

# 様々な広視野カメラを利用可能な オンライン SfM システムの構築

井上 優希<sup>1,a)</sup> 鳥居 秋彦<sup>1,b)</sup> 奥富 正敏<sup>1,c)</sup>

**概要:** Structure from Motion (SfM) の発展により, 撮影対象の 3D モデルの復元やカメラの位置姿勢推定を, 画像から容易に行うことが可能になりつつある. 本論文では, 広範囲を撮影できるカメラを利用することで効率よくシーンを撮影し, オンラインで復元を行う SfM システムを提案する. 投影モデルが異なる様々な広視野カメラを単一のシステムで利用可能にするため, 各投影モデルを表す投影関数と逆投影関数を一様な形で表現することで, カメラの種類に依らないシステムの構築を行った. さらに, 実装上の工夫として, 処理時間の大部分を占める特徴点検出と特徴点マッチングを画像ごとに並列化することで, 効率のよいシステムを実現した. また, 広視野カメラで屋内, 屋外のシーンを撮影して復元を行い, 実験を通して提案システムの有効性を示す.

## 1. はじめに

コンピュータビジョンの分野において, 多視点の画像を用いたオブジェクトやシーンの三次元形状復元は盛んに研究されている問題である. 多視点の画像群から特徴点の三次元位置とカメラの位置姿勢を推定する Structure from Motion (SfM) は, Bundler [16], [17] や VisualSFM [19] といったシステムが提案され, 多視点画像を用意すれば三次元復元を容易に行うことが可能になりつつある. また, 街全体などの大規模なシーンの復元に成功 [2] するなど, 発展をしている.

実際に SfM でシーンを復元する場合, 対象シーンを撮影をしながら同時に復元結果を確認することが出来れば, まだ撮影されていない領域や復元が不十分な領域を撮影者が復元結果を元に判断して撮影にフィードバックすることが出来る. しかし全ての画像を一度に入力して復元を行う従来システムの場合は, 画像が追加される毎に最初から処理をやり直すことになり非効率であった. そこで, 逐次的に入力される画像に対して効率的に処理を行うことで, 入力画像を追加するたびに復元結果が更新される実用的なシステム [8], [21], [22] が提案されている. また, 複数のカメラを組み合わせた広視野カメラの位置姿勢の推定 [13] や, ワイドベースラインのパノラマ画像を用いた SfM [14], [18] な

どで広視野カメラを用いた SfM が提案されている.

本論文では復元処理の効率化に加え, 広視野カメラを利用することで撮影の効率化を図り, 効率の良い実用的な SfM システムを提案する. 広視野カメラには魚眼レンズカメラや全方位ミラーカメラ, Ladybug カメラのような投影の異なる様々なカメラがある. それぞれのカメラごとに SfM システムを構築すると煩雑なシステムになってしまう. そこで提案システムでは, 各投影モデルを表す投影関数と逆投影関数を一様な形で表現し, カメラの種類に依存せず単一のシステムで様々な広視野カメラを利用可能なオンライン SfM システムを構築する. さらに, 処理時間の大部分を占める特徴点検出と特徴点マッチングを画像ごとに並列化して処理をすることで, 処理効率を高めた実装を行う. 2 章で様々な広視野カメラの投影とそのキャリブレーション手法を説明し, 投影関数と逆投影関数を一様な形で表現する. 3 章では, 2 章で表現した投影関数と逆投影関数を用いて三次元復元を行うオンライン SfM システムの詳細を述べる. 4 章で屋内外のシーンを様々な広視野カメラで撮影して復元を行い, 提案システムの有効性を示す.

## 2. 様々な広視野カメラとその投影モデル

広視野カメラには様々なカメラが存在し, カメラによって投影が異なる. 提案システムでは, カメラごとに異なる投影を以下に述べる投影関数  $p$  と逆投影関数  $q$  の形で表現することで, カメラの種類に依らずに広視野カメラを扱えるようにシステムを構築する.

