

マーカーレス AR のための複数平面を利用した カメラ姿勢推定手法の検討

巻渕 有哉^{1,a)} 加藤 晴久^{1,b)} 柳原 広昌^{1,c)}

概要：平面を対象としたマーカーレス AR を実現する上で、複数枚の床面の撮影画像からカメラ姿勢を推定する方式が提案されている。しかしながら、安定的なカメラ姿勢を得るのに多くの入力画像を要するという課題があった。本稿では、少ない入力画像でマーカーレス AR を実現できるカメラ姿勢推定手法を提案する。具体的には、床面と直交した壁面が同時に存在する撮影画像が屋内環境において得られることに着目し、平面間の角度が既知という強い制約を加えることで最適化計算が高速に収束することを利用する。実験により、提案手法は従来手法と同等の精度で 21% 高速に収束することを確認した。

1. まえがき

拡張現実感 (Augmented Reality : AR) とは、現実空間に対して CG キャラクター等の仮想情報をあたかもその場所に存在しているかのように重ね合わせて表示する技術である。具体的には、撮影画像から現実空間に対するカメラ姿勢を推定するものであり、精度が高いほど違和感のない合成画像を得ることができる。

AR アプリケーションは主に 2 種類に分類され、平面上の特定の幾何模様 (配置が既知の点や直線、平行線や直交線、円等) を利用するマーカー型 [2-4] と、連続的な撮影により得られる複数枚の平面画像を利用するマーカーレス型 [5] が存在する。マーカー型の例としては、現実空間の平面上に設置された専用のマーカーを利用する ARToolKit [1] が一般的である。なお、文献によっては白黒の二値画像を利用するものをマーカー型、風景写真等の自然画像を利用するものをマーカーレス型と表現しているが、本稿ではいずれもマーカー型として扱う。一枚の撮影画像から比較的単純な計算処理により AR 表示が可能である一方で、対象の平面がマーカーを有するものに限定されるという制約がある。

マーカーレス型は、任意の平面環境に対して適用可能であるという特長がある。AR 技術を用いたマップナビゲーションの例では、行き先を示す矢印アイコンを地面に沿っ

て重畳表示することが求められる。しかしながら、その都度地面にマーカーを設置することや、対象の空間を事前にモデリングすることは現実的に困難である。一時的な情報を非特定の空間に AR 表示する用途においては、マーカーレス型が不可欠である。但し、AR 表示が可能となるまでに多くの撮影画像を必要とするという課題がある。

本稿では、少ない入力画像でマーカーレス AR を実現できるカメラ姿勢推定手法を提案する。床面と直交した壁面が同時に存在する撮影画像が屋内環境において得られることに着目し、平面間の角度が既知であるという強い制約により最適化計算が高速に収束することを利用する。

以下、2 章で関連研究について言及した後、3 章で複数枚の撮影画像を利用した平面画像の平行化手法 [5] の詳細について説明する。4 章で平面間の角度制約を利用した平行化手法を提案し、5 章で評価実験の結果と考察について述べる。最後に 6 章をむすびとする。

2. 関連研究

カメラ姿勢計算は、任意の角度で撮影された対象平面の画像から正面画像を生成する変換を求めることであり、「平面画像の平行化 (Planar Metric Rectification)」と呼ばれる。マーカーレス型では、正面画像に関する事前情報なしに正面画像への変換を求める必要があり、マーカー型よりも困難な課題となっている。

平面画像の平行化は、一般に 4 自由度の虚円点 (Image of Circular Points) を求めることで達成できる。マーカー型では、撮影画像内に存在する最少 5 組の直交線 [4] や円と消失線 (Vanishing Line) の交点 [2] から計算できるほか、複数の円の交点 [3] を利用することもできる。カメラの内

¹ (株) KDDI 研究所
KDDI R&D Laboratories Inc., 2-1-15 Ohara, Fujimino-shi,
Saitama, 356-8502 Japan

a) na-makibuchi@kddilabs.jp

b) hkato@kddilabs.jp

c) yanap@kddilabs.jp

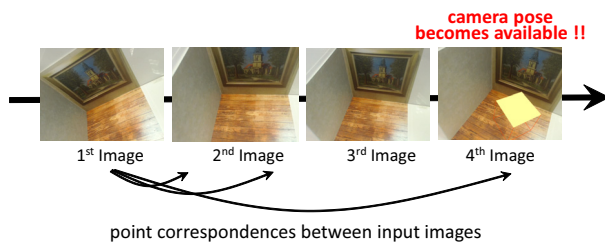


図 1 マーカーレス AR：入力画像間の点対応からカメラ姿勢を計算する。4 枚目の入力画像においてカメラ姿勢が収束し、立方体形状の CG を床面に重畳表示している様子。従来手法では、実際にはより多くの入力画像（10 枚程度）を必要とする。

Fig. 1 Marker-less AR.

