

(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2005-4201

(P2005-4201A)

(43) 公開日 平成17年1月6日(2005.1.6)

(51) Int. Cl.⁷
G03B 21/00

F I
G O 3 B 21/00 D

テーマコード (参考)
2 K 1 0 3

審査請求 未請求 請求項の数 18 O L 外国語出願 (全 47 頁)

(21) 出願番号 特願2004-166072 (P2004-166072)
 (22) 出願日 平成16年6月3日 (2004.6.3)
 (31) 優先権主張番号 10/458386
 (32) 優先日 平成15年6月10日 (2003.6.10)
 (33) 優先権主張国 米国 (US)

(71) 出願人 597067574
 ミツビシ・エレクトリック・リサーチ・ラ
 ボラトリーズ・インコーポレイテッド
 アメリカ合衆国、マサチューセッツ州、ケ
 ンブリッジ、ブロードウェイ 201
 201 BROADWAY, CAMBR
 IDGE, MASSACHUSETTS
 02139, U. S. A.
 (74) 代理人 100057874
 弁理士 曾我 道照
 (74) 代理人 100110423
 弁理士 曾我 道治
 (74) 代理人 100084010
 弁理士 古川 秀利

最終頁に続く

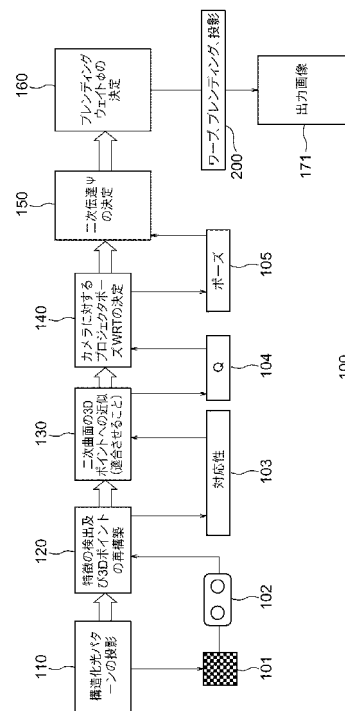
(54) 【発明の名称】 画像を表示面上に投影する方法及びシステム

(57) 【要約】

【課題】 1つまたは複数の画像を湾曲した表示面上に投影する方法。

【解決手段】 まず、所定の構造化光パターンが表示面上に投影される。表示面上に投影された画像から、立体画像対が得られる。次に、表示面を介しての所定の画像と立体画像対の間の二次伝達関数が求められる。したがって、任意の出力画像を二次伝達関数に従ってワーピングすることができ、その出力画像は、表示面上に投影されるときに正しく表される。

【選択図】 図1



【特許請求の範囲】

【請求項 1】

所定の画像を表示面上に投影すること、
 前記所定の画像の立体画像対を得ること、
 前記表示面を介しての前記所定の画像と前記立体画像対の間の二次伝達関数を求めること、
 前記二次伝達関数に従って出力画像をワーピングすること、
 及び
 前記ワーピングした出力画像を前記表示面上に投影すること
 を含む画像を表示面上に投影する方法。

10

【請求項 2】

前記表示面は二次曲面である
 請求項 1 記載の画像を表示面上に投影する方法。

【請求項 3】

前記表示面は、球面、半球面、回転楕円面、ドーム、柱面、錐面、放物面、双曲面、双曲放物面、及び楕円面からなる群から選択される
 請求項 2 記載の画像を表示面上に投影する方法。

【請求項 4】

前記二次曲面は、凹面形である
 請求項 2 記載の画像を表示面上に投影する方法。

20

【請求項 5】

前記二次曲面は、凸面形である
 請求項 2 記載の画像を表示面上に投影する方法。

【請求項 6】

複数の所定の画像を複数の視点から前記表示面上に投影すること、
 前記複数の所定の画像の立体入力画像対を得ること、
 前記表示面を介しての各所定の画像と前記立体入力画像対の間の二次伝達関数を求めること、
 前記対応する二次伝達関数に従って複数の出力画像をワーピングすること、
 及び
 前記ワーピングされた複数の出力画像を前記複数の視点から前記表示面上に投影すること
 をさらに含む請求項 1 記載の画像を表示面上に投影する方法。

30

【請求項 7】

前記表示面は、二次曲面である
 請求項 6 記載の画像を表示面上に投影する方法。

【請求項 8】

前記ワーピングされた複数の出力画像は、重ねられ、1つのシームレスな画像として見える
 請求項 1 記載の画像を表示面上に投影する方法。

40

【請求項 9】

前記所定の画像の特徴と前記立体画像対の対応する特徴の間の三次元対応性を求めること、
 前記三次元対応性に二次曲面を近似すること、
 前記立体入力画像対に対する前記所定の画像のポーズを求めること、
 及び
 前記ポーズ及び前記対応性から前記二次伝達関数を求めること
 をさらに含む請求項 1 記載の画像を表示面上に投影する方法。

【請求項 10】

投影前に、ブレンディングウェイトに従って前記ワーピングした複数の出力画像を減衰

50

すること

をさらに含む請求項 6 記載の画像を表示面上に投影する方法。

【請求項 1 1】

前記所定の画像は、チェッカー盤パターンである

請求項 1 記載の画像を表示面上に投影する方法。

【請求項 1 2】

非線形最小化を目的関数に適用して前記二次伝達関数を得ること

をさらに含む請求項 1 記載の画像を表示面上に投影する方法。

【請求項 1 3】

前記目的関数は、所定の画像中のすべてのピクセルの総二乗伝達誤差を利用する

請求項 1 2 記載の画像を表示面上に投影する方法。

10

【請求項 1 4】

前記出力画像は、任意の仮想視点に関してレンダリングされる

請求項 1 記載の画像を表示面上に投影する方法。

【請求項 1 5】

前記減衰されワーピングされた複数の出力画像を任意の視錐台へクロッピングすること

をさらに含む請求項 1 0 記載の画像を表示面上に投影する方法。

【請求項 1 6】

前記二次伝達関数は、パラメトリック的に求められる

請求項 1 記載の画像を表示面上に投影する方法。

20

【請求項 1 7】

前記ワーピングは、グラフィック処理装置のパーテックスシェーダを使用して実施される

請求項 1 記載の画像を表示面上に投影する方法。

【請求項 1 8】

所定の画像を表示面上に投影するように構成されたプロジェクタと、

前記所定の画像の立体画像対を得るように構成されたカメラと、

前記表示面を介しての前記所定の画像と前記立体画像対の間の二次伝達関数を求める手段と、

前記二次伝達関数に従って出力画像をワーピングする手段と、

前記ワーピングされた出力画像を前記表示面上に投影する手段と

を備える画像を表示面上に投影するシステム。

30

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、包括的に画像の投影に関し、特に、画像の湾曲面への投影に関する。

【背景技術】

【0002】

プロジェクタシステムは、大型の画像を表示面上にレンダリングする際に使用されてきた。複数のプロジェクタを使用すれば、さらに大きくシームレスな表示を生成することが可能である。このようなシステムは、エンターテイメント、教育、トレーニング、および科学的シミュレーション用に高解像度画像を提示することが可能な没入型視覚化環境を構築する際に特に有用である。既知のマルチプロジェクタ技術としては、Cruz-Neira他著「Surround-screen Projection-based Virtual Reality: The Design and Implementation of the CAVE」SIGGRAPH 93 Conference Proceedings, Vol. 27, pp. 135-142, 1993, Stadl他、「The blue-c: Integrating real humans into a networked immersive environment」ACM Collaborative Virtual Environments, 2000が挙げられる。

40

【0003】

電気光学技法を使用して位置決めし、ブレンディングパラメータを求める(Li他著「Optical Blending for Multi-Projector Display Wall System」Proceedings of the 12th

50

Lasers and Electro-Optics Society, 1999参照)、またはループ中のカメラを使用する (Surati著「Scalable Self-Calibrating Display Technology for Seamless Large-Scale Displays」マサチューセッツ工科大学博士論文, 1999, Chen他著「Automatic Alignment of High-Resolution Multi-Projector Displays Using An Un-Calibrated Camera」IEEE Visualization, 2000, およびYang他著「PixelFlex: A Reconfigurable Multi-Projector Display System」IEEE Visualization, 2001, Brown他著「A Practical and Flexible Large Format Display system」Tenth Pacific Conference on Computer Graphics and Applications, pp. 178-183, 2002, Humphreys他著「A Distributed Graphics System for Large Tiled Displays」IEEE Visualization, 1999, およびHumphreys他著「WireGL: A Scalable Graphics System for Clusters」Proceedings of SIGGRAPH, 2001参照)、平坦な表面上にシームレスな画像を生成するいくつかの技法が知られている。 10

【0004】

複数のプロジェクタが使用される場合、シームレスな表示を実現するために、重複する画像の幾何学的関係を正確に推定することが重要である。幾何学的関係は、レンダリングプロセスおよびソフトエッジブレンディングに影響を及ぼす。3×3行列によって表現されたホモグラフィを利用するカメラベースの方法は、面倒な手動での位置合わせをなくしながら不用意に設置されたプロジェクタを許容する。

【0005】

二次式に従う表面の関係は、二次画像伝達関数を使用して定義することができる。Shasha他著「The quadric reference surface: Theory and applications」Tech. Rep. AIM-1448, 1994を参照のこと。 20

【0006】

湾曲表面のマルチプロジェクタ位置合わせは、「ナビゲータ」パターンを投影し、それからプロジェクタの位置を手動で調整することによって補助することができる。ニューヨークのハイデンプラネタリウムで使用されるもの等、大規模なディスプレイの場合、技術者が重複する7台のプロジェクタを位置合わせするのに毎日数時間かかる。

【0007】

1つの問題は、3D画像が湾曲表面上に表示されるとき、その画像は、空間中の1つのポイントでしか遠近的に正しくないことである。この3D位置は仮想視点、または「スイートスポット」として知られている。閲覧者がスイートスポットから遠ざかるにつれ、画像は歪んで見える。表示画面が非常に大きく、多くの視点がある場合、この歪みをなくすることは難しい。しかし、現実世界での用途では、閲覧者は、プロジェクタを理想的に配置する必要がある場所とまったく同じ場所にいたいと思う。さらに、プロジェクタをスイートスポットに配置するという事は、非常に視野の広いプロジェクタを使用することを意味し、これは高価であるとともに、放射方向歪みまたは「魚眼」歪みが過度になりがちである。 30

【0008】

別の方法では、非パラメトリックプロセスにより、カメラがスイートスポットに配置される。カメラは、プロジェクタによって投影された構造化光パターンの画像を取得する。次に、試行錯誤手法で、サンプルがとられて、補間によりカメラ入力画像と投影された出力画像の間の逆ワーピング関数を構築する。次に、ワーピング関数が出力画像を正しく表示するまで、この関数が適用されてリサンプリングされる。Jarvis著「Real Time 60Hz Distortion Correction on a Silicon Graphics IG」Real Time Graphics 5, pp.6-7, February 1997, およびRaskar他著「Seamless Projection Overlaps Using Image Warping and Intensity Blending」Fourth International Conference on Virtual Systems and Multimedia, 1998を参照のこと。 40

【0009】

平坦な表面用のホモグラフィベースの手法を二次曲面に拡張する、複数のプロジェクタを位置合わせするパラメトリック的方法を提供することが望ましい。

【0010】

コンピュータビジョンでは、画像伝達関数に二次式を使用することについて、いくらか研究が行われている。上記Shashua他、およびCross他著「Quadric Surface Reconstruction from Dual-Space Geometry」Proceedings of 6th International Conference on Computer Vision, pp.25-31, 1998を参照のこと。しかし、カメラを意図した線形的方法では、以下述べるように、カメラの代わりにプロジェクタと併せて使用される場合に大きな誤差が生じる。

