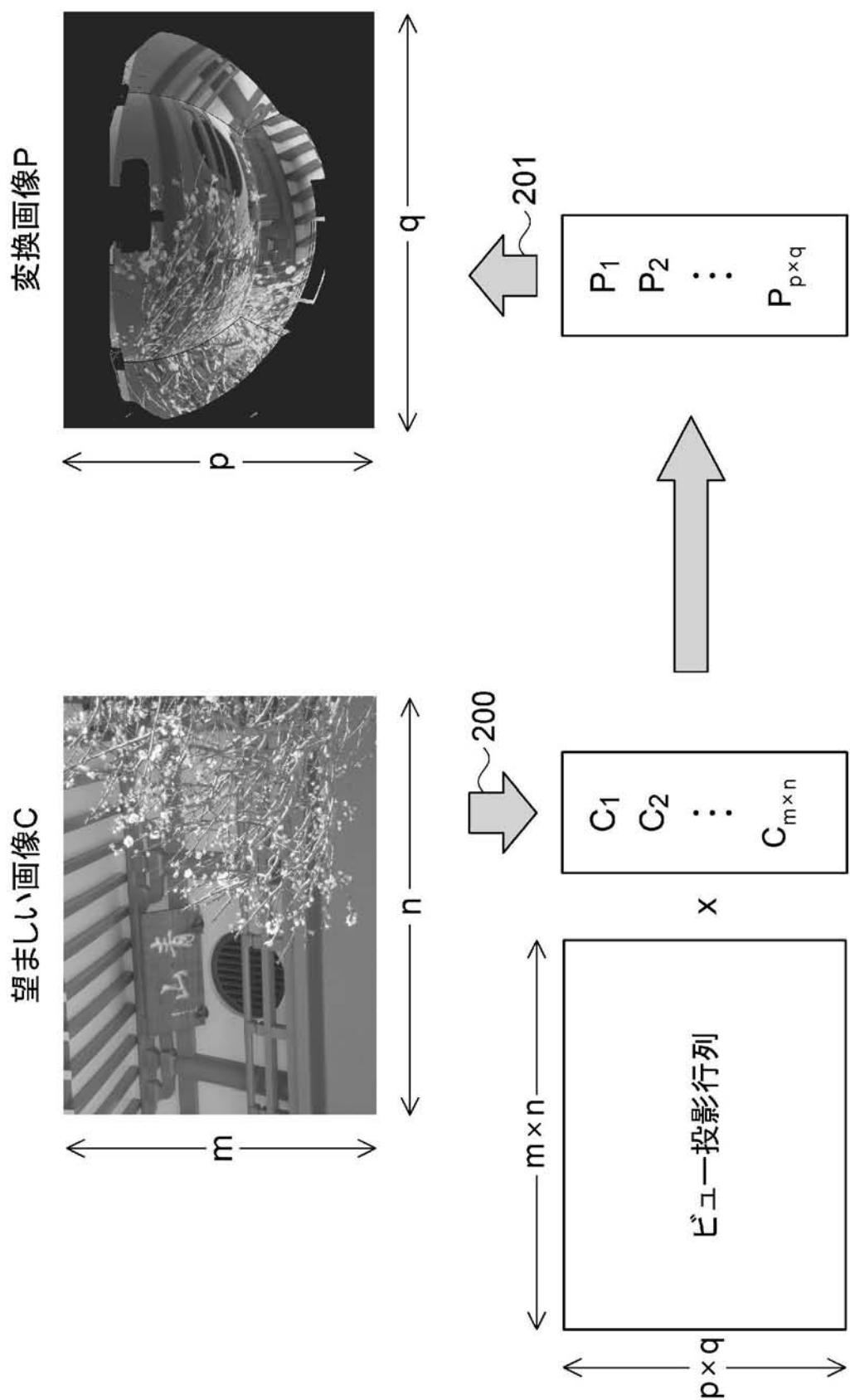
【図 3 3】



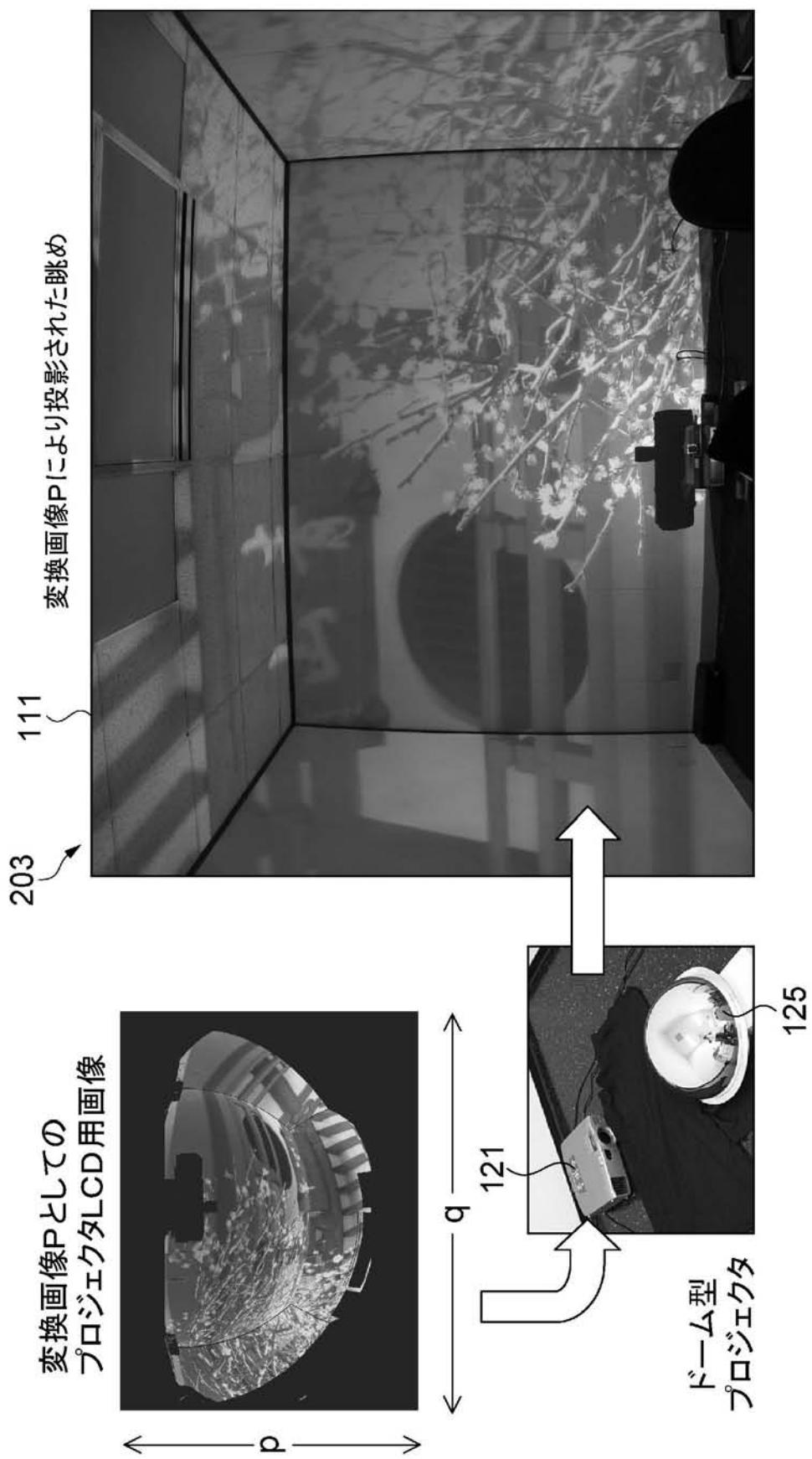
【図 3 4】



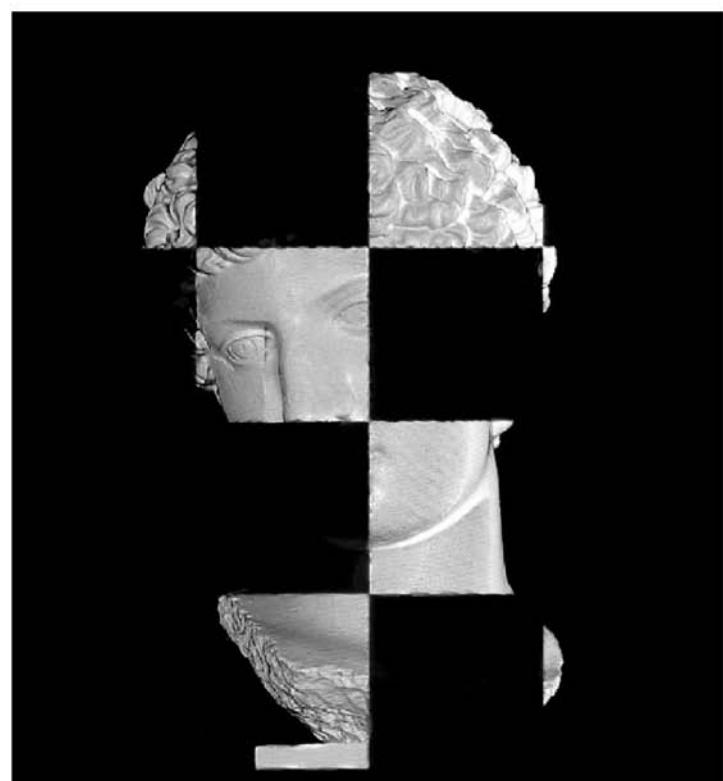
【図 3 5】



【図 3 6】



【図44】



フロントページの続き

(51) Int.Cl.	F I	テーマコード(参考)
H 04N 5/232 (2006.01)	G 09G 5/00 510B	5C122
	G 09G 5/00 X	
	G 09G 5/00 550C	
	G 09G 5/00 550X	
	G 09G 5/00 510V	
	G 09G 5/00 530H	
	G 09G 5/00 550H	
	H 04N 9/31 A	
	H 04N 5/232 Z	

(72)発明者 カー ハン タン

アメリカ合衆国 カリフォルニア州 サンタ クララ アベニダ ディ ラゴ 4943

F ターム(参考) 2K103 AA01 AA16 AA22 AB06 AB08 BB06 BB07 BB08 BC38 BC44
CA21 CA28 CA32 CA38 CA55 CA72
5C058 BA27 BB25 EA02
5C060 GB06 HA00 JA01
5C080 BB05 DD01 EE29 JJ01 JJ02 JJ06
5C082 AA03 AA21 CA32 CA81 CB01
5C122 DA03 DA30 EA61 FH04 FH06 FK23 GA18 GA24 HB05 HB06