部パラメータが既知の場合は、2 自由度の消失線を特定するだけでよく、最少 2 組の平行線から計算できる。一方でマーカーレス型では、対象平面の撮影画像間の点対応から消失線に代わる 2 自由度の平行化パラメータを推定する [5]。対象平面に対してカメラを動かした際に連続的に入力される画像に適用することで、事前情報を持たない任意の平面に対して AR 表示が可能である (図 1) [6]。但し、入力画像が少ない場合に最適化計算が収束しないという課題があり、結果としてカメラ姿勢を得るのに多くの入力画像を必要とする。図 1 には 4 枚目でカメラ姿勢が得られた例を示しているが、実際にはより多くの入力画像（10 枚程度）を必要とする。

提案手法は収束の高速化を目的として、複数平面環境において平面間の角度が既知であるという事前情報を利用する。屋内環境でのナビゲーションを想定した場合、人工建造物においては壁面と床面が直交の関係にあると仮定できる。また、ユーザーはカメラを進路の前方にわざとため、床面と壁面といった複数の平面が同時に存在する撮影画像が得られる。しかしながら、図 1 のように床面に対して AR 表示を行う場合、従来手法では床面の点対応のみが利用され、壁面の点対応と平面間の直交性が活用されていない。本稿では、平面間の角度に基づく最適化計算を導入し、強い制約のもとで高速に収束する平行化手法を提案する。

3. 従来手法：複数枚の撮影画像を利用した平面画像の平行化

図 2 をもとに平面画像の平行化について述べる。対象平面に対して任意の角度から撮影された N 枚のカメラ画像とそのピクセル座標系をそれぞれ I_i , $m_i = (u, v)^T (i = 1, 2, 3, \dots, N)$ により表す。図 1 のように複数の平面が撮影されている場合は、 m_i は平行化の対象平面（例えば底面）の領域のみを指す。また、カメラの内部パラメータ K を用いて $\bar{x}_i \sim K^{-1}m_i$ により正規化されたピクセル座標系を $x_i = (p, q)^T$ により表す。 $\bar{m}_i$ と $\bar{x}_i$ はそれぞれの同次座標

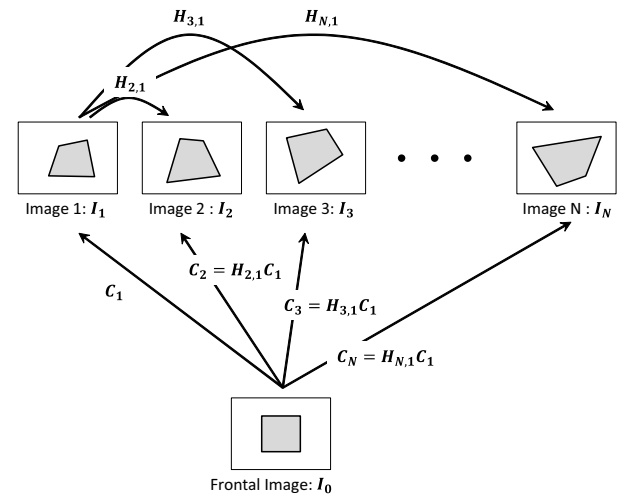


図 2 複数枚の撮影画像を利用する平行化手法の問題設定：入力画像間のホモグラフィ行列 $H_{j,1}$ から平行化行列 C_1 を求める。

Fig. 2 Problem setting of planar metric rectification from multiple views.