【0011】

マルチプロジェクタシステムでは、平坦なホモグラフィの関係を使用して平坦な表面上に画像をシームレスに位置合わせするいくつかの技法が知られている。しかし、より高次の表面上に表示された画像のパラメータ化ワーピングおよび自動位置合わせについての技法に対してはあまり研究されてこなかった。

10

【0012】

二次曲面はプロジェクタベースのディスプレイでは多くの形状および形態で見えるため、これは深刻な手抜きである。大型フライトシミュレータは従来から、円筒体形状またはドーム形状であり（Scott他著「Report of the IPS Technical Committee: Full-Dome Video Systems」The Planetarian, Vol. 28, p.25-33, 1999参照）、プラネタリウムおよびOmniMaxシアターは半球形のスクリーンを使用しており（Albin著「Planetarium special effects: A classification of projection apparatus」The Planetarian, Vol. 23, pp.12-14, 1994）、多くの仮想現実システムが円筒形状のスクリーンを使用している。

20

【発明の開示】

【発明が解決しようとする課題】

【0013】

したがって、湾曲した表示面上に投影された画像の較正方法、二次伝達関数、およびパラメトリック輝度ブレンディングを提供することが望ましい。

【課題を解決するための手段】

【0014】

湾曲した表示画面は、高解像度没入型視覚化環境にますます使用されるようになっていく。本発明は、1台または複数の重複するプロジェクタを使用して、球面または柱面等二次曲面上にシームレスな画像を表示する方法およびシステムを提供する。二次元または三次元の画像を相互作用させて表示しながら、サブピクセル位置合わせを実現する新しい二次画像伝達関数が定義される。

30

【発明を実施するための最良の形態】

【0015】

図3および図4は、1台または複数のプロジェクタ301および401を使用してシームレスな画像を凸形または凹形の湾曲表面、たとえば凹形ドーム302または凸形ドーム402上に表示する、本発明によるシステムの基本的なセットアップを示す。1台のプロジェクタが使用される場合、投影画像は表示面の大部分をカバーし（点線参照）、複数のプロジェクタが使用される場合、投影画像は破線で示すように部分的に重複する。図7は、3つの重複する複数の画像701でより大きな矩形画像710をどのように作り出すことができるかを示す。

40

【0016】

二次伝達関数

3Dでの不透明な二次曲面Qの2つの任意の透視図の間の写像は、二次伝達関数を使用して表現することができる。二次曲面は、球面、半球面、回転楕円面、ドーム、柱面、錐面、放物面、双曲面、双曲放物面、または楕円面であることができる。本発明による二次伝達関数は、二次曲面を介しての第1のビュー、たとえばプロジェクタ出力画像から第2のビュー、たとえばカメラ入力画像への画像伝達関数を意味する。平坦ホモグラフィ伝達関数は、4個以上のピクセルの対応性から求めることができるが、二次伝達関数では9個以上の対応性が必要である。二次伝達関数は、3D二次曲面Q、および2つのビューの

50

透視投影に関連するさらなるパラメータを使用して閉じた形で定義することができる。

【 0 0 1 7 】

二次曲面 Q は、 4×1 ベクトルで表され、表面上にある 3 D 同次ポイント X が、二次制約 $X^T Q X = 0$ を満たすように、 4×4 対称行列で表される。二次曲面 Q は、行列の独立要素に対応して自由度 9 を有する。この行列は対称であり、最大で全体規模まで定義される。

【 0 0 1 8 】

対応するピクセル：第 1 のビュー中の x と第 2 のビュー中の x' の同次座標は、

【 0 0 1 9 】

【 数 1 】

$$x' \cong Bx - \left(q^T x \pm \sqrt{((q^T x)^2 - x^T Q_{33} x)} \right) e$$

10

【 0 0 2 0 】

によって関連する。

【 0 0 2 1 】

図 6 に示すように、B は、ピクセル 6 0 1 が可視である、二次曲面 6 0 5 に接した平面 6 0 4 上のポイント 6 0 3 を介しての投影出力ピクセル x 6 0 1 と対応するカメラ入力ピクセル x' 6 0 2 の間の 3×3 ホモグラフィ行列である。

20

【 0 0 2 2 】

ピクセル対応性 (x, x') が与えられると、従来、この式を使用して 2 1 個の未知数：未知の 3 D 二次曲面 Q、 3×3 ホモグラフィ行列 B、および同次座標におけるエピポール e を算出する。エピポール e は、第 2 のビューにおける第 1 のビューの投影中心である。記号 \cong は、同次座標のスケールまで等しいことを表す。行列 Q は以下：

【 0 0 2 3 】

【 数 2 】

$$Q = \begin{bmatrix} Q_{33} & q \\ q^T & 1 \end{bmatrix}$$

30

【 0 0 2 4 】

のように分解される。

【 0 0 2 5 】

したがって、 Q_{33} は Q の一番上の 3×3 対称部分行列であり、 q は 3 ベクトルである。 $Q(4, 4)$ は、二次曲面が原点、すなわち第 1 のビューの投影中心を通過しない場合はゼロではない。それ故、大半の表示面に 1 . 0 を安全に割り当てることができる。同次ピクセル x' の最終的な 2 D ピクセル座標は ($x'(1) / x'(3), x'(2) / x'(3)$) である。

40

【 0 0 2 6 】

単純化

上に述べた形態は、Shashua 他、1 9 9 4 と、Wexler および Shashua 著「Q-Warping: Direct Computation of Quadratic Reference Surface」IEEE Conf. on Computer Vision and Pattern Recognition, CVPR, June, 1999 により使用されている。この形態は 2 1 個の変数を含み、これは必要な分よりも 4 個多い。本発明では、 $A = B - e q^T$ 、 $E = q q^T - Q_{33}$ を定義することによってその曖昧さの一部を取り除き、本発明において使用する形態を得る。

【 0 0 2 7 】

【数 3】

$$x' \cong Ax \pm (\sqrt{x^T Ex})e$$

【0028】

ここで、 $x^T E x = 0$ は、第 1 のビューにおける二次曲面の輪郭円錐曲線を画定する。輪郭円錐曲線は、ビューレイが局所的に表面に接する表面上のシルエットまたはポイントの画像、たとえば、球体外から見た球体の楕円形のシルエットの画像として幾何学的に視覚化することができる。

【0029】

値 A は、第 1 のビューと第 2 のビューの間の極平面を介してのホモグラフィである。この式は、全体規模から離れて、E および e の相対的なスケールリングに起因する 1 つの曖昧な自由度のみを含むことに留意する。この曖昧さは、 $E(3, 3) = 1$ 等、さらなる正規化制約を導入することによって取り除くことができる。

【0030】

さらに、平方根の前にある符号は、画像中の輪郭円錐曲線内で固定である。この符号は、対応するピクセル対の座標を使用して上記式をテストすることによって容易に求められる。二次伝達関数のパラメータは、投影座標システムにおける 9 個以上のピクセル対応性から直接算出することができることに留意する。このため、ユークリッドパラメータをいずれも算出することなく、平坦なディスプレイの場合に平坦なホモグラフィを推定するのと同様の手法に従いたくなる。しかし、以下述べるように、実際には、多くの場合でエピポール関係を推定することは難しい。したがって、本発明では、以下述べるように擬似ユークリッド手法を使用する。

【0031】

事前処理

すべての位置合わせ情報は、立体画像対に対して予め計算される。ここでは、立体カメラが 3D 表示面全体を見ているものと仮定する。カメラ画像の 1 つが、原点 c_0 を有するものとして任意に選択される。立体画像は、表示面上の 3D ポイントのみを求めるために使用される。したがって、任意の適した 3D 取得システムを使用することができる。

【0032】

図 1 に示すように、本発明の事前処理方法 100 の基本ステップは以下のようなものである。これらステップの詳細について以下に述べる。各プロジェクタ i 毎に、所定の画像 101、たとえば、チェッカー盤の形態の構造化パターンが二次曲面上に投影される (110)。

【0033】

ステップ 120 において、立体カメラ 102 によって得られた所定の画像 101 の画像の特徴が検出され、3D ポイント、すなわち所定の画像の特徴に対応する二次曲面上の対応性 103 が再構築される。次に、二次曲面 Q 104 が検出された対応性に近似される (130)。

【0034】

各プロジェクタ i 毎に、カメラに対するプロジェクタのポーズ (pose) 105 が、プロジェクタピクセルとピクセルによって照らされた表面上のポイントの 3D 座標との間の対応性を使用して求められる (140)。カメラ c_0 とプロジェクタ i の間の二次伝達関数 I_i および逆二次伝達関数 I_i^{-1} を求める (150)。次に、投影画像が重複する領域における輝度ブレンディングウェイト w_i を求める (160)。この時点で、ワーピング、ブレンディング、および投影により、歪みのない出力画像 171 として見えるプロジェクタ画像を二次曲面上に投影することができる (200)。

【0035】

ランタイム処理

図 2 は、本発明のレンダリング方法 200 の基本ステップを示す。まず、投影画像が生

10

20

30

40

50

成される(210)。プロジェクタ画像は、三角形のテクスチャマッピングセットとして表される。この画像は、立体カメラまたはビデオカメラによって取得することもでき、または仮想視点から3Dシーンをコンピュータグラフィックレンダリングすることによって生成することもできる。プロジェクタ画像は、二次伝達関数 o_i に従ってプロジェクタのフレームの中にワーピングされる(220)。次に、ピクセル輝度がブレンディングウェイト i で減衰され(230)、それから投影される(240)。

【0036】

しかし、これらステップには、解決する必要があるいくつかの問題が含まれる。二次伝達関数の推定は線形演算であるが、ピクセルの再投影誤差を低減するために非線形最適化を必要とする。さらに、二次曲面上に投影された3Dポイントは通常、略平坦であり、劣化条件につながるため、プロジェクタのポーズ、すなわち外部パラメータを推定することは難しい。これらおよび他の問題、ならびに実際の解決策について以下に述べる。

10

【0037】

較正(calibration: キャリブレーション)

本発明では、投影出力画像が湾曲表面上で幾何学的に位置合わせされるような二次伝達関数 $o_i = \{A_i, E_i, e_i\}$ のパラメータを求める。カメラについて既知である従来技術の方法では、ピクセルの対応性から直接、二次伝達パラメータを求める。これは、対応するピクセルの三角形分解および線形的方法を使用して3Dにおける 4×4 二次行列 Q を推定することを含む。2つのビューの内部パラメータがわかっていない場合、すべての計算は、エピポルジオメトリ、すなわちエピポールおよび基本行列を算出した後、投影空間において行われる。

20

【0038】

しかし、カメラではなくプロジェクタが関わる場合、線形的方法では、3D二次曲面 Q を推定するにあたり非常に大きな再投影誤差がもたらされる。この誤差は、従来のXGAプロジェクタの場合に約30ピクセルのオーダのものである。

【0039】

これにはいくつかの理由がある。取得した画像から直接カメラを較正することは比較的単純なことである。しかし、プロジェクタの較正は、カメラによって得られた投影画像の画像を解析することによって間接的にしか行うことができない。これは誤差をもたらす。さらに、プロジェクタでは、内部光学系は通常、画像を上向きに投影する原則的なポイントからずれている。したがって、プロジェクタの内部パラメータは推定が難しく、カメラのものと異なる。さらに、1台のプロジェクタによって照らされている二次曲面上の3Dポイントが、球面または柱面のセグメント等、表示設定にあまり深さのバリエーションを持たない場合、基本行列には本質的にノイズが多い。

