

高速な3次元再構成のための 最適化アプローチ

筑波大学大学院 システム情報工学研究科

正木 俊行, 久野 誉人

Toshiyuki Masaki and Takahito Kuno

Graduate School of Systems and Information Engineering,
University of Tsukuba

概要

3次元再構成問題を最適化手法を用いて解く F.Karl らの研究 [4, 5] をもとに, より高速な求解のための新たなモデルを提案する. モデル化の際に残差の定義を見直すことで問題の非線形性を回避し, 線形計画問題や最小二乗問題などの解きやすい問題へ帰着させることで, 大幅な高速化に成功した. また, 計算機実験により提案モデルの優位性を確かめるとともに, 両モデルにより得られた解の間に成り立つ関係を示す.

Key words: Computer vision, multiple view geometry, linear programming, second-order cone programming.

1 はじめに

3次元再構成 (3D-reconstruction) とは, 2次元の画像情報から3次元の座標情報を復元する問題のことである. この問題はコンピュータビジョンの分野において古くから扱われているスタンダードな問題である一方, 近年の高速な計算機による処理を前提とした大規模な問題の提案もなされており, 高速かつ高精度な解法が求められている.

本研究では, 幾つかの3次元再構成問題を最適化問題へ帰着させるための手法を提案する. 最適化問題の多くのクラスには専用の高速なソルバーが数多く存在しており, 最適化問題への帰着にはこれらの高速なソルバーを利用できるという利点がある. 関連する先行研究としては 2008

年の F.Karl らによるもの [4, 5] があり, 3次元再構成問題を最適化問題としてモデル化している. この最適化問題は非線形構造を持つために求解が難しく, 彼らは二次錐計画問題のソルバーと二分法を利用した反復解法を提案している. しかし, 二分法の収束までにかかる反復処理がネックとなり, 解が高速に求まらないという問題があった.

我々は問題構造の非線形性を避けるために新たなモデルを定義し, 3次元再構成問題を線形な最適化問題へと帰着させる. これにより問題構造を単純化し, 従来よりも大幅に高速な求解を目指す. さらに目的関数に用いるノルムを変更することで, 問題を二次錐計画問題や線形計画問題, 二次計画問題, 最小二乗問題へと帰着させ, それぞれの問題に帰着したときの求解速度や解の精度の違いを数値実験により比較する. また, 従来モデルと提案モデルそれぞれの解の間に成り立つ関係式を導出し, 特定の条件下で両モデルの解が一致することを示す.

2 ピンホールカメラモデル

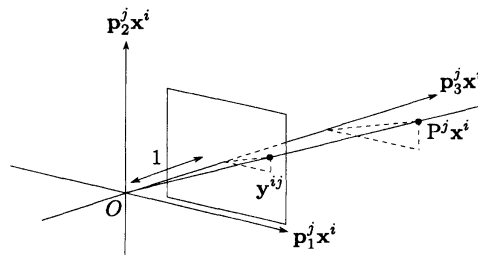


図 1: ピンホールカメラモデル

撮影される n 個のオブジェクトの座標を, 3次元の同次座標系におけるベクトル $\mathbf{x}^i = [x_1^i, x_2^i, x_3^i, 1]^\top \in \mathbb{R}^4$ ($i = 1, \dots, n$) で表す. m 個のカメラに対応するカメラ行列を

$$\mathbf{P}^j = \begin{bmatrix} \mathbf{p}_1^j \\ \mathbf{p}_2^j \\ \mathbf{p}_3^j \end{bmatrix} \in \mathbb{R}^{3 \times 4}, \quad j = 1, \dots, m$$

とし, カメラ $\mathbf{P}^j$ によるオブジェクト $\mathbf{x}^i$ のイメージを $\mathbf{y}^{ij} = [y_1^{ij}, y_2^{ij}, 1]^\top$ とする. ピンホールカメラモデルによる撮影の仕組みは図2のようになり, 三角形の相似の関係からすぐに

$$y_k^{ij} = \frac{\mathbf{p}_k^j \mathbf{x}^i}{\mathbf{p}_3^j \mathbf{x}^i}, \quad k = 1, 2 \quad (1)$$

が導かれる.