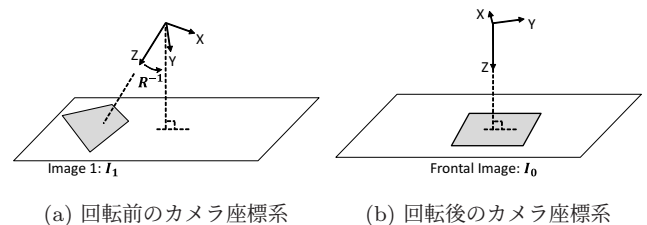


図 3 カメラ座標系の回転による平行化：カメラ座標系の光軸 (Z 軸) を平面の法線に重ね合わせる回転を加える。

Fig. 3 Planar rectification by rotating camera coordinate system

表現である。平面画像 $I_j (j = 2, 3, \dots, N)$ が入力される毎に、最初に入力された平面画像 I_1 について点対応を求め、ロバスト推定によりホモグラフィ行列 $H_{j,1}$ を得る。

対象平面の正面画像とその正規化ピクセル座標系を I_0 と x_0 により表し、平行化行列を $\bar{x}_i \sim C_i x_0$ の関係を満たす C_i により定義する。平行化行列はその定義より、逆行列 C_i^{-1} により各平面画像 I_i を正面画像 I_0 に変換するものである。平行化の目的は、 $H_{j,1}$ から各平面画像に対応する N 個の平行化行列 C_i を求めることである。但し、 $C_j = H_{j,1}C_1$ であるから、 $H_{j,1}$ から C_1 を求める問題に帰着する。

3.1 回転行列による平行化

I_1 のカメラ座標系に回転変化のみ (Pure Rotation) を加えることで I_1 を正面画像に変換する。図 3 に示すように、カメラ座標系の光軸 (Z 軸) を平面の法線に重ね合わせる回転となり、回転行列を便宜上、 R^{-1} と表記する。このとき、回転前のカメラ画像 I_1 と回転後に得られる正面画像 I_0 の関係式は、 $\bar{x}_1 \sim R\bar{x}_0$ となる [7]。 R は平行化行列の定義を満たすため、 $C_1 = R$ として求める。

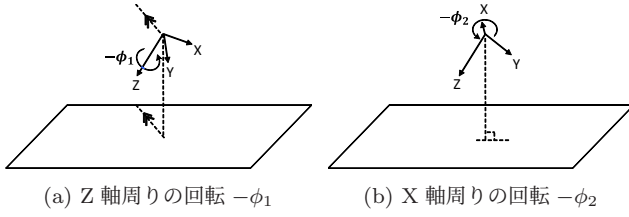


図 4 回転行列を構成する 2 軸の回転：図 3(a) に Z 軸周りと X 軸周りの回転を段階的に加えることで図 3(b) の状態にする。

Fig. 4 Two rotations constructing the rectifying rotation.

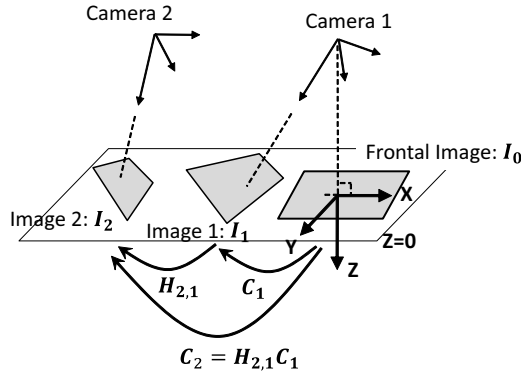


図 5 世界座標系における対象平面とカメラの位置関係： C_1 と C_2 は対応するカメラの位置・姿勢に依存するため、姿勢ホモグラフィ行列と表現する。

Fig. 5 Relationship between the rectified plane and the cameras in the world coordinate system.

回転行列 R は、3 次元座標系における 2 軸の回転行列により構成され、各軸の回転量が平行化パラメータとなる。これは図 4 に従って次のように解釈できる。まず、X 軸が平面の平行線に重なるように Z 軸周りに $-\phi_1$ ($-\pi/2 \leq \phi_1 < \pi/2$) 回転させる。続いて、Z 軸が平面の法線に重なるように X 軸周りに $-\phi_2$ ($-\pi/2 < \phi_2 < \pi/2$) 回転させる。つまり、 C_1^{-1} と C_1 はそれぞれ式 (1) と式 (2) で表される。