30

【0040】

したがって、本発明は、カメラおよびプロジェクタの内部パラメータおよび外部パラメータがおおよそわかっている擬似ユークリッド手法を使用する。これらパラメータは、厳格なユークリッド変換の推定に使用される。したがって、平坦な場合と異なり、湾曲した画面の場合の正確な画像伝達関数の算出には、三次元量に関わる。

【0041】

二次曲面

本発明では、すべての幾何学的形状の関係を算出するベースとして、精密な立体カメラ対 C_0 および C'_0 を使用する。カメラの1つを任意に選択して、原点および座標系を定義する。チェッカー盤パターン101を使用して、ベースとなる立体カメラ対を較正する。本発明の較正では、カメラは必ずしもスイートスポットに配置する必要はなく、これは従来技術による非パラメトリック手法のいくつかに対する重要な相違である。

40

【0042】

立体カメラ対102は、各プロジェクタによって投影された構造化パターンを観察する。三角形分解を利用して、パターン特徴に対応する、表示面上の N 個の3Dポイントセット $\{X_j\}$ が検出される。各 X_j を通過する二次曲面 Q が、各3Dポイント毎に線形式の

50

セット $X^T j Q X_j = 0$ を解くことによって算出される。この式は $x_i V = 0$ の形態で書くことができ、式中、 x_i は X のみの関数である 1×10 行列であり、 V は二次曲面 Q の別個の独立した未知変数を含む同次ベクトルである。N 9 を使用して、 $N \times 10$ 行列 X を構築し、線形行列式 $X V = 0$ を解く。

【0043】

一般的な位置にあるポイントの場合、 V ひいては Q の要素は行列 X の一次元ヌル空間である。

【0044】

プロジェクタビュー

二次曲面 Q に加えて、本発明では、カメラの原点に対する各プロジェクタの内部パラメータおよび外部パラメータを知る必要がある。本発明では、プロジェクタピクセルと、プロジェクタピクセルが照らす 3D ポイントの座標との間の対応性を用いて、ポーズおよび内部パラメータを算出する。

【0045】

しかし、二次曲面上の既知の 3D ポイントからプロジェクタのポーズを見つけることは、3D ポイントが通常、平面にかなり近く、解が不安定になることから誤差が出やすい。Faugeras, Three-dimensional computer vision: a geometric viewpoint, MIT Press, 1993、および Forsyth 他著「Computer Vision, A Modern Approach」FUTURESLAB, 2002, ActiveMural, Argonne National Labs, 2002 を参照のこと。

【0046】

略平坦なポイントへの対処は難しい問題である。ポイントの深さが分散している場合、線形的方法を使用してプロジェクタの内部パラメータならびに外部パラメータを容易に推定することができる。一方、ポイントが平坦であるとわかっている場合、内部パラメータのいくつかはわかっているときに、外部パラメータを推定することができる。

【0047】

略平坦な表面に対処するために、本発明では、反復手順を用いる。プロジェクタの内部パラメータがわかっている場合、まず、ホモグラフィに基づいて外部パラメータの初期推定を見つけ、それから Lu 他著「Fast and globally convergent pose estimation from video images」IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22:6, pp.610-622, 2000 に基づいて反復手順を使用することができる。Powell の方法は、再投影誤差の非線形的改良に使用することができる。しかし、プロジェクタの内部パラメータを推定することも等しく難しい。上に述べたように、プロジェクタを容易に動かすことができない場合、プロジェクタの較正では通常、2 つ以上の位置で照らされている大きな表面が必要である。

【0048】

本発明の戦略は、おおよそわかっているプロジェクタの内部パラメータを使用するというものである。本発明では、1 台のみのプロジェクタの内部パラメータを見つけ、これら内部パラメータをすべてのプロジェクタに対して使用する。後で、同じおよび他のプロジェクタはおそらく、異なるズーム設定を有し、また他の機械的または光学的偏差を有することになる。さらに、Lu 他の反復方法によって算出される外部パラメータも近似にすぎない。

【0049】

カメラからプロジェクタへの伝達

したがって、本発明では、プロジェクタのおおよその投影行列とともに、カメラの透視投影パラメータを使用し、線形的方法を使用してカメラからプロジェクタへの二次伝達を見つける。次に、非線形最適化を使用して解を改良する。

【0050】

二次伝達パラメータ $o_i = \{ A_i, E_i, e_i \}$ は、二次曲面 Q 、カメラ投影行列 $[I | 0]$ 、およびプロジェクタ投影行列 $[P_i | e_i]$ から、

【0051】

10

20

30

40

50

【数 4】

$$A_i = P_i - e_i q^T \quad E_i = q q^T - Q_{33}$$

【0052】

により容易に計算される。

【0053】

上に述べたように、パラメータを見つける従来技術による線形的方法は、本目的には不正確すぎる。表示画面上での位置合わせずれはプロジェクタ1台あたり15～30ピクセル程度になる恐れがあり、不快な視覚的アーチファクトになる。シームレスな表示にはサブピクセルの精度が必要である。したがって、本発明では、非線形最小化を適用して、線形的方法を介して得られた結果を改良する。これは、目的関数または「費用」関数を使用して行うことができる。

10

【0054】

目的関数の場合、すべてのピクセルの総二次伝達誤差をとる。

【0055】

【数 5】

$$\varepsilon_i = \sum_j \left(\frac{x_i^j(1,2)}{x_i^j(3)} - \frac{\hat{x}_i^j(1,2)}{\hat{x}_i^j(3)} \right)^2$$

20

【0056】

式中、

【0057】

【数 6】

$$\hat{x}_i^j$$

【0058】

は、既知の各投影ピクセル x_j の伝達後のピクセル、すなわち、

30

【0059】

【数 7】

$$\hat{x}_i^j = A_i x_i^j \pm \sqrt{x_i^{jT} E_i x_i^j e_i}$$

【0060】

と表されるパターン特徴ポイントである。

【0061】

線形的方法を使用して見つけられる符号は、すべてのピクセルで同じであり、非線形最適化、たとえば、既知のネルダーモード (Nelder-Mead) シンプレックス中同じままであることに留意する。

40

【0062】

部分的なユークリッド再構築

ユークリッド手法をすべて無視し、投影空間および非線形最適化に直接進んでもよい。正確なプロジェクタ内部パラメータを有する場合、非線形最適化段階を回避してもよい。しかし先に触れたように、ユークリッドビューパラメータを無視し、二次伝達関数を純粹にピクセルの対応性から解くと、不良な再投影誤差が生じる。さらに、推定された3D二次曲面Qは、解が収束しないため、それ以降の非線形最適化の推定として使用することができない。

【0063】

50

正確な内部プロジェクタパラメータは、再投影誤差を低減するだけであり、誤差をなくしはしない。これは、三角形分解による表示面上の3Dポイントの推定、線形的方法を使用しての3D二次曲面の推定、およびプロジェクタのポーズの発見を含め、多くの種類の誤差が三次元ユークリッド計算において伝搬するためである。非線形最適化は、最も注意を払う物理量、すなわち、既知の対応するピクセルセットのカメラからプロジェクタへ伝達される画像中のピクセル再投影誤差、を最小化しようと試みるものである。

【0064】

重複するプロジェクタピクセル間の対応性は、この画像伝達関数によって間接的に定義されるため、ピクセル再投影誤差を最小化することにより、表示されるプロジェクタのピクセル間の幾何学的位置合わせが確実にされる。

10

【0065】

レンダリング

レンダリングは2パス手法を含む。2D画像データの場合、しかるべき入力画像を抽出する。3Dシーンの場合、まず、スイートスポットに従って3Dモデルをレンダリングする。第2のパスでは、結果得られた画像が次に、二次画像伝達関数を使用してプロジェクタ画像空間にワーピングされる。

【0066】

仮想ビュー

3Dシーンが湾曲した画面上に表示されるとき、画像は、空間中の特定のポイントからしか遠近的に正しくない。このようなポイントは一般に、スイートスポットまたは仮想視点として知られている。凹形半球面の場合、スイートスポットは、球体を半分に切断する切断面に垂直な、球体の中心を通るライン上にある。凸形半球面の場合、スイートスポットは表面を通して鏡映することができる。柱面の場合、スイートスポットは、投影画像が表面全体に覆われていない限り同様に配置され、投影画像が表面全体に覆われている場合、スイートスポットはいずれの場所であってもよい。スイートスポットは、カメラの焦点距離と無関係であるが、湾曲したディスプレイの形状およびサイズ、ならびに周囲の視角に依存する。投影画像は、スイートスポットの場所、視覚カメラの焦点距離、または閲覧者に応じて表示面の前または後ろに現れうる。閲覧者がスイートスポットから離れるにつれ、画像は歪んで見える。さらに、視錐台、すなわち見ている方向または主軸、および視野範囲を指定する必要がある。

20

30

【0067】

ディスプレイによっては、スイートスポットを自動的に決定することが可能である。たとえば、凹形半球形ドームの場合、ドームの中心を良好なスイートスポットとみなすことができる。スイートスポットは、二次曲面 Q の式、すなわち $Q(1, 1)q$ から直接求めることができる。円筒形の画面の場合、円筒体の広がりによって中間にある円筒体の軸上のポイントが良い選択である。時により、スイートスポットは、実際の考慮事項によって決まる。たとえば、画像が現実世界の水平軸および垂直軸に略常に位置合わせされているため、おおよそ目の高さにあるスポットが理想的であるとみなされる。

【0068】

本発明の場合では、ディスプレイジオメトリを近似したユークリッド再構築であるため、仮想視点は相互作用的に固定または移動させることができる。ロケーションの3D座標はカメラ座標系においてわかっているため、カメラに対してスイートスポットを見つけるのは比較的単純である。

40

【0069】

表面ポイントからのスイートスポット

レンダリングのために仮想視点のパラメータを求めることが難しい場合、一技法により、表示面の照明部分上で見つけられるポイントセットに最も近似した平面が見つけられる。有向バウンディングボックス(OBB)を表示面上の3Dポイントセット $\{X_i\}$ に近似する。このボックスの中心を通過し、最良近似平面に垂直なベクトルに沿って、画面の前からしかるべき距離にあるポイントをスイートスポットとして選択することができる。

50

二次曲面上の3Dポイントはすべて3Dポイントの凹形外殻に存在するため、OBBは以下のように求めることができる。

【0070】

Yが、

【0071】

【数8】

$$\{X_i - \bar{X}\}$$

【0072】

の $N \times 3$ 行列である場合(式中、

【0073】

【数9】

$$\bar{X} = (\bar{x}, \bar{y}, \bar{z})$$

【0074】

は N 3Dポイントの中心である)、 3×3 行列 $Y^T Y$ の最小固有値に対応する固有ベクトルにより、最良近似平面への法線、すなわち分散が最小の軸が与えられる。一方、OBBの最大辺、すなわち、最良近似平面に投影された3Dポイントの広がりにより、画面のおおよその「直径」が与えられる。画面前のスイートスポットの距離は、用途および所望の視野に応じて、この直径に比例するように選択することができる。