表 1: 3次元再構成問題の未知変数と既知変数

problem	# of cameras	# of objects	known	unknown	# of variables
triangulation	$m \geq 2$	1	$\mathbf{P}^j, \mathbf{y}^{1j}$	$\mathbf{x}^1$	3
camera-resectioning	1	$n \geq 6$	$\mathbf{x}^i, \mathbf{y}^{i1}$	$\mathbf{P}^1$	11
known-rotation-problem	m	n	$\mathbf{R}^j, \mathbf{y}^{ij}$	$\mathbf{x}^i, \mathbf{t}^j$	$3(n-1) + 3m - 1$

3 本研究であつかう問題

本研究であつかう3つの3次元再構成問題の未知変数と既知変数をまとめたものを表1に示す。ただし、回転行列 $\mathbf{R}^j \in \mathbb{R}^{3 \times 3}$ と並進ベクトル $\mathbf{t}^j \in \mathbb{R}^3$ はそれぞれカメラ行列 $\mathbf{P}^j$ の向きと座標に対応するパラメタで、

$$\mathbf{P}^j = [\mathbf{R}^j | \mathbf{t}^j]$$

と書けるものとする。

triangulation は、カメラの位置 $\mathbf{R}^j$ 、方向 $\mathbf{t}^j$ とオブジェクトのイメージ $\mathbf{y}^{1j}$ が与えられた状態で、オブジェクトの3次元座標 $\mathbf{x}^1$ を推定する問題である。camera-resectioning は、オブジェクトの座標 $\mathbf{x}^i$ とそのイメージ $\mathbf{y}^{i1}$ が与えられた状態で、カメラの位置 $\mathbf{R}^1$ 、方向 $\mathbf{t}^1$ を推定する。known-rotation-problem は、カメラの方向 $\mathbf{R}^j$ とイメージ $\mathbf{y}^{ij}$ が与えられた状態で、カメラの位置 $\mathbf{t}^j$ およびオブジェクトの座標 $\mathbf{x}^i$ を推定する。それぞれの変数の数に着目すると、triangulation と camera-resectioning は m, n にかかわらず変数の数が決まっているのに対し、known-rotation-problem は m, n の増加にともなって変数の数も増加することがわかる。

4 従来モデル

いま、 $\mathbf{x}^i, \mathbf{R}^j, \mathbf{t}^j$ のいずれかに未知数が含まれるとし、等式(1)より L_p ノルムを用いて残差 γ^{ij} を定義する。

$$\begin{aligned} \gamma^{ij} &= \|r_1^{ij}, r_2^{ij}\|_p, \\ r_k^{ij} &= y_k^{ij} - \frac{\mathbf{p}_k^j \mathbf{x}^i}{\mathbf{p}_3^j}, \quad k = 1, 2 \end{aligned} \quad (2)$$

この残差(2)を用いて、3節で述べた3次元再構成問題を次のような最

適化問題として定式化する.

$$\left\{ \begin{array}{ll} \min. & \|\gamma^{11}, \dots, \gamma^{nm}\|_q \\ \text{s.t.} & \gamma^{ij} = \|r_1^{ij}, r_2^{ij}\|_p \\ & r_k^{ij} = y_k^{ij} - \frac{\mathbf{p}_k^j \mathbf{x}^i}{\mathbf{p}_3^j \mathbf{x}^i}, \quad k = 1, 2 \end{array} \right\} \quad \begin{array}{l} i = 1, \dots, n \\ j = 1, \dots, m \end{array} \quad (3)$$

たとえばカメラの位置 $\mathbf{R}^j$, 方向 $\mathbf{t}^j$ およびオブジェクトのイメージ $\mathbf{y}^{1j}$ を定数として与え, 未知変数 $\mathbf{x}^1$ について問題 (3) を解くことで, triangulation 問題におけるオブジェクトの推定座標 $\mathbf{x}^1$ を得るのである.