$$C_1^{-1}(\phi_1, \phi_2) = R_X(-\phi_2)R_Z(-\phi_1) \quad (1)$$

$$C_1(\phi_1, \phi_2) = R_Z(\phi_1)R_X(\phi_2) \quad (2)$$

3.2 姿勢ホモグラフィ行列の性質を利用した平行化推定

図 5 をもとに平行化パラメータの推定について述べる。なお、本章の説明は [6] を参考としている。対象平面を世界座標系 (X, Y, Z) における $Z = 0$ により表す。このとき、世界座標系におけるカメラ 1 の位置及び姿勢パラメータをそれぞれ t , R とすると、 x_1 と (X, Y, Z) との間には式 (3) が成り立つ。

$$\bar{x}_1 \sim \begin{pmatrix} R & t \\ & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ 0 \\ 1 \end{pmatrix} \sim \begin{pmatrix} r_1 & r_2 & t \\ & & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ 1 \end{pmatrix} \quad (3)$$

なお、 r_1 と r_2 はそれぞれ R の 1 列目と 2 列目のベクトルである。式 (3) の右辺の $(X, Y, 1)^T$ は、 x_0 の同次座標表現であるから、 $C_1 = (r_1, r_2, t)$ が得られる。 C_1 は世界座標系におけるカメラの位置・姿勢に起因するものであるため、本稿ではこれを姿勢ホモグラフィ行列と表現する。回転行列の性質より、姿勢ホモグラフィ行列 C_1 は式 (4) を満たす。

$$C_1^T \cdot C_1 = \begin{pmatrix} r_1^T \\ r_2^T \\ t^T \end{pmatrix} \begin{pmatrix} r_1 & r_2 & t \end{pmatrix} = \begin{pmatrix} 1 & 0 & \cdot \\ 0 & 1 & \cdot \\ \cdot & \cdot & \cdot \end{pmatrix} \quad (4)$$

なお、式 (4) 中の (1, 3) 成分、(2, 3) 成分、(3, 1) 成分、(3, 2) 成分、及び (3, 3) 成分は不定値であり、ドット記号で表現している。他の姿勢ホモグラフィ行列 C_j ($j = 2, 3, \dots, N$) についても同様にこの性質を満たす。 $C_j = H_{j,1}C_1$ より式 (5) が導かれる。

$$C_j^T \cdot C_j = (C_1^T H_{j,1}^T) \cdot H_{j,1}C_1 = \begin{pmatrix} 1 & 0 & \cdot \\ 0 & 1 & \cdot \\ \cdot & \cdot & \cdot \end{pmatrix} \quad (5)$$

ここで、 C_1 が式 (5) を満たすように非対角成分が 0、対角成分が同値となることを制約としたコスト値を式 (7) で定義する。

$$(C_1^T H_{j,1}^T) \cdot H_{j,1}C_1 = \begin{pmatrix} a_{1,1} & a_{1,2} & \cdot \\ a_{1,2} & a_{2,2} & \cdot \\ \cdot & \cdot & \cdot \end{pmatrix} \quad (6)$$

$$e_j(C_1) = (a_{1,2}/a_{1,1})^2 + (a_{2,2}/a_{1,1} - 1)^2 \quad (7)$$

コスト値 $e_j(C_1)$ は、1 つの画像間ホモグラフィ行列 $H_{j,1}$ をもとに計算され、 C_1 の姿勢ホモグラフィ行列に対する代数学的な距離を示す。そこで、入力された全ての画像間ホモグラフィ行列について、計算されるコスト値の総和を最小にするような C_1 を探索することで平行化行列を推定する。式 (2) より、具体的には式 (8) を最小化する平行化パラメータ (ϕ_1, ϕ_2) を計算する。

$$e(\phi_1, \phi_2) = \sum_{j=2}^N e_j(C_1(\phi_1, \phi_2)) \quad (8)$$

平行化パラメータ (ϕ_1, ϕ_2) の解の範囲はそれぞれ、 $[-\pi/2, \pi/2]$, $[-\pi/2, \pi/2]$ である。このパラメータ空間において、次のようにして大域的な最適解を探索する。まず、一定間隔（本研究では 20° ）毎に式 (8) に従ってコスト値をサンプリングする。その結果、各要素にコスト値を持つ行列が得られる。続いて、各要素の値を上下左右の 8 つの隣接要素と比較することで局所最小解を見つける。局所最小解の各パラメータについて、それを初期値として滑降シンプレックス法 [9] によりコスト値をリファインす