【0075】

没入形用途の場合、広い視野が望ましい。したがって、この距離は直径の約半分であるべきである。グループ閲覧の場合、この距離は直径と同等であることができる。

【0076】

本発明では、仮想ビュー画像空間と各プロジェクタ出力画像の間の二次伝達関数 i を再計算する。このプロセスは i_0 の算出に非常によく似ている。まず、二次曲面上の3Dポイントの仮想ビュー画像空間への投影を見つける。ここで、仮想ビューとプロジェクタ画像の間の対応性、ならびに仮想ビューおよびプロジェクタの内部パラメータおよび外部パラメータで i を更新するのに十分である。

【0077】

表示エリア

仮想ビューの視錐台は、スイートスポットおよびOBBの広がりを使用して画定される。視点ベクトルは、仮想視点からOBBの中心に向けられる。複数のプロジェクタからの重複画像の連合体は大きなエリアを照らしうるため、視錐台を矩形または円形等、見た目に美しい形状に「クロッピング」することができる。3D用途の場合、黒色の四辺形のセットをレンダリングして、所望の表示エリア外の領域をクロッピングする。たとえば、矩形ビューの場合、プロジェクタ画像中のビューポート外縁の付近の4つの大きな四辺形により、ビューポートが作成される。黒色四辺形は、3Dモデルの残りの部分とともに、以下のようにレンダリングされワーピングされる。2D用途の場合、表示すべき入力画像外のエリアは黒色であるとみなされる。

【0078】

画像の伝達

レンダリングするシーン中の3D頂点Mが与えられる場合、仮想ビューにおけるその画面空間座標 m を見つける。次に、二次伝達関数 $o_i = \{A_i, E_i, e_i\}$ を使用してプロジェクタ i の出力画像中の伝達後のピクセル座標 m_i を見つける。次に、頂点Mが頂点 m_i で置き換えられて、シーン中のポリゴンがレンダリングされる。したがって、各プロジェクタでのレンダリングプロセスは非常に似ている。各プロジェクタのフレームバッファが、仮想ビュー画像のしかるべき部分を自動的に格納し、プロジェクタの範囲を明示

10

20

30

40

50

的に決定する必要はない。

【0079】

したがって、レンダリングするには、各プロジェクタにおいて、各頂点M毎に、仮想ビュー投影(M)を介してピクセルmを求め、さらに二次伝達関数 $t_i(m)$ を介してワーピングされたピクセル m_i を求め、そして頂点 $\{M_j\}$ を有する各三角形T毎に、2D頂点 $\{m_{j_i}\}$ を有する三角形をレンダリングする。

【0080】

この手法には2つの問題がある。第1に、ポリゴン内部ではなく、シーン中の頂点のみが正確にワーピングされる。第2に、ポリゴンの可視性分類に特別な処理が必要である。二次伝達後、理論的に、ポリゴンの頂点間のエッジをプロジェクタ画像中の二次曲線に写像する必要がある。 10

【0081】

しかし、走査変換により、曲線をワーピングされた頂点位置間の直線の線分に変換する。この問題は1台のプロジェクタの場合は認められない。しかし、重複するプロジェクタの場合、これは、元の曲線からの個々の異なる逸脱につながり、ひいては、エッジが表示画面上で位置合わせがずれて現れる。したがって、三角形の十分に細かいテッセレーション(tessellation)を用いる必要がある。

【0082】

湾曲した画面上での遠近的に正しいレンダリングにおいて入力モデルが真っ直ぐに現れるように、入力モデルをオンザフライで分割し予め歪ませる市販のシステムをすでに利用することができる。参照により本明細書に援用される1997年2月26日にIdaszak他に付与された米国特許第6,104,405号「Systems, methods and computer program products for converting image data to non-planar image data」、および1994年6月7日にKelly他に付与された米国特許第5,319,744号「Polygon fragmentation method of distortion correction in computer image generating systems」を参照のこと。本発明の方法は、このようなシステムによって提供される細かいテッセレーションと両立できる。市販のシステムにおけるシーンジオメトリの事前歪みは、第1のパスのテクスチャマッピング結果を含む2パスレンダリングを回避するために使用される。本発明の場合、ジオメトリを予め歪ませる代わりに、画像空間投影を予め歪ませる。利点として、本発明は、プログラマブルグラフィクスユニット(GPU)のパーテックスシェーダを使用して一部実施することができる。 20 30

【0083】

走査変換

三角形の投影におけるピクセル位置がワーピングされる時、深度バッファがしかるべき可視性情報を生成するように、情報も共に渡す必要がある。さらに、遠近的に正しい色およびテクスチャ座標補間のために、しかるべき重み値「w」を渡す必要がある。したがって、

$m(x, y, z, w) = \text{仮想視点投影}(M(X))$

$m_i(x'_i, y'_i, w'_i) = t_i(m(x/w, y/w), 1)$ 40

$m_i(x_i, y_i, z_i, w_i) = [wx'_i/w'_i, wy'_i/w'_i, z, w]$

に従ってピクセル座標に「w」を後ろから乗算する。

【0084】

したがって、頂点 m_i は、二次伝達関数とともに元の深度およびw値により、しかるべき最終ピクセル座標 $(x'_i/w'_i, y'_i/w'_i)$ を有する。

【0085】

図5は、パーテックスシェーダのコード500を示す。2D画像をレンダリングする場合、仮想ビュー画像空間を三角形に細かく分割し、画像をこれら三角形上のテクスチャとして写像する。各三角形の頂点mは、上記のように、二次伝達関数を使用して頂点(およびピクセル) m_i にワーピングされる。走査変換は、頂点mにおける色属性およびテクス 50

チャ属性を頂点 m_i に自動的に変換し、その間を補間する。同様にして 3 D シーンをレンダリングすることが可能である。

【 0 0 8 6 】

本発明の二次伝達関数を使用してのプロジェクト画像のワーピングは、平坦な表面上での二次曲線のレンダリングと異なることに留意する。Watson他著「A fast algorithm for rendering quadratic curves on raster displays」Proc. 27th Annual SE ACM Conference, 1989を参照のこと。

【 0 0 8 7 】

輝度レンディング

重複画像のエリアにおけるピクセル輝度は、グラフィックスハードウェアのアルファブレンディングを使用して減衰される。二次伝達関数の助変数方程式を使用して、アルファマップが以下のようにして求められる。

【 0 0 8 8 】

プロジェクト i におけるあらゆるプロジェクトピクセル x_i 毎に、式：

【 0 0 8 9 】

【 数 1 0 】

$$x_i \cong \Psi_{0j}(\Psi_{0i})^{-1}(x_i)$$

【 0 0 9 0 】

を使用してプロジェクト j において対応するピクセルを見つける。

【 0 0 9 1 】

クロスフェードの場合、プロジェクト画像の境界にあるピクセルが減衰される。したがって、重みはフレーム境界からの最短距離に比例する。範囲 $[0, 1]$ にある、正規化されたウィンドウピクセル座標 (u_i, v_i) で表されるピクセル x_i に割り当てられる重みは、

【 0 0 9 2 】

【 数 1 1 】

$$\Phi_i(x_i) \cong d(x_i) / \sum_j d(x_j)$$

【 0 0 9 3 】

である。式中、 $d(x)$ は、 $0 \leq u, v \leq 1$ の場合、 $\min(u, v, 1-u, 1-v)$ であり、その他の場合、 $d(x) = 0$ である。パラメトリック手法を使用するため、対応するプロジェクトピクセルおよびそれらの場所における重みをサブピクセル位置合わせ精度で算出することが可能である。対応するプロジェクトピクセルにおける重みの和は決して正確に 1.0 になる。

【 0 0 9 4 】

各プロジェクトにおいて、対応するアルファマップはテクスチャマップとして格納され、レンダリングの最後の段階で画面に位置合わせされた四辺形としてレンダリングされる。

【 0 0 9 5 】

本発明の方法は、凹形ドーム上への投影にも使用することができる。この投影は、ドームが半透明または透明な材料からできている場合に特に有用である。後ろから投影される場合、閲覧者は、プロジェクトをいずれも遮ることなく完全に没入型の体験を得ることができる。

【 0 0 9 6 】

発明の効果

本発明は、2 D または 3 D 視覚化に湾曲した表示面を使用してプロジェクトシステムを構築することができる。本発明のシステムでは、高価なインフラストラクチャが必要なく、複数のプロジェクトと表示面との間の位置合わせが不用意な状態で動作することができ

10

20

30

40

50

る。本発明の自動位置合わせは、二次画像伝達関数を利用し、プロジェクタの面倒なセットアップおよびメンテナンスをなくし、したがってコストを低減する。本発明は、フライトシミュレータ、プラネタリウム、およびハイエンドビジュアライゼーションシアター等で使用されるなど、広く使用されている用途の構築、較正、およびレンダリングプロセスを単純化することができる。可能になる新しい用途としては、低コストで柔軟性のあるドームディスプレイ、ショッピングアーケード、および円柱または支柱への投影が挙げられる。

【0097】

本発明は、これまで離散サンプリングによって解決されてきた問題に対する洗練された解決策を提供する。利点は、従来技術によるシステムと異なり、本発明のプロジェクタはスイートスポットに配置する必要がないことである。これは、スイートスポットが通常、閲覧者がいたいと思うその場所である現実世界で適用の際に重要である。

10

【0098】

本発明について好ましい実施形態例として説明したが、他の様々な適合および変更を本発明の精神および範囲内で行いうることを理解されたい。したがって、添付の特許請求の範囲の目的は、本発明の真の精神および範囲内にあるこのような変形および変更をすべて保護範囲に包含することである。