先行研究では $p = 2, q = \infty$ のときの問題 (3) の解法を与えている. $p = 2, q = \infty$ のとき, 問題 (3) は次の等価な問題に書き換えられる.

$$\left\{ \begin{array}{ll} \min. & \gamma \\ \text{s.t.} & \|y_1^{ij} \mathbf{p}_3^j \mathbf{x}^i - \mathbf{p}_1^j \mathbf{x}^i, y_2^{ij} \mathbf{p}_3^j \mathbf{x}^i - \mathbf{p}_2^j \mathbf{x}^i\|_2 \leq \gamma \mathbf{p}_3^j \mathbf{x}^i, \end{array} \right\} \quad \begin{array}{l} i = 1, \dots, n \\ j = 1, \dots, m \end{array} \quad (4)$$

(ただし, 全てのオブジェクトが全てのカメラの前面に位置している, すなわち $\mathbf{p}_3^j \mathbf{x}^i > 0$ であると仮定する)

残差 γ が未知変数であることから, この問題の制約式の右辺は未知変数が $\mathbf{x}^i, \mathbf{R}^j, \mathbf{t}^j$ のいずれであっても非線形である. 制約条件が非線形の問題を解くことは困難であるため, 残差 γ を固定し, 問題 (4) の制約条件の実行可能性のみに着目する. Algorithm 1 に示すように, 制約条件の実行可能性を満たす最小の γ を二分法を用いて探索することで, 問題 (4) の解が得られる. このとき, 残差 γ が定数であれば問題 (4) は二次錐計画問題 (Second Order Cone Programming, SOCP) と呼ばれる最適化問題になることから, Algorithm 1 の実装には既存の二次錐計画ソルバーを利用することができる. 二次錐計画ソルバーとしては, SeDuMi[8] などがある.

5 提案モデル

従来解法では, 3次元再構成問題を二分法と最適化ソルバーの組み合わせにより解いていた. しかし, 二分法が収束するまで二次錐ソルバーを繰り返し呼び出す必要があることから, 問題サイズが大きくなるにつれて求解に時間がかかってしまうという問題があった. 問題を解くために二分法が必要なのは, 問題 3 の制約条件が非線形であることに原因がある.

Algorithm 1 二分法による問題 (3) の解法 ($p = 2, q = \infty$)

Require: an interval $[\gamma_\ell, \gamma_u]$ known to contain the optimal value r^* and

a tolerance $\epsilon > 0$.

repeat

$\gamma \leftarrow (\gamma_\ell + \gamma_u)/2$;

check if (4) is feasible or not, by solving an associated SOCP problem;

if (4) is feasible **then**

$\gamma_u \leftarrow \gamma$

else

$\gamma_\ell \leftarrow \gamma$

end if

until $\gamma_u - \gamma_\ell \leq \epsilon$;

そこで我々は、先行研究で用いられていた残差 (2) の定義を見直し、3次元再構成問題を線形制約を持つ最適化問題へと帰着させることを考える。式 (1) を変形して得られる次の式

$$y_k^{ij} \mathbf{p}_3^j \mathbf{x}^i = \mathbf{p}_k^j \mathbf{x}^i, \quad k = 1, 2 \quad (5)$$

から、残差 δ^{ij} を次のように定義する。

$$\begin{aligned} \delta^{ij} &= \|d_1^{ij}, d_2^{ij}\|_p, \\ d_k^{ij} &= y_k^{ij} \mathbf{p}_3^j \mathbf{x}^i - \mathbf{p}_k^j \mathbf{x}^i, \quad k = 1, 2 \end{aligned} \quad (6)$$