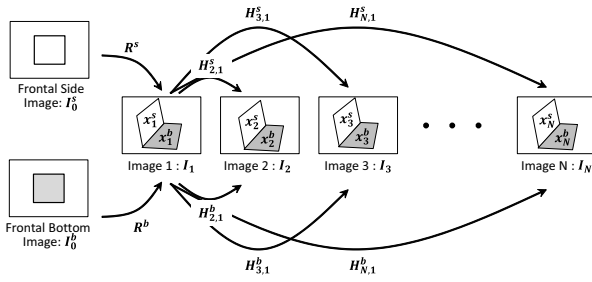


図 6 平面間の角度制約を利用した平行化の問題設定：底面と側面の平行化パラメータが既知の角度関係にあることを利用する。

Fig. 6 Problem setting of planar metric rectification using the angle constraint between planes.

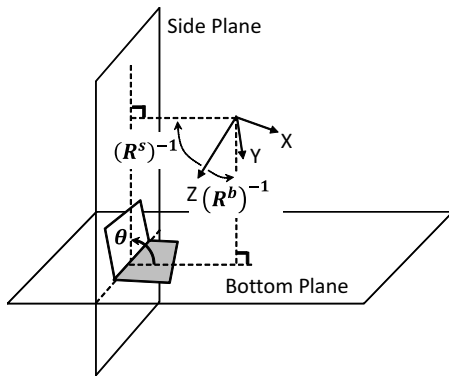


図 7 平面間の角度制約： R^s を R^b と θ により定式化する。

Fig. 7 Angle constraint between planes.

る。最後に、リファインされたコスト値同士を比較し、最も小さな値に対する二番目に小さな値の比率（収束判定パラメータ σ ）を計算する。閾値以下であれば解が収束したと判定し、最小値をとるパラメータを最適解として得る。

4. 提案手法：平面間の角度制約を利用した平行化推定

本稿では、平面間の角度が事前情報として与えられた場合に有効な平行化手法を提案する。以後、図 6 をもとに述べる。底面と側面の二平面が同時に撮影された入力画像のうち、底面と側面の各領域の正規化ピクセル座標系をそれぞれ、 $x_i^b, x_i^s (i = 1, 2, 3, \dots, N)$ により表す。また、底面と側面の各領域について計算されるホモグラフィ行列をそれぞれ、 $H_{j,1}^b, H_{j,1}^s (j = 2, 3, \dots, N)$ により表す。平面間の角度を θ として、底面と側面の平行化行列をそれぞれ R^b と R^s により表す（図 7）。従来手法により各平面の平行化パラメータを独立に推定する場合、未知パラメータの総数は $2 (\text{in } R^b) + 2 (\text{in } R^s) = 4$ となる。提案手法では、 R^s を R^b と θ により定式化することで未知パラメータの総数を 4 から 3 に減らし、より狭い解空間での最適化計算を実現する。

R^s を R^b と θ により定式化するために、底面に関して平行化を行った後のカメラ座標系（図 3(b)）に対して更なる回転変化を加える。具体的には、X 軸が側面の平行線に

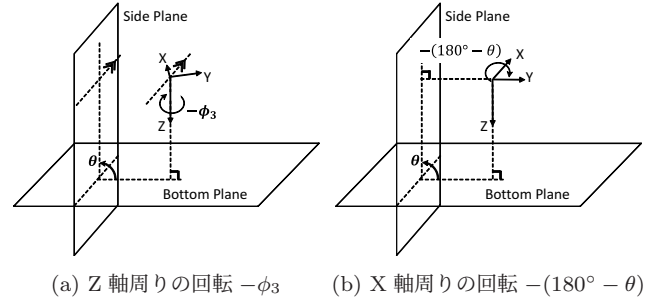


図 8 回転行列を構成する 2 軸の回転：図 3(b) に図 8(a) の回転を加えることにより、側面の平行化を達成するのに必要な残りの X 軸周りの回転量が $-(180^\circ - \theta)$ となる。

Fig. 8 Additional two rotations constructing the rectifying rotation.