【図面の簡単な説明】

【0099】

【図1】本発明による、湾曲表面上に画像を投影する方法において使用される事前処理ステップの流れ図である。

20

【図2】本発明による、湾曲表面上に画像を投影する方法において使用されるレンダリングステップの流れ図である。

【図3】本発明によるマルチプロジェクタシステムの図である。

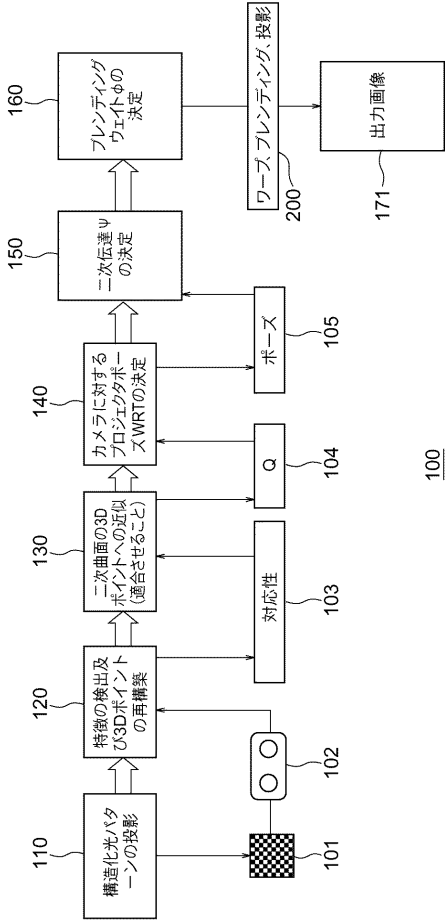
【図4】本発明によるマルチプロジェクタシステムの図である。

【図5】本発明による二次伝達関数のためのバーテックスシェーダのコードである。

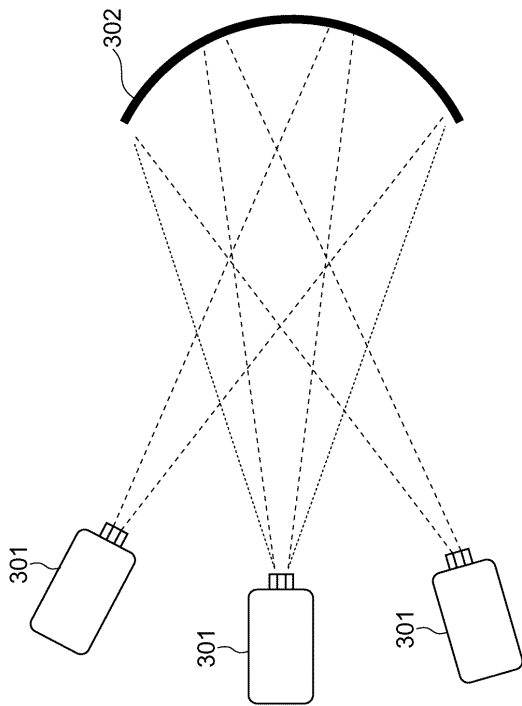
【図6】本発明によるホモグラフィを示す図である。

【図7】複数の重複画像のブロック図である。

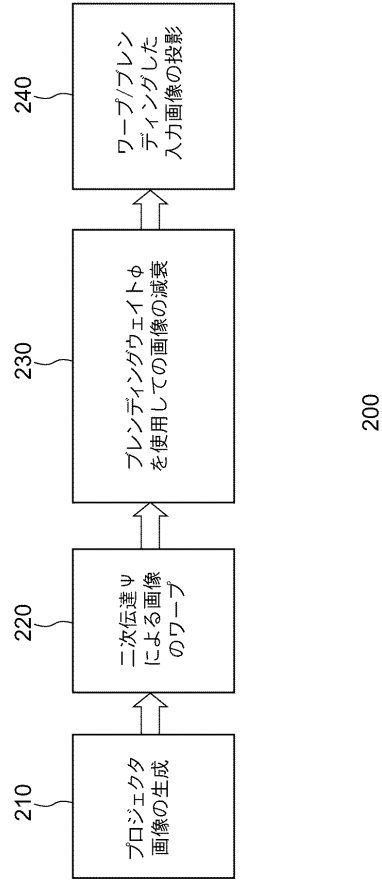
【 図 1 】



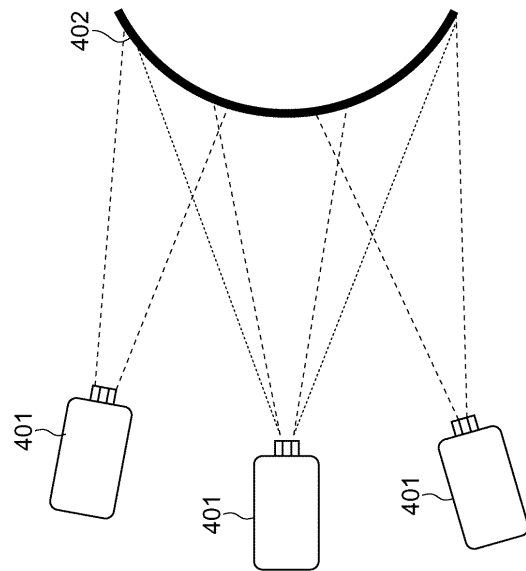
【 図 3 】



【 図 2 】



【 図 4 】



【 図 5 】

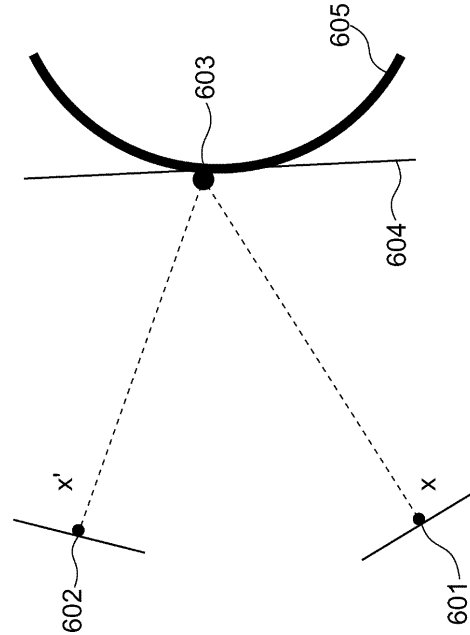
```

二次伝達関数のためのバッチシェーダのコード
vertout main( appin IN, uniform float4x4 modelViewProj, uniform float4
constColor, uniform float3x3 A, uniform float3x3 E, uniform float3 e ) {
vertout OUT;
float4 m1 = float4(IN.position.x, IN.position.y, IN.position.z, 1.0f);
float4 m, mi; float3 m2, mp; float scale;
m = mul( modelViewProj, m1);
m2.x = m.x/m.w; m2.y = m.y/m.w; m2.z = 1;
scale = mul(m2, mul(E, m2));
mp = mul(A, m2) + sqrt(scale)*e;
mi.x = m.w * (mp.x)/(mp.z);
mi.y = m.w * (mp.y)/(mp.z);
mi.zw = m.zw;
OUT.position = mi;
OUT.color0 = IN.color0; //元の頂点ごとに指定した色を使用する
return OUT;
}

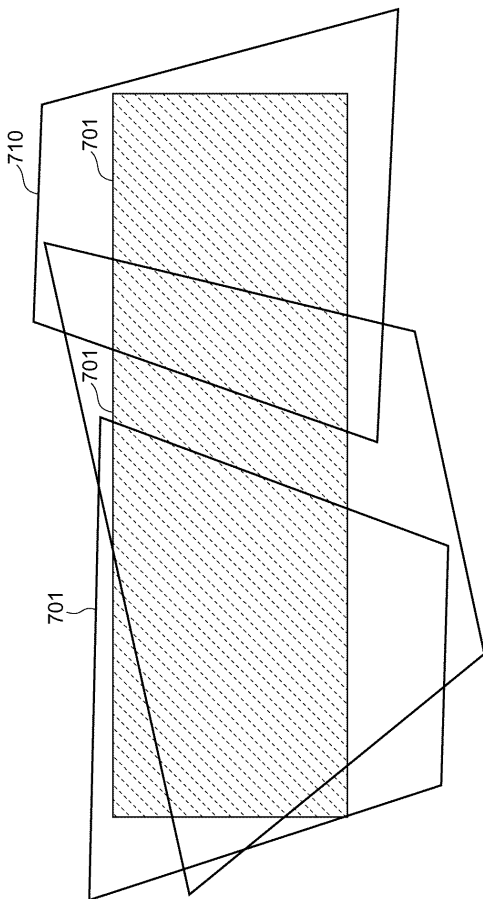
```

500

【 図 6 】



【 図 7 】



フロントページの続き

(74)代理人 100094695

弁理士 鈴木 憲七

(74)代理人 100111648

弁理士 梶並 順

(72)発明者 ラメッシュ・ラスカー

アメリカ合衆国、マサチューセッツ州、ケンブリッジ、エイッス・ストリート 48

(72)発明者 ジェロン・ヴァン・パール

アメリカ合衆国、マサチューセッツ州、ブルックライン、ビーコン・スリート 1815、ナンバー3

(72)発明者 スリニヴァサ・ジー・ラオ

アメリカ合衆国、メリーランド州、ハイアッツビル、ハイランド・ドライブ 9006

(72)発明者 トーマス・ハー・ヴィルヴァッヘル

ドイツ連邦共和国、56470、バート・マリーンベルク、セバスチアン - クナイプ・ストラッセ 9

Fターム(参考) 2K103 AA16 AA18 AA27 AA28 CA02 CA38

【外国語明細書】

Title of the Invention

Method and System for Projecting Image onto Display Surface

Detailed Description of Invention

Field of the Invention

This invention relates generally to projecting images, and more particularly to projecting images onto curved surfaces.

Background of the Invention

Projector systems have been used to render large images onto display surfaces. With multiple projectors, it is possible to generate even larger seamless displays. Such systems are particularly useful for constructing immersive visualization environments capable of presenting high-resolution images for entertainment, education, training, and scientific simulation. Known multi-projector technologies include Cruz-Neira et al., “*Surround-screen Projection-based Virtual Reality: The Design and Implementation of the CAVE*,” SIGGRAPH 93 Conference Proceedings, Vol. 27, pp. 135–142, 1993, Staadt et al., “*The blue-c: Integrating real humans into a networked immersive environment*,” ACM Collaborative Virtual Environments, 2000.

A number of techniques are known for generating seamless images on planar surfaces using electro-optical techniques to determine registration and blending parameters, see Li et al., “*Optical Blending for Multi-Projector Display Wall*

System,” Proceedings of the 12th Lasers and Electro-Optics Society, 1999, or using a camera in a loop, Surati, “*Scalable Self-Calibrating Display Technology for Seamless Large-Scale Displays*,” Ph.D. Thesis, Massachusetts Institute of Technology, 1999, Chen et al., “*Automatic Alignment of High-Resolution Multi-Projector Displays Using An Un-Calibrated Camera*,” IEEE Visualization, 2000, and Yang et al., “*PixelFlex: A Reconfigurable Multi-Projector Display System*,” IEEE Visualization, 2001, Brown et al., “*A Practical and Flexible Large Format Display system*,” Tenth Pacific Conference on Computer Graphics and Applications, pp. 178–183, 2002, Humphreys et al., “*A Distributed Graphics System for Large Tiled Displays*,” IEEE Visualization, 1999, and Humphreys et al., “*WireGL: A Scalable Graphics System for Clusters*,” Proceedings of SIGGRAPH, 2001.

When multiple projectors are used, an accurate estimation of the geometric relationship between overlapping images is key for achieving a seamless display. The geometric relationship influences the rendering process and soft edge blending. Camera-based methods, which exploit a homography expressed by a 3 by 3 matrix, admit casually installed projectors while eliminating cumbersome manual alignment.

The relationship for surfaces that adhere to quadric equations can be defined using a quadric image transfer function, see Shashua et al., “*The quadric reference surface: Theory and applications*,” Tech. Rep. AIM-1448, 1994.

Multi-projector alignment for curved surfaces can be aided by projecting a ‘navigator’ pattern and then manually adjusting the position of the projectors. For a

large scale display, such as used at the Hayden Planetarium in New York, it takes technicians several hours each day to align seven overlapping projectors.

One problem is that when 3D images are displayed on a curved screen, the images are perspective correct from only a single point in space. This 3D location is known as the virtual viewpoint or 'sweet-spot'. As the viewer moves away from the sweet-spot, the images appear distorted. For very large display screens and many view points, it is difficult to eliminate this distortion. However, in real-world applications, viewers would like to be at the exact same place where the projectors ideally need to be located. In addition, placing projectors at the sweet-spot means using a very wide-field of view projectors, which are expensive and tend to have excessive radial or 'fish-eye' distortion.

In another method, a *non-parametric* process places a camera at the sweet-spot. The camera acquires an image of a structured light pattern projected by the projector. Then, in a trial-and-error approach, *samples* are taken, to build an inverse warping function between a camera input image and a projected output image by means of interpolation. Then, the function is applied, and resampled until warping function correctly displays the output image, see Jarvis, "*Real Time 60Hz Distortion Correction on a Silicon Graphics IG,*" *Real Time Graphics 5*, pp. 6-7, February 1997, and Raskar et al., "*Seamless Projection Overlaps Using Image Warping and Intensity Blending,*" *Fourth International Conference on Virtual Systems and Multimedia*, 1998.

It is desired to provide a parametric method for aligning multiple projectors that extends the homography-based approach for planar surfaces to quadric surfaces.

In computer vision, some work has been done on using quadric formulations for image transfer functions, see Shashua et al., above, and Cross et al., “*Quadric Surface Reconstruction from Dual-Space Geometry*,” Proceedings of 6th International Conference on Computer Vision, pp. 25–31, 1998. However, the linear methods intended for cameras, as described below, produce large errors when used with projectors, instead of cameras.