3次元再構成問題においてはイメージ $\mathbf{y}^{ij}$ は既知であるので、 $\mathbf{P}^j \mathbf{x}^i$ の各列が線形である限り d_k^{ij} は線形であることに注意されたい。この新たな残差 (6) を用いれば、3次元再構成問題を次の最適化問題に帰着させることができる。

$$\left\{ \begin{array}{ll} \min. & \|\delta^{11}, \dots, \delta^{nm}\|_q \\ \text{s.t.} & \delta^{ij} = \|d_1^{ij}, d_2^{ij}\|_p \\ & d_k^{ij} = y_k^{ij} \mathbf{p}_3^j \mathbf{x}^i - \mathbf{p}_k^j \mathbf{x}^i, \quad k = 1, 2 \end{array} \right\} \quad \begin{array}{l} i = 1, \dots, n \\ j = 1, \dots, m \end{array} \quad (7)$$

問題 (3) と問題 (7) の違いは、2本目の制約式の線形性にある。項 $\mathbf{p}_1^j \mathbf{x}^i, \mathbf{p}_2^j \mathbf{x}^i, \mathbf{p}_3^j \mathbf{x}^i$ は、triangulation, camera-resectioning, known-rotation-problem のすべての問題で、未知変数について線形である。known-rotation-problem では $\mathbf{P}^j$ の並進ベクトル $\mathbf{t}^j$ が未知変数なので一見線形でなく見えるが、式を展開すれば

$$\mathbf{P}^j \mathbf{x}^i = [\mathbf{R}^j | \mathbf{t}^j] \mathbf{x}^i = \mathbf{R}^j \begin{bmatrix} x_1^j \\ x_2^j \\ x_3^j \end{bmatrix} + \mathbf{t}^j$$

表 2: 問題 (3) の解法

$p \setminus q$	∞
∞	二分法&線形計画
1	二分法&線形計画
2	二分法&二次錐計画

表 3: 問題 (7) の解法

$p \setminus q$	∞	1	2
∞	線形計画	線形計画	二次計画
1	線形計画	線形計画	二次計画
2	二次錐計画	二次錐計画	最小二乗

となることから線形であることがわかる。よって (7) の制約式

$$d_k^{ij} = y_k^{ij} \mathbf{p}_3^j \mathbf{x}^i - \mathbf{p}_k^j \mathbf{x}^i, \quad k = 1, 2$$

は線形である。ところが, (3) の制約式

$$r_k^{ij} = y_k^{ij} - \frac{\mathbf{p}_k^j \mathbf{x}^i}{\mathbf{p}_3^j \mathbf{x}^i}, \quad k = 1, 2$$

は, 未知変数を含む式 $\mathbf{p}_k^j \mathbf{x}^i$ と $\mathbf{p}_3^j \mathbf{x}^i$ の分数形式であることから線形ではない。

この違いは問題の複雑さに大きく関係しており, たとえば問題 (7) において $p = q = 1$ とした場合には次のような線形計画問題 (Linear Programming, LP) に帰着できる。

$$\left| \begin{array}{ll} \min. & \sum_{i,j,k} |\delta_k^{ij}| \\ \text{s.t.} & d_k^{ij} = y_k^{ij} \mathbf{p}_3^j \mathbf{x}^i - \mathbf{p}_k^j \mathbf{x}^i, \quad k = 1, 2 \quad \left\{ \begin{array}{l} i = 1, \dots, n \\ j = 1, \dots, m \end{array} \right. \end{array} \right. \quad (8)$$

問題 (8) は二分法を用いることなく, 線形計画ソルバーのみで解くことができる。他にも $p = 1, q = 2$ とすれば二次計画問題 (Quadratic Programming, QP) として, $p = q = 2$ とすれば最小二乗問題 (Least Square Problem, LSP) として解くことができる。

問題 (3) と問題 (7) の解法をまとめたものが表 2, 表 3 である。表 2 が $q = \infty$ の場合のみ示してあるのは, 二分法により変数を固定する関係上必ず $q = \infty$ である必要があり, $q = 1, 2$ のときには解けないためである。