重なるように Z 軸周りに $-\phi_3 (-\pi/2 \leq \phi_3 < \pi/2)$ 回転させる（図 8(a)）。結果として、側面の平行化を達成するのに必要な残りの X 軸周りの回転量が $-(180^\circ - \theta)$ となる（図 8(b)）。以上により、提案手法における底面の平行化行列 R^b と側面の平行化行列 R^s をそれぞれ式 (9) と式 (10) で表す。

$$R^b(\phi_1, \phi_2, \phi_3) = R_Z(\phi_1)R_X(\phi_2)R_Z(\phi_3) \quad (9)$$

$$R^s(\phi_1, \phi_2, \phi_3) = R_Z(\phi_1)R_X(\phi_2)R_Z(\phi_3)R_X(180^\circ - \theta) \quad (10)$$

提案手法における平行化パラメータは (ϕ_1, ϕ_2, ϕ_3) であり、底面と側面で定義されるコスト値の総和を最小化するパラメータを探索することにより解を得る。 $e_j^b(R^b)$ と $e_j^s(R^s)$ をそれぞれ、 $H_{j,1}^b$ と $H_{j,1}^s$ について計算されるコスト値として、最小化する関数は式 (11) で定義される。

$$e(\phi_1, \phi_2, \phi_3) = \alpha \sum_{j=2}^N e_j^b(R^b(\phi_1, \phi_2, \phi_3)) + (1 - \alpha) \sum_{j=2}^N e_j^s(R^s(\phi_1, \phi_2, \phi_3)) \quad (11)$$

なお、 α は重み係数であり、各平面のホモグラフィ行列の推定精度をもとに設定することもできるが、本研究では簡易的に $\alpha = 0.5$ とする。

平行化パラメータ (ϕ_1, ϕ_2, ϕ_3) の解の範囲はそれぞれ、 $[-\pi/2, \pi/2]$, $[-\pi/2, \pi/2]$, $[-\pi/2, \pi/2]$ である。このパラメータ空間において、サンプリング（ 20° 刻み）と滑降シンプレックス法によるリファインにより大域最適解を探索する。なお、提案手法ではパラメータが 3 つ存在するため、サンプリングは 3 次元の解空間に対して行い、局所最小解は上下左右の 26 個の隣接要素と比較することで判定する。最適解の収束条件は従来手法と同様に、最も小さなコスト値に対する二番目に小さなコスト値の比率（収束判定パラメータ σ ）が閾値以下であることとする。

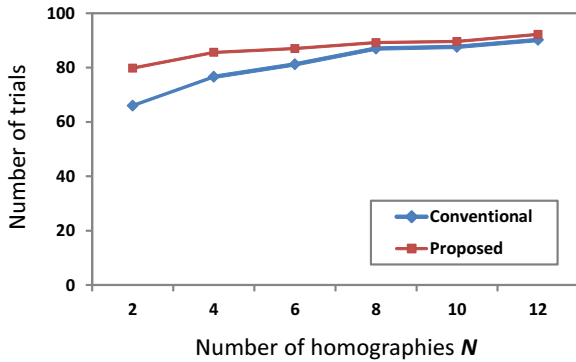


図 9 最適化計算が収束した試行数: $\sigma = 0.05$ で 500 試行のうち、最適化計算が収束した試行数 (100 試行あたりに換算したもの)。

Fig. 9 Number of trials where the optimization is converged.

5. 評価実験

図 1 に示す床面と壁面が直交の関係にある二平面環境 ($\theta = 90^\circ$) に対して、底面と側面が同時に存在する撮影画像を 150 枚取得する。撮影画像中の底面と側面の領域は事前に既知とする。ホモグラフィ行列の入力数に対する依存性を評価するために、次のようにしてデータセットから画像列を生成する。まず、一枚目の入力画像をランダムに一枚選択する。二枚目以降の入力画像については、一枚目の入力画像の各平面領域に対して一定数 (100) のインライアが得られるものをランダムに選択する。なお、点対応は SiftGPU [10] を用いて計算し、ホモグラフィ行列は RANSAC により推定する。以上により、床面と壁面のホモグラフィ行列をそれぞれ N 個を得る $N+1$ 枚の画像列を用意する。従来手法が床面の N 個のホモグラフィ行列を利用する一方で、提案手法は床面と壁面の計 $2N$ 個のホモグラフィ行列を利用する。