In multi-projector systems, several techniques are known for aligning images seamlessly on flat surfaces using planar homography relationships. However, there has been little work on techniques for parameterized warping and automatic registration of images displayed on higher order surfaces.

This is a serious omission because quadric surfaces do appear in many shapes and forms in projector-based displays. Large format flight simulators have traditionally been cylindrical or dome shaped, see Scott et al., “*Report of the IPS Technical Committee: Full-Dome Video Systems*,” The Planetarian, Vol. 28, p. 25–33, 1999, planetariums and OmniMax theaters use hemispherical screens, Albin, “*Planetarium special effects: A classification of projection apparatus*,” The Planetarian, Vol. 23, pp. 12–14, 1994, and many virtual reality systems use a cylindrical shaped screen.

Therefore, it is desired to provide calibration methods, quadric transfer functions, and parametric intensity blending for images projected onto a curved display surface.

Summary of the Invention

Curved display screens are increasingly being used for high-resolution immersive visualization environments. The invention provides a method and system for displaying seamless images on quadric surfaces, such as spherical or cylindrical surfaces, using a single or multiple overlapping projectors. A new quadric image transfer function is defined to achieve sub-pixel registration while interactively displaying two or three-dimensional images.

Detailed Description of the Preferred Embodiment

Figures 3 and 4 shows the basic set-up for a system according to the invention that uses one or more projectors 301 and 401 to display a seamless image on a convex or concave curved surface, for example, a concave dome 302 or a convex dome 402. If a single projector is used the projected image covers most of the display surface, see dotted lines, and if multiple projectors are used, the projected images partially overlap as shown by dashed lines. Figure 7 shows how three multiple overlapping images 701 can produce a larger rectangular image 710.

Quadric Transfer Function

A mapping between two arbitrary perspective views of an opaque quadric surface, Q , in 3D can be expressed using a quadric transfer function, Ψ . The quadric surface can be a sphere, hemi-sphere, spheroid, dome, cylinder, cone, paraboloid, hyperboloid, hyperbolic paraboloid, or an ellipsoid. The quadric transfer function according to our invention means an image transfer function from a first view, e.g. a projector output image, to a second view, e.g., a camera input image, via a quadratic surface. While planar homography transfer functions can be determined from four or more pixel correspondences, the quadric transfer function requires

nine or more correspondences. The quadric transfer function can be defined in a closed form using the 3D quadric surface, Q and additional parameters that relate perspective projection of the two views.

The quadric surface, Q , is represented by a 4×4 symmetric matrix, such that 3D homogeneous points X , expressed as a 4×1 vector, that lie on the surface satisfy the quadratic constraint, $X^T Q X = 0$. The quadric surface, Q , has nine degrees of freedom corresponding to the independent elements of the matrix. The matrix is symmetric and defined up to an overall scale.

The homogeneous coordinates of corresponding pixels, x in the first view and x' in the second view are related by

$$x' \cong Bx - \left(q^T x \pm \sqrt{((q^T x)^2 - x^T Q_{33} x)} \right) e.$$

As shown in Figure 6, B is a 3×3 homography matrix between the projected output pixel x 601 and the corresponding camera input pixel x' 602 via point 603 on a plane 604 tangential to the quadric surface 605 where the pixel 601 is visible.

Given pixel correspondences (x, x') , this equation is traditionally used to compute the 21 unknowns: the unknown 3D quadric surface Q , the 3×3 homography matrix B , and an epipole, e , in homogeneous coordinates. The epipole, e , is the center of projection of the first view in the second view. The symbol \cong denotes equality up to scale for the homogeneous coordinates. The matrix Q is decomposed as follows

$$Q = \begin{bmatrix} Q_{33} & q \\ q^T & 1 \end{bmatrix}.$$

Thus, Q_{33} is the top 3×3 symmetric submatrix of Q , and q is a three-vector. $Q(4, 4)$ is non-zero if the quadric surface does not pass through the origin, i.e., the center of projection of the first view. Hence, it can be safely assigned to be 1.0 for most display surfaces. The final 2D pixel coordinate for homogeneous pixel x' is $(x'(1)/x'(3), x'(2)/x'(3))$.

Simplification

The form described above is used by Shashua et al., 1994, and Wexler and Shashua, “*Q-warping: Direct Computation of Quadratic Reference Surfaces*,” IEEE Conf. on Computer Vision and Pattern Recognition, CVPR, June, 1999. That form includes 21 variables, 4 more than needed. We remove part of that ambiguity by defining

$$A = B - eq^T \quad E = qq^T - Q_{33}, \text{ and obtain the form used by our invention,}$$

$$x' \cong Ax \pm \left(\sqrt{x^T E x} \right) e.$$

Here, $x^T E x = 0$ defines the outline conic of the quadric surface in the first view.

The outline conic can be geometrically visualized as an image of the silhouette or the points on the surface where the view rays are locally tangent to the surface, e.g., the elliptical silhouette of a sphere viewed from outside the sphere.

The value A is the homography via the polar plane between the first view and the second view. Note that this equation contains, apart from the overall scale, only one ambiguous degree of freedom resulting from relative scaling of E and e . This ambiguity can be removed by introducing an additional normalization constraint, such as, $E(3,3) = 1$.

Furthermore, the sign in front of the square root is fixed within the outline conic in the image. The sign is easily determined by testing the equation above with the coordinates for one pair of corresponding pixels. Note that the parameters of the quadric transfer function can be directly computed from nine or more pixel correspondences in a projective coordinate system. So it is tempting to follow an approach similar to estimating a planar homography for planar displays, without computing any Euclidean parameters. However, as described below, in practice it is difficult to estimate the epipolar relationship in many cases. Hence, we use a pseudo-Euclidean approach, as described below.

Pre-Processing

All registration information is pre-calculated relative to a pair of stereo images. We assume that the stereo camera views the entire 3D display surface. One of the camera images is arbitrarily selected as having an origin c_0 . The stereo images are used to determine only the 3D points on the display surface. Hence, any suitable 3D acquisition system can be used.

As shown in Figure 1, the basic steps of our pre-processing method 100 are as follows. The details of these steps are described below. For each projector i , a predetermined image 101, e.g., a structured pattern in the form of a checkerboard, is projected 110 onto the quadric surface.

In step 120, features in images acquired of the predetermined image 101 by the stereo camera 102 are detected, and 3D points are reconstructed, i.e., correspondences 103, on the quadric surface, which correspond with the features of

the predetermined. Then, the quadric surface, Q 104 is fitted 130 to the detected correspondences.

For each projector i , a pose 105 of the projector, with respect to the camera, is determined 140 using the correspondence between projector pixels and 3D coordinates of points on the surface illuminated by the pixels. Determine 150 the quadric transfer function, Ψ_i and its inverse Ψ_i^{-1} , between the camera c_0 and projector i . Then, determine 160 intensity blending weights, Φ_i , in regions where the projected images overlap. At this point, a projector image can be projected 200 by warping, blending and projecting to appear as an undistorted output image 171 on the quadric surface.

Run-Time Processing

Figure 2 shows the basic steps of our rendering method 200. First, a projector image is generated 210. The projector image is represented as a texture mapped set of triangles. The image can be acquired by a still camera or a video camera, or the image can be generated by computer graphic rendering a 3D scene from a virtual viewpoint. The projector image is warped 220 into a frame of the projector according to the quadric transfer function Ψ_{0i} . Then, pixel intensities are attenuated 230 with blending weights Φ_i before projecting 240.

However, these steps involve several issues that need to be resolved. The quadric transfer function estimation, although a linear operation, requires non-linear optimization to reduce pixel re-projection errors. In addition, it is difficult to estimate the pose of the projector, i.e., the external parameters, because the 3D points projected on the quadric surface are usually nearly planar leading to a

degenerate condition. These and other issues, and a practical solution are described below.

Calibration

We determine parameters of the quadric transfer function, $\Psi_{oi} = \{A_i, E_i, e_i\}$, so that the projected output images are geometrically registered on the curved surface. The prior art methods, known for cameras, determine the quadric transfer parameters directly from pixel correspondences. That involves estimating the 4×4 quadric matrix, Q , in 3D using a triangulation of corresponding pixels and a *linear* method. If the internal parameters of the two views are not known, all the calculations are done in projective space after computing the epipolar geometry, i.e., the epipoles and the fundamental matrix.

However, if projectors rather than cameras are involved, the linear method produces very large re-projection errors in estimating the 3D quadric surface, Q . The errors are of the order of about 30 pixels for a conventional XGA projector.

There are several reasons for this. It is relatively straightforward to calibrate a camera directly from acquired images. However, projector calibration can only be done indirectly by analyzing images of projected images acquired by a camera. This can introduce errors. In addition, in a projector, the internal optics are usually offset from the principle point to project images upwards. Thus, the projector internal parameters are hard to estimate, and they are different than those of a camera. Furthermore, the fundamental matrix is inherently noisy given that the 3D points on the quadric surface illuminated by a single projector do not have

significant depth variation in display settings such as segments of spherical or cylindrical surfaces.

Therefore, the invention uses a pseudo-Euclidean approach where the internal and external parameters of the camera and the projectors are known *approximately*. These parameters are used to estimate Euclidean rigid transformations. Hence, unlike the planar case, computation of an accurate image transfer function for curved screens involves three-dimensional quantities.

Quadric Surface

We use a rigid stereo camera pair, C_0 and C'_0 , as a base for computing all geometric relationships. We arbitrarily select one of the cameras to define the origin and coordinate system. We calibrate a baseline stereo pair of cameras with the checkerboard pattern 101. For our calibration, the cameras do not necessarily have to be located at the sweet-spot, which is an important difference with respect to some of the prior art non-parametric approaches.

The stereo pair of cameras 102 observes the structured patterns projected by each projector. Using triangulation, a set of N 3D points, $\{X_j\}$ on the display surface, that correspond to pattern features, are detected. The quadric surface, Q , passing through each X_j is computed by solving a set of linear equations, $X_j^T Q X_j = 0$, for each 3D point. This equation can be written in the form $X_i V = 0$, where X_i is a 1×10 matrix, which is a function of X_i only and V is a homogeneous vector containing the distinct independent unknown variables of the quadric surface Q . With $N \geq 9$, we construct a $N \times 10$ matrix X , and solve the linear matrix equation $XV = 0$.

Given points in general position, the elements of V , and hence Q , are the one dimensional null-space of the matrix X .

Projector View

In addition to the quadric surface, Q , we need to know the internal and external parameters of each projector with respect to the origin of the camera. We use the correspondence between the projector pixels and coordinates of the 3D points they illuminate to compute the pose and internal parameters.

However, finding the pose of a projector from known 3D points on the quadric surface is error-prone because the 3D points are usually quite close to a plane leading to an unstable solution, see Faugeras, *Three-dimensional computer vision: a geometric viewpoint*, MIT Press, 1993, and Forsyth et al., "*Computer Vision, A Modern Approach*," FUTURES LAB, 2002, ActiveMural, Argonne National Labs, 2002.

Dealing with near-planar points is a difficult problem. If points are distributed in depth, then we can easily use a linear method to estimate the internal as well as the external parameters of the projector. On the other hand, if the points are known to be planar, then we can estimate the external parameters when some of the internal parameters are known.