6 ふたつの問題の関係

問題 (3) と (7) により得られた解の間には次の定理が成り立つ。

定理. 問題 (3), (7) の一方の問題の最適解は, もう一方の問題の実行可能解である. また, それぞれの最適解を $\bar{\Omega}$, $\tilde{\Omega}$ とし, 解 Ω が与えられたときのそれぞれの問題の目的関数値を $\Gamma(\Omega)$, $\Delta(\Omega)$ とするとき,

$$\begin{aligned}\Delta(\tilde{\Omega}) &\leq \Delta(\bar{\Omega}) \leq L \cdot \Delta(\tilde{\Omega}), \\ \Gamma(\bar{\Omega}) &\leq \Gamma(\tilde{\Omega}) \leq L \cdot \Gamma(\bar{\Omega})\end{aligned}$$

が成り立つ. ただし, L は

$$L = \frac{\max_{i,j} \mathbf{p}_3^j \mathbf{x}^i(\bar{\Omega})}{\min_{i,j} \mathbf{p}_3^j \mathbf{x}^i(\tilde{\Omega})}$$

であたえられる.

また, 以下のいずれかを満たすとき, 二つの問題の最適解は一致する.

1. $\Gamma(\bar{\Omega}) = 0$
2. $\Delta(\tilde{\Omega}) = 0$
3. $L = 1$

この定理から, 一方の問題で得られた最適解をもう一方の問題の近似解と見なせ, その精度の上界が L で与えられると考えることができる. また, どちらかの問題に残差 0 の最適解が存在する場合には, 二つの問題の最適解が必ず一致することに注目されたい.

7 数値実験

known-rotation-problem をベンチマークとして数値実験を行った. 実装には MATLAB (version 7.9, R2009b)[7] 用いている. SOCP ソルバーとして SeDuMi[8] を, LP ソルバーとして GLPK[1] およびその MATLAB インターフェース GLPK MEX[2] を, QP ソルバーとして MATLAB optimization toolbox から quadprog 関数を, LSM ソルバーとして MATLAB 標準の mldivide 関数を利用している.

数値実験の手順を以下に示す.

1. 乱数を用いてオブジェクト座標 $\mathbf{x}^i$, カメラ行列 $\mathbf{P}^j$ を生成する. このとき, $\mathbf{p}_3^j \mathbf{x}^i > 0$ を満たすため, カメラが全オブジェクトをぐるりと取り囲むような構造を与えた.

表 4: 実験結果 (試行回数:10 回)

残差	p	q	問題	誤差	計算時間 (s)	速度 (倍)
γ	2	∞	二分法 & 二次錐計画	0.000706	11.551	1x
	2	∞	二次錐計画	0.000761	0.479	24x
δ	1	∞	線形計画	0.001568	0.196	59x
	1	1	線形計画	0.000813	0.399	29x
	2	2	最小二乗	0.000684	0.011	1050x

2. イメージ $\mathbf{y}^{ij}$ を次式より求める.

$$\mathbf{y}^{ij} = \left[\frac{\mathbf{p}_1^j \mathbf{x}^i}{\mathbf{p}_3^j \mathbf{x}^i}, \frac{\mathbf{p}_2^j \mathbf{x}^i}{\mathbf{p}_3^j \mathbf{x}^i} \right]^\top$$

3. カメラの回転行列 $\mathbf{R}^j$ とイメージ $\mathbf{y}^{ij}$ に適当な誤差を与えたものをパラメータとして, known-rotation-problem を解く.

4. 得られた最適解 $\bar{\mathbf{x}}^i$, $\bar{\mathbf{t}}^j$ と真値との (L_2 ノルムによる) 平均誤差

$$\frac{1}{n} \sum_{i=1}^n \|\bar{\mathbf{x}}^i - \mathbf{x}^i\|_2 + \frac{1}{m} \sum_{j=1}^m \|\bar{\mathbf{t}}^j - \mathbf{t}^j\|_2$$

を評価する.