カメラ姿勢の推定精度は、画像列に含まれる入力画像中の各特徴点の再投影誤差により評価する。画像列に含まれる入力画像 I_i 中の各特徴点の座標と、対応する平面テクスチャ画像上の座標をそれぞれ、 $m_{i,k}$, $l_{i,k}$ により表す。これらは I_i とテクスチャ画像間の点対応から求める。また、 C_i により入力画像 I_i から正面画像 I_0^f 上へ逆投影される特徴点の座標 $m'_{i,k}$ は、 $\bar{m}'_{i,k} \sim C_i^{-1} \bar{m}_{i,k}$ により得られる。ここで、特徴点群 $l_{i,k}$ と $m'_{i,k}$ は、スケール、回転及び並進のパラメータが異なるため、再投影誤差を計算する前に点群間の位置合わせを行う必要がある。具体的には、 $\bar{l}_{i,k} \sim S \bar{m}'_{i,k}$ を満たす相似変換行列 S を線形最小二乗法により推定する [8]。以上より、再投影誤差を式 (13) で計算する。

$$\bar{l}_{i,k} \sim C_i S^{-1} \bar{l}_{i,k} \quad (12)$$

$$repro = \sum_{i=1}^{N+1} \sum_k |l'_{i,k} - m_{i,k}| \quad (13)$$

収束判定パラメータを $\sigma = 0.05$ とし、データセットか

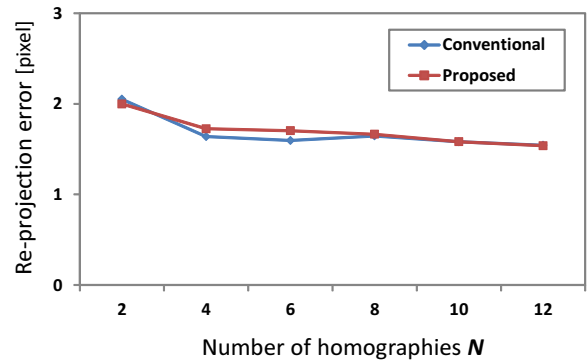


図 10 再投影誤差: $\sigma = 0.05$ で 500 試行の中央値。

Fig. 10 Re-projection error.

ら生成した 500 個の画像列に対してカメラ姿勢推定を行った。最適化計算が収束した試行数 (100 試行あたりに換算したもの) と再投影誤差の中央値を図 9 と図 10 に示す。なお試行数は、 σ により収束したと判定された試行の数を意味しており、滑降シンプレックス法における反復回数ではない。また、再投影誤差を中央値で評価した理由は、最適解から離れた局所解が初期値として選択された場合は正しい最適解が得られず、結果として再投影誤差の値が大きく跳ね上がるためである。

収束した試行数については、提案手法が従来手法を全体的に上回っており、増加幅は $N = 2$ のときに最大 21% 改善とした (図 9 の $N = 2$)。提案手法では、平面間の角度制約という強い制約によって最適解周辺でコスト関数の勾配が大きくなる。他の局所解との差が広がった結果、少ない入力のもとで収束した試行数が増加したものと考えられる。また、再投影誤差については、提案手法と従来手法の差は最大 0.1 ピクセルと小さく、CG の位置合わせ表示に与える影響が少ないことから、提案手法は従来手法と同等の精度が得られたと結論付けた (図 10)。いずれの手法も、最適解周辺でサンプリングされた初期値から滑降シンプレックス法により最適解に収束し、同等の精度が得られたものと考えられる。

ホモグラフィ行列の入力数 N に対する依存性については、収束した試行数はホモグラフィ行列の増加と共に単調に増加した (図 9 の $N = 2$ から $N = 12$ までの区間)。従来手法は、ホモグラフィ行列の推定誤差により生じた最適解以外の局所解との差が広がるのに多くの入力を要する。但し、ホモグラフィ行列の増加と共に勾配が急になり、収束判定されやすくなるため、従来手法と提案手法との差は $N = 2$ から $N = 12$ にかけて単調に減少する。一方で、再投影誤差については $N = 4$ 以降ほとんど減少していない (図 10 の $N = 2$ から $N = 12$ までの区間)。提案手法と従来手法で使用する最適化計算が、再投影誤差ではなく平行化行列の代数的な距離を最小化するものであるためと考えられる。

表 1 収束パラメータと再投影誤差の安定性との関係: $N = 2$ で 500 試行のうち, 再投影誤差が 20 以上となった試行数 (100 試行あたりに換算したもの). かつこ内は収束条件を満たす試行数.

Table 1 Relationship between the convergence parameter and the stability of re-projection errors.