カメラの投影中心を原点としてカメラに固定したカメラ

<sup>1</sup> 東京工業大学  
Tokyo Institute of Technology

a) yinoue@ok.ctrl.titech.ac.jp

b) torii@ctrl.titech.ac.jp

c) mxo@ctrl.titech.ac.jp

座標系上の空間点を  $\mathbf{X} = (X, Y, Z)^T$ , 画像に固定した画像座標系で表された画像上の点を  $\mathbf{u} = (u, v)^T$  とする.  $\mathbf{X}$  からカメラの投影中心に向かってきた光線が画像上で点  $\mathbf{u}$  に投影されるとき, その関係を投影関数  $p$  を用いて表す.

$$p(\mathbf{X}) : \mathbf{X} \mapsto \mathbf{u} \quad (1)$$

反対に, 画像上の点  $\mathbf{u}$  から, 投影中心と  $\mathbf{u}$  を結ぶ光線の方法を表した三次元方向ベクトル  $\mathbf{x} = (x, y, z)^T$  を求める変換を逆投影関数  $q$  で表す.

$$q(\mathbf{u}) : \mathbf{u} \mapsto \mathbf{x} \quad (2)$$

続いて, 透視投影カメラに加えて広視野カメラの例を挙げ, それらの投影モデルを投影関数  $p$  と逆投影関数  $q$  の形で表現しキャリブレーションの方法を説明する.

## 2.1 透視投影カメラ

本論文では, 通常のレンズや広角レンズを装着した一般的なカメラを透視投影カメラと呼ぶ. 透視投影は, 図 1a のような投影中心に向かってくる光線が  $Z = f$  の投影面に投影されるというモデルである. 投影関数と逆投影関数は次のように表される.

$$p(\mathbf{X}) = \frac{f}{Z} \cdot \begin{pmatrix} X \\ Y \end{pmatrix} + \mathbf{u}_0 \quad (3)$$

$$q(\mathbf{u}) = \begin{pmatrix} (\mathbf{u} - \mathbf{u}_0) \\ f \end{pmatrix} \quad (4)$$

焦点距離  $f$  と主点  $\mathbf{u}_0 = (u_0, v_0)^T$  は内部パラメータである. Zhang [20] の手法であらかじめキャリブレーションする, または, 画像の Exif 情報に含まれる焦点距離の情報を元に推定することで既知のパラメータとなり, 空間点  $\mathbf{X}$  から画像上の点  $\mathbf{u}$  への変換と画像上の点  $\mathbf{u}$  から光線の方法  $\mathbf{x}$  への変換を行うことができる.

## 2.2 魚眼レンズカメラ

本論文では, 魚眼レンズを装着し画角  $180^\circ$  程度でカメラ前方半球の範囲を撮影することができるカメラを魚眼レンズカメラと呼ぶ.

魚眼レンズカメラを表す投影モデルとして OCamCalib Toolbox [15] の投影モデルを用いる. この投影モデルでは, 透視投影モデルで  $Z = f$  平面としていた投影面を図 1b のように  $Z = z(\rho)$  と拡張することで, 魚眼レンズの投影を表現する. 投影面を表す  $z(\rho)$  は, 歪みのない理想的な画像面にとった座標系  $(u', v')^T$  で表した投影点  $\mathbf{u}'$  の中心からの距離  $\rho = \sqrt{(u')^2 + (v')^2}$  の多項式で表される.

$$z(\rho) = \sum_{i=0}^n a_i \rho^i \quad (5)$$

多項式の係数  $a_i$  はカメラの内部パラメータで, OCamCalib

Toolbox を用いてキャリブレーションする.  $(u', v')^T$  座標系と実際の画像座標系  $(u, v)^T$  はアフィン変換で変換される. このときのパラメータである,  $2 \times 2$  線形変換行列  $A$  と平行移動ベクトル  $\mathbf{u}_0 = (u_0, v_0)^T$  もカメラの内部パラメータで,  $a_i$  と同様に OCamCalib Toolbox を用いてキャリブレーションする. 逆投影関数  $q$  は, 式 5 を用いて以下のように表される.

$$q(\mathbf{u}) = \begin{pmatrix} A^{-1}(\mathbf{u} - \mathbf{u}_0) \\ z(\rho) \end{pmatrix} \quad (6)$$

ここで,  $A$ ,  $\mathbf{u}_0$ ,  $z(\rho)$  はキャリブレーションにより既知であり, 画像上の点  $\mathbf{u}$  から光線の方法  $\mathbf{x}$  への変換を行うことができる.