For dealing with near-planar surfaces, we use an iterative procedure. If we know the projector internal parameters, we can first find an initial guess for external parameters based on homography and then use an iterative procedure based on Lu et al., "*Fast and globally convergent pose estimation from video images*," IEEE

Transactions on Pattern Analysis and Machine Intelligence, Vol. 22:6, pp. 610–622, 2000. Powell's method can be used for non-linear refinement of reprojection error. However, estimating projector internals is equally difficult. If the projectors cannot be easily moved, as mentioned above, then calibrating the projectors usually requires large surfaces illuminated in two or more positions.

Our strategy is to use projector internal parameters that are *approximately* known. We find internal parameters of just one projector and use these internal parameters for all projectors. At a later time, the same and other projectors will probably have different zoom settings and other mechanical or optical deviations. In addition, the external parameters computed by iterative method of Lu et al. is only approximate.

Camera to Projector Transfer

Therefore, we use perspective projection parameters of the camera along with approximate projection matrix of the projector to find the camera to projector quadric transfer using linear methods. Then, we refine the solution using non-linear optimization.

The quadric transfer parameters, $\Psi_{0i} = \{A_i, E_i, e_i\}$ are easy to calculate from the quadric surface Q , the camera projection matrix $[I | 0]$, and the projector projection matrix $[P_i | e_i]$ by

$$A_i = P_i - e_i q^T \quad E_i = q q^T - Q_{33}.$$

As stated above, the prior art linear method for finding the parameters is too imprecise for our purpose. Misalignment on the display screen can be as much as 15-30 pixels per projector resulting in annoying visual artifacts. A seamless display

requires sub-pixel accuracy. Therefore, we apply a non-linear minimization to refine the results obtained via the linear method. This can be done with an objective or 'cost' function.

For the objective function, we take the total squared transfer error for all pixels

$$\varepsilon_i = \sum_j \left(\frac{x_i^j(1,2)}{x_i^j(3)} - \frac{\hat{x}_i^j(1,2)}{\hat{x}_i^j(3)} \right)^2,$$

where \hat{x}_i^j are the transferred pixels for each known projected pixel x_j , i.e., pattern feature points, expressed as:

$$\hat{x}_i^j = A_i x_i^j \pm \sqrt{x_i^{jT} E_i x_i^j} e_i.$$

Note that the sign found using the linear method, which is same for all the pixels, remains the same during the non-linear optimization, e.g., the well known Nelder-Mead Simplex.

Partial Euclidean Reconstruction

One could ignore the Euclidean approach altogether, and proceed directly to projective space and non-linear optimization. If we have accurate projector internal parameters, then the non-linear optimization stages could be avoided. However, as mentioned earlier, ignoring Euclidean viewing parameters and solving the quadric transfer function purely from pixel correspondences leads to poor re-projection errors. Furthermore, the estimated 3D quadric surface, Q , cannot be used as an estimate for further non-linear optimization because the solution did not converge.

Accurate internal projector parameters only reduces the re-projection errors but do not eliminate the errors. This is because, many kinds of errors are propagated in the three dimensional Euclidean calculations, including estimating 3D points on the display surface by triangulation, estimating the 3D quadric surface using linear methods and finding the pose of the projector. The non-linear optimization attempts to minimize the physical quantity we care about the most, i.e., pixel re-projection error in image transferred from the camera to the projector for a known corresponding set of pixels.

Because the correspondence between overlapping projector pixels is indirectly defined by this image transfer function, minimizing pixel re-projection errors ensures geometric registration between the displayed pixels of the projectors.

Rendering

The rendering involves a two-pass approach. For 2D image data, we extract the appropriate input image. For 3D scenes, we first render the 3D models according to the sweet-spot. In the second pass, the resultant image is then warped into the projector image space using the quadric image transfer function.

Virtual View

When 3D scenes are displayed on a curved screen, the images are perspectively correct only from specific points in space, Such as point is popularly known as the sweet-spot or the virtual viewpoint. For a concave hemispherical surface, the sweet-spot is on a line through the center of the sphere, perpendicular to the cut-plane that cuts the sphere in half. For a convex hemispherical surface, the sweet-

spot can be mirrored through the surface. For a cylindrical surface, the sweet spot is similarly located, unless projected image wraps around the entire surface, in which case the sweet-spot can be anywhere. The sweet-spot is independent of the camera's focal length, but dependent on the shape and size of the curved display, and the peripheral viewing angle. The projected imagery can appear in front of, or behind the display surface depending on the location of the sweet-spot and the focal length of the virtual camera or viewer. As the viewer moves away from the sweet-spot, the images look distorted. In addition, one needs to specify the view frustum i.e., the viewing direction or principal axis, and the extent or field of view.

For some displays, it is possible to automatically determine the sweet-spot. For example, for a concave spherical dome, the center of the dome can be considered a good sweet-spot. The sweet-spot can be determined directly from the equation of the quadric surface, Q , i.e., $Q(1, 1)q$. For a cylindrical screen, a point on the axis of the cylinder that is midway along the extent of the cylinder is a good choice. Sometimes, the sweet-spot is decided by practical considerations e.g., a spot that is approximately at eye-level is considered ideal because images are almost always aligned with the horizontal and vertical axes of the real world.

In our case, the virtual viewpoint can be interactively fixed or moved because we have an approximate Euclidean reconstruction of the display geometry. Because the 3D coordinates of the location are known in the camera coordinate system, it is relatively simple to locate the sweet-spot with respect to the camera.

Sweet-Spot from Surface Points

When it is difficult to determine the parameters of the virtual viewpoint for rendering, one technique finds the best-fit plane to the set of points found on the illuminated part of the display surface. We fit an oriented bounding box (OBB) to the set of 3D points $\{X_i\}$ on the display surface. A point at a suitable distance in front of the screen, along the vector passing through the center of this box and normal to the best-fit plane can be selected as the sweet-spot. Because all the 3D points on a quadric surface lie on the convex hull of those 3D points, the OBB can be determined as follows.

If Y is an $N \times 3$ matrix of $\{X_i - \bar{X}\}$ where $\bar{X} = (\bar{x}, \bar{y}, \bar{z})$ is the centroid of the N 3D points, then the eigenvector corresponding to the smallest eigenvalue of the 3×3 matrix $Y^T Y$ gives the normal to the best-fit plane i.e., the axis of minimum variance. On the other hand, the largest side of the OBB, i.e., the extent of the 3D points projected in the best-fit plane, gives the approximate 'diameter' of the screen. The distance of sweet-spot in front of the screen can be selected to be proportional to this diameter, depending on the application and the desired field of view.

For immersive applications, a large field of view is desired. Hence, the distance should be about half of the diameter. For group viewing, the distance can be comparable to the diameter.

We recalculate the quadric transfer function Ψ_i between the virtual view image space and each projector output image. The process is very similar to computing Ψ_i Ψ_{0i} . First, we find the projection of 3D points on the quadric surface into the virtual view image space. Then, the correspondence between the virtual view and the projector image and the internal and external parameters of the virtual view and projector are sufficient to update Ψ_i .

Display Area

The view frustum for the virtual view is defined using the sweet-spot and the extents of the OBB. The viewing -at vector is from the virtual viewpoint toward the center of the OBB. Because the union of overlapping images from multiple projectors can illuminate a large area, we can ‘crop’ the view frustum to an aesthetic shape such as a rectangle or a circle. For 3D applications, we render a set of black quadrilaterals to crop regions outside the desired display area. For example, for a rectangular view, the view port is made by four large quadrilaterals near the outer edge of the view port in the projector’s image. The black quadrilaterals along with the rest of the 3D models are rendered and warped as described below. For 2D applications, the area outside the input image to be displayed is considered black.

Image Transfer

Given a 3D vertex, M , in a scene to be rendered, we find its screen space coordinates, m , in the virtual view. Then, we find the transferred pixel coordinate, m_i , in the output image of projector i , using the quadric transfer function, $\Psi_{0i} = \{A_i, E_i, e_i\}$. The polygons in the scene are then rendered with vertices M replaced with vertices m_i . Thus, the rendering process at each projector is very similar. Each projector frame buffer automatically stores the appropriate part of the virtual view image, and there is no need to explicitly determine extents of the projector.

Therefore, to render, at each projector, for each vertex M ,
 determine pixel m via a virtual view projection (M), and

determine a warped pixel m_i via the quadric transfer function $\Psi_i(m)$, then for each triangle T with vertices $\{M_j\}$,
render triangle with 2D vertices $\{m_{ji}\}$.

There are two issues with this approach. First, only the vertices in the scene, but not the polygon interiors, are accurately warped. Second, visibility sorting of polygons needs special treatment. After the quadric transfer, the edges between vertices of the polygon should theoretically map to second-degree curves in the projector's image.

However, scan conversion converts the curves to straight-line segments between the warped vertex locations. This problem is not discernible for a single projector. However, in the case of overlapping projectors, this causes individual different deviations from the original curve, and hence, the edge appears to be mis-registered on the display screen. Therefore, it is necessary to use sufficiently fine tessellation of triangles.

Commercial systems are already available that tessellate and pre-distort the input models on the fly so that they appear straight in a perspective correct rendering on the curved screen, see U.S. Patent No. 6,104,405, "Systems, methods and computer program products for converting image data to non-planar image data," issued to Idaszak et al. on Feb 26, 1997, and U.S. Patent No., 5,319,744, "Polygon fragmentation method of distortion correction in computer image generating systems," issued to Kelly et al. on June 7, 1994, incorporated herein by reference. Our method is compatible with fine tessellation provided by such systems. Pre-distortion of the scene geometry in commercial systems is used to avoid the two-pass rendering, which involves texture-mapping result of the first pass. In our case,

instead of pre-distorting the geometry, we pre-distort the image space projection. As an advantage, our invention can be implemented in part with a vertex shader of a programmable graphics unit (GPU).

Scan Conversion

When pixel locations in the projection of a triangle are warped, information needs to be passed along so that the depth buffer generates appropriate visibility information. In addition, for perspective correct color and texture coordinate interpolation, appropriate weight values ‘ w ’ need to be passed. Therefore, we post-multiply the pixel coordinates with ‘ w ’ according to

$$\begin{aligned} m(x, y, z, w) &= \text{VirtualViewProjection}(M(X)) \\ m_i(x'_i, y'_i, w'_i) &= \Psi'_i(m(x/w, y/w), 1) \\ m_i(x_i, y_i, z_i, w_i) &= [wx'_i/w'_i, wy'_i/w'_i, z, w]. \end{aligned}$$

Thus, vertex m_i has appropriate final pixel coordinate $(x'_i/w'_i, y'_i/w'_i)$ due to the quadric transfer function along with original depth and w values.

Figure 5 shows the code 500 for the vertex shader. For rendering 2D images, we densely tessellate the virtual view image space into triangles, and map the image as a texture on these triangles. Vertex, m , of each triangle is warped using the quadric transfer function into vertex (and pixel) m_i as above. Scan conversion automatically transfers colors and texture attributes at vertex m to vertex m_i , and interpolates in between. It is possible to render 3D scenes in a similar manner.

Note that our *warping* of a projector image using a quadric transfer function is different than *rendering* a quadric curve on a flat surface, see Watson et al., “*A fast*

algorithm for rendering quadratic curves on raster displays,” Proc. 27th Annual SE ACM Conference, 1989.

Intensity Blending

Pixels intensities in the areas of overlapping images are attenuated using alpha blending of the graphics hardware. Using the parametric equations of the quadric transfer function, the alpha maps are determined as follows.

For every projector pixel, x_i in projector i , we find the corresponding pixels in projector j using the equation

$$x_i \cong \Psi_{0j}(\Psi_{0i})^{-1}(x_i).$$

For cross-fading, pixels at the boundary of the projector’s image are attenuated.