	Conventional	Proposed
$\sigma = 0.05$	1.8 (66)	2 (79.8)
$\sigma = 0.5$	17.4 (92.4)	18.2 (96.4)

最適解の安定性と収束判定される試行数はトレードオフの関係にあり, 収束判定パラメータ値に大きく依存する. $\sigma = 0.05$ と $\sigma = 0.5$ という二種類の条件下で, 500 試行のうち再投影誤差が 20 以上となった試行数 (100 試行あたりに換算したもの) を表 1 に示す. また, かつこ内に収束判定された試行数を示す. いずれの手法も, σ の値を大きくするほど収束判定されやすくなった. 一方で, 最適解を判定するのに十分なコスト関数の勾配が得られず, 誤った局所解が選択されることで最適解が外れ値をとる試行が増加した.

最後に, 計算時間について述べる. 従来手法, 提案手法共に, 特徴点の抽出, ホモグラフィ行列の推定, サンプリング, リファインから構成される. 但し, 提案手法についてのみ, 底面に加えて側面のホモグラフィ行列を推定する必要があるため, ホモグラフィ行列の推定処理は二回行われる. $N = 2$, $\sigma = 0.5$ のときの計算時間の内訳を表 2 に示す. 実行環境は, Core i7-2600K 3.4 GHz, Geforce GTX 780 である. サンプリング, リファインに要する時間はいずれも提案手法の方が増加する結果となった. 提案手法では解空間が 2 次元から 3 次元に拡張されたことで, コスト値を計算するサンプル数が増加したと同時に, 各局所最小解のリファインに要する反復回数が増加したことが主な要因として挙げられる. また, コスト値の計算に従来手法の 2 倍のホモグラフィ行列を利用することも一因である. 提案手法の計算時間を改善するためには, サンプリング間隔を狭くすることで初期値が最適解に近くなり, 反復回数が減少することが期待できる. 但し, その分サンプリング時間が増加するため, 総合的な計算時間の改善については今後の検討課題とする.

6. まとめ

本稿では, 平面間の角度が既知の複数平面環境において少ない入力画像でマーカーレス AR を実現できるカメラ姿勢推定手法を提案した. 提案手法は, 最適化計算に平面間の角度制約を加えることで最適解が高速に収束することを実現する. 実験により, 提案手法は従来手法と同等の精度で 21% 高速に収束することを確認した.

表 2 計算時間 [ms]: $N = 2$, $\sigma = 0.05$

Table 2 Computational cost.

	Conventional	Proposed
Feature Extraction	26.7	
Homography Estimation 1	17.5	
Homography Estimation 2	-	11.6
Sampling	1.4	1.8
Refining	2.4	4.5
Total	48.0	62.0

参考文献

- [1] H. Kato and M. Billinghurst, "Marker tracking and hmd calibration for a video-based augmented reality conferencing system," in Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR '99), pp.85–94, 1999.
- [2] Y. Chen, et al., "Planar metric rectification by algebraically estimating the image of the absolute conic," in Proceedings of the 17th International Conference on Pattern Recognition (ICPR '04), vol.4, pp.88–91, 2004.
- [3] Y. Chen, P. Lu, W. Li and S. Wang, "Metric planar rectification from perspective view via circles," in Proceedings of Computational Intelligence for Image Processing (CIIP '09), pp.69–75, 2009.
- [4] D. Liebowitz and A. Zisserman, "Metric rectification for perspective images of planes," in Proceedings of Computer Vision and Pattern Recognition (CVPR '98), pp.482–488, 1998.
- [5] A. Ruiz, P.E. López-de-Teruel, and L. Fernández, "Practical Planar Metric Rectification," BMVC, 579–588, 2006.
- [6] C. Pirchheim and G. Reitmayr, "Homography-based planar mapping and tracking for mobile phones," in Proceedings of the 10th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR '11), pp.27–36, 2011.
- [7] R. Hartley and A. Zisserman, "Multiple view geometry in computer vision," vol.2, Cambridge Univ Press, 2000.
- [8] 玉木徹, "姿勢推定と回転行列," 信学技報, SIP2009-48, pp.59–64, 2009.
- [9] GSL: GNU Scientific Library, <http://www.gnu.org/software/gsl/>.
- [10] SiftGPU: A GPU Implementation of Scale Invariant Feature Transform (SIFT), <http://cs.unc.edu/~ccwu/siftgpu/>.