オブジェクト数 $n = 30$, カメラ数 $m = 10$ としたときの結果を表 4 に示す. この結果から, 提案したモデル (残差 δ) が既存の手法にくらべ精度・速度ともに優れたパフォーマンスを示していることがわかる. 特に最小二乗問題として解いた $p = q = 2$ の提案モデルでは, 従来手法の 1000 倍という速度が出ている. これだけの高速化が実現できた理由は, 一つには二分法のための反復の有無によるところが大きい. もう一つの理由としては, 問題そのものがかなり解きやすい形になっていることが挙げられる. 特に $p = q = 2$ の提案モデルでは制約条件が存在しないため, 提案モデルを用いた他のモデルに比べても高速である.

また図 7 は, カメラ数 $m = 40$, オブジェクト数を $n = 5, 10, 50, 100, \dots, 300$ と変化させたときの, CPU 時間の変化をプロットしたものである. 提案手法とは, 表 4 で最も高速であった $p = q = 2$ のモデルを指す. オブジェクト数が増加しても, 提案手法は従来手法よりも安定して高速であることがわかる. 最も大きなサイズである $n = 300$, $m = 40$ の問題でも,

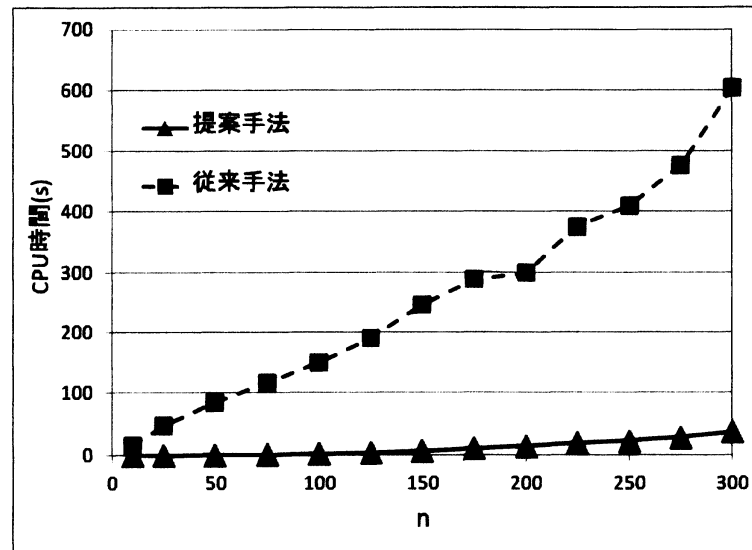


図 2: オブジェクト数 n に対する計算時間の変化 (試行回数:10 回)

従来手法の 603 秒に対して提案手法は 37 秒となっている。

参考文献

- [1] GLPK(GNU Linear Programming Kit)
<http://www.gnu.org/software/glpk/>.
- [2] GLPKMEX - A Matlab MEX Interface For The GLPK Library
<http://glpkmex.sourceforge.net/>.
- [3] Hartley, R.I. and Sturm, P. , Triangulation In *COMPUTER ANALYSIS OF IMAGES AND PATTERNS*, pages 190-197, 1995.
- [4] Kahl, F. , Multiple view geometry and the L_∞ -norm. In *Int. Conf. Computer Vision*, pages 1002-1009, Beijing, China, 2005.
- [5] Kahl, F. and Hartley, R.I. , Multiple View Geometry Under the L_∞ -norm. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1603-1617, September 2008.
- [6] Olsson, C. and Kahl, F. , Generalized Convexity in Multiple View Geometry. In *JOURNAL OF MATHEMATICAL IMAGING AND VISION*, pages 35-51, September 2010.

- [7] MATLAB www.mathworks.com/products/matlab/.
- [8] SeDuMi <http://sedumi.ie.lehigh.edu/>.
- [9] Seo, Y. and Hartley, R.I. , Sequential L_∞ Norm Minimization for Triangulation. In *Computer Vision - ACCV 2007*, vol. 4844, pages 322-331, 2007.
- [10] SPDA In <http://sdpa.indsys.chuo-u.ac.jp/sdpa/>.