Hence, the weights are proportional to the shortest distance from the frame boundary. The weight assigned to pixel x_i , expressed in normalized window pixel coordinates (u_i, v_i) , which are in the range $[0, 1]$, is

$$\Phi_i(x_i) \cong d(x_i)/\sum_j d(x_j),$$

where, $d(x)$ is $\min(u, v, 1-u, 1-v)$ if $0 \leq u, v \leq 1$, else $d(x) = 0$. Because we use a parametric approach, we are able to compute corresponding projector pixels and the weights at those locations at sub-pixel registration accuracy. The sum of weights at corresponding projector pixels accurately adds up to 1.0.

At each projector, the corresponding alpha map is stored as a texture map and rendered as screen aligned quadrilaterals during the last stage of the rendering.

Our method can also be used to project onto a convex dome. This projecting is particularly useful when the dome was made of translucent or transparent material. When the projection is from the rear, the viewer can experience a fully immersive experience without blocking any of the projectors.

Effect of the Invention

The invention enables the construction of projector systems with curved display surfaces for 2D or 3D visualization. Our system does not require an expensive infrastructure, and can be operated with casual alignment between multiple projectors and the display surface. Our automatic registration exploits a quadric image transfer function, and eliminates tedious setup and maintenance of projectors, and hence reduces cost. The invention can simplify the construction, calibration and rendering process for widely used applications such as used in flight simulators, planetariums and high-end visualization theaters. New applications enabled include low-cost, flexible dome displays, shopping arcades and projection on cylindrical columns or pillars.

The invention provides an elegant solution to a problem that has so far been solved by discrete sampling. An advantage is that, unlike prior art systems, our projectors do not need to be placed at the sweet-spot. This is important in real-world applications where the sweet-spot is usually exactly where viewers would like to be.

Although the invention has been described by way of examples of preferred embodiments, it is to be understood that various other adaptations and modifications may be made within the spirit and scope of the invention. Therefore, it is the object of the appended claims to cover all such variations and modifications as come within the true spirit and scope of the invention.

Brief Description of the Drawings

Figure 1 is a flow diagram of pre-processing steps used by a method for projecting images onto a curved surface according to the invention;

Figure 2 is a flow diagram of rendering steps used by a method for projecting images onto a curved surface according to the invention;

Figures 3 and 4 are diagrams a multi-projector system according to the invention;

Figure 5 is vertex shader code for a quadric transfer function according to the invention;

Figure 6 is a diagram illustrating a homography according to the invention; and

Figure 7 is a block diagram of multiple overlapping images.

1. A method for projecting an image onto a display surface, comprising:
 - projecting a predetermined image onto a display surface;
 - acquiring a stereo pair of images of the predetermined image;
 - determining a quadric transfer function between the predetermined image and the stereo pair of images via the display surface;
 - warping an output image according to the quadric transfer function; and
 - projecting the warped output image onto the display surface.
2. The method of claim 1 wherein the display surface is quadric.
3. The method of claim 2 wherein the display surface is selected from a group consisting of a sphere, hemisphere, spheroid, dome, cylinder, cone, paraboloid, hyperboloid, hyperbolic paraboloid, and ellipsoid.
4. The method of claim 2 wherein the quadric surface is concave.
5. The method of claim 2 wherein the quadric surface is convex.
6. The method of claim 1 further comprising:
 - projecting a plurality of predetermined images onto the display surface from a plurality of viewpoints;
 - acquiring a stereo pair of input images of the plurality of predetermined images;
 - determining a quadric transfer function between each predetermined image and the stereo pair of input images via the display surface;

warping a plurality of output images according to the corresponding quadric transfer function; and

projecting the plurality of warped output images onto the display surface from the plurality of viewpoints.

7. The method of claim 6 wherein the surface is quadric.

8. The method of claim 1 wherein the plurality of warped output images overlap and appear as a single seamless image.

9. The method of claim 1 further comprising:

determining three-dimensional correspondences between features of the predetermined image and corresponding features in the pair of stereo images;

fitting a quadric surface to the three-dimensional correspondences;

determining a pose of the predetermined image with respect to the stereo pair of input images; and

determining the quadric transfer function from the pose and the correspondences.

10. The method of claim 6 further comprising:

attenuating the plurality of warped output images according to blending weights before projecting.

11. The method of claim 1 wherein the predetermined image is a checkerboard pattern.

12. The method of claim 1 further comprising:

applying non-linear minimization to an objective function to obtain the quadric transfer function.

13. The method of claim 12 wherein the objective function takes a total squared transfer error for all pixels in predetermined image.

14. The method of claim 1 wherein the output image is rendered for an arbitrary virtual viewpoint.

15. The method of claim 10 further comprising:

cropping the plurality of attenuated and warped output images to an arbitrary view frustum.

16. The method of claim 1 wherein the quadric transfer function is determined parametrically.

17. The method of claim 1 wherein the warping is implemented with a vertex shader of a graphic processing unit.

18. A system for projecting an image onto a display surface, comprising:

a projector configured to project a predetermined image onto a display surface;

a camera configured to acquire a stereo pair of images of the predetermined image;

means for determining a quadric transfer function between the predetermined image and the stereo pair of images via the display surface;

means for warping an output image according to the quadric transfer function; and

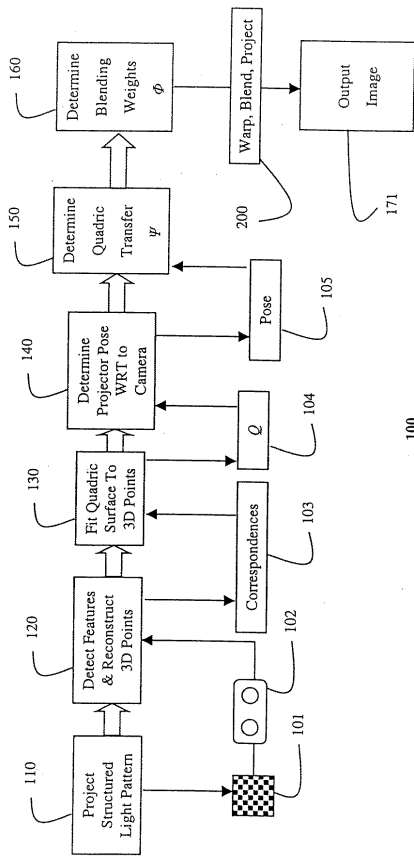
means for projecting the warped output image onto the display surface.

ABSTRACT

A method projects one or more image onto a curved display surface. First, a predetermined structured light pattern is projected onto the display surface. A stereo pair of images is acquired of the projected images on the display surface. Then, a quadric transfer function between the predetermined images and the stereo pair of images, via the display surface, is determined. Thus, an arbitrary output image can be warped according to the quadric transfer function so that when it is projected onto the display surface it appears correct.

Representative Drawing

Fig. 1



100

Fig. 1

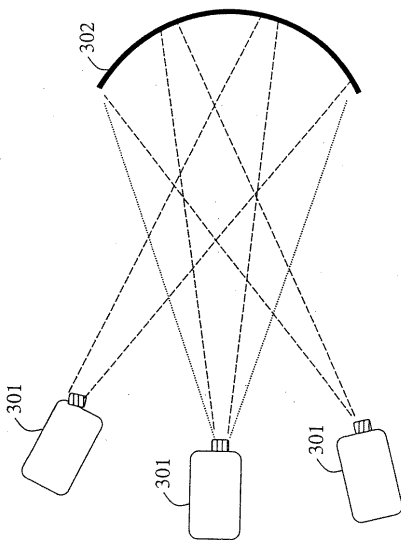


Fig. 3

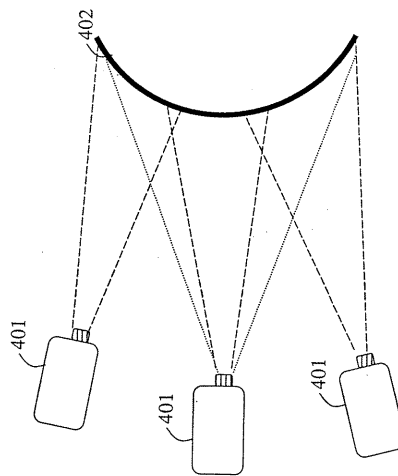
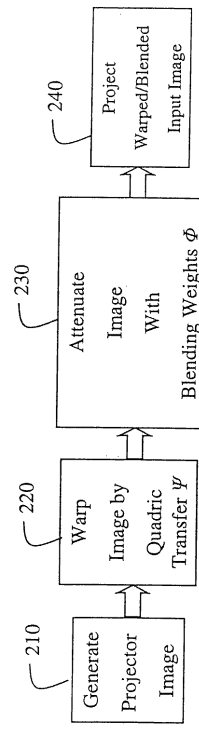


Fig. 4



200

Fig. 2

```

Vertex Shader Code for Quadric Transfer Function
vertout main( appin IN, uniform float4x4 modelViewProj, uniform float4
constColor, uniform float3x3 A, uniform float3x3 E, uniform float3 e ) {
  vertout OUT;
  float4 m1 = float4(IN.position.x, IN.position.y, IN.position.z, 1.0f);
  float4 m, mi; float3 m2, mp; float scale;
  m = mul( modelViewProj, m1);
  m2.x = m.x/m.w; m2.y = m.y/m.w; m2.z = 1;
  scale = mul(m2, mul(E, m2));
  mp = mul(A, m2) + sqrt(scale)*e;
  mi.x = m.w * (mp.x)/(mp.z);
  mi.y = m.w * (mp.y)/(mp.z);
  mi.zw = m.zw;
  OUT.position = mi;
  OUT.color0 = IN.color0; // Use original per-vertex specified color
  return OUT;
}

```

500

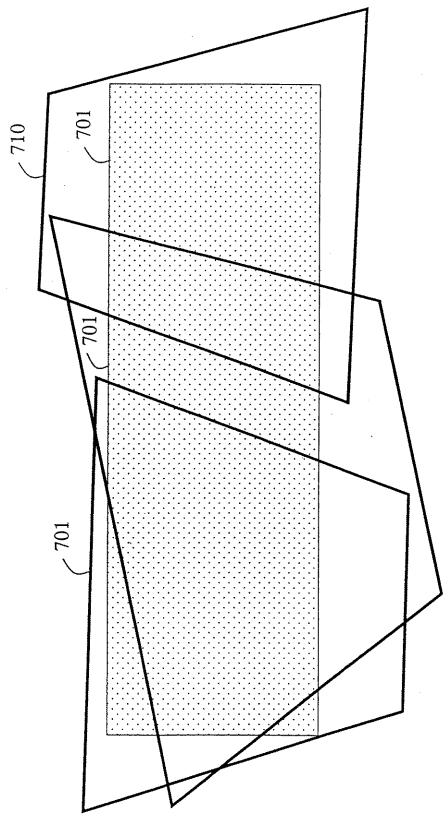


Fig. 7

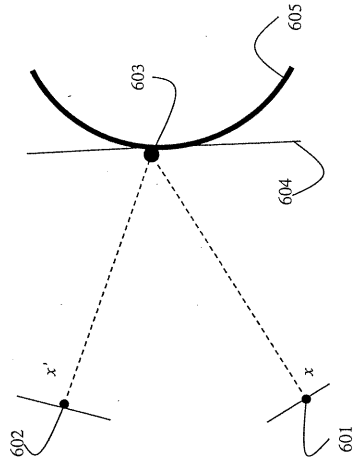


Fig. 6