反対に, 空間点  $\mathbf{X}$  を歪みのない理想的な画像面  $(u', v')^T$  での投影点  $\mathbf{u}'$  へ投影するとき,  $\mathbf{u}'$  は次のように表される.

$$\mathbf{u}' = \frac{r(\theta)}{\sqrt{X^2 + Y^2}} \begin{pmatrix} X \\ Y \end{pmatrix} \quad (7)$$

$\theta$  は  $XY$  平面からの光線の角度であり,  $r(\theta)$  は  $\theta$  の多項式で次のように表される.

$$r(\theta) = \sum_{i=0}^m b_i \theta^i \quad (8)$$

多項式の係数  $b_i$  は  $a_i$  と同時にキャリブレーションする. 投影関数  $p$  は, 式 7 を用いて以下のように表される.

$$\begin{aligned} p(\mathbf{X}) &= A\mathbf{u}' + \mathbf{u}_0 \\ &= A \frac{r(\theta)}{\sqrt{X^2 + Y^2}} \begin{pmatrix} X \\ Y \end{pmatrix} + \mathbf{u}_0 \end{aligned} \quad (9)$$

ここで,  $A$ ,  $\mathbf{u}_0$ ,  $r(\theta)$  はキャリブレーションにより既知であり, 空間点  $\mathbf{X}$  から画像上の点  $\mathbf{u}$  への変換を行うことができる.

## 2.3 全方位ミラーカメラ

本論文では, 一般的なカメラのレンズ前方にミラーを固定しミラーに周囲を映して撮影するカメラを全方位ミラーカメラと呼ぶ. 典型的な全方位ミラーカメラでは, 水平方向  $360^\circ$  垂直方向に  $100^\circ$  程度の範囲を撮影することができる.

本システムでは, 全方位ミラーカメラの投影を表す投影モデルとして, 魚眼レンズカメラと同様に OCamCalib Toolbox の投影モデルを用いる. 投影面を  $Z = z(\rho)$  と拡張することで, 全方位ミラーの投影を表すことができる (図 1c).

## 2.4 Ladybug カメラ (パノラマ画像)

本論文では, 透視投影カメラを複数組み合わせることで周囲  $360^\circ$  のほぼ全球に近い範囲を撮影することができるカメラを Ladybug カメラと呼ぶ. Google Street View\*1

\*1 <http://maps.google.com/>

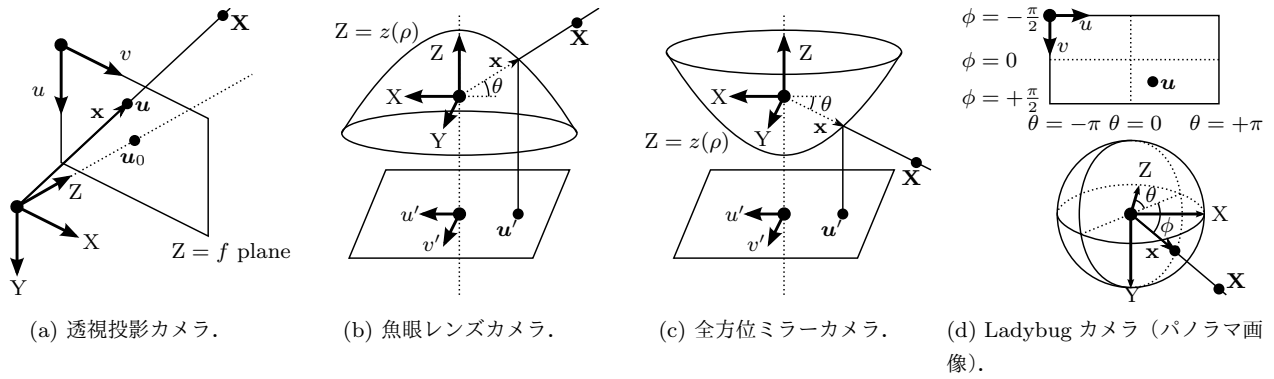


図 1: 様々な広視野カメラの投影モデル.

における街のパノラマ画像など、主に車両にカメラを固定して走行しながら周囲の環境を撮影する用途で用いられている。Ladybug カメラは透視投影カメラを複数組み合わせたカメラであり、一度の撮影で複数の透視投影画像が撮影される。提案システムでは Ladybug カメラにより合成されたパノラマ画像を入力画像として扱い、パノラマ画像は正距円筒図法 (Equirectangular projection) で投影されているものとする。

正距円筒図法では、光線の経度  $\theta$  と緯度  $\phi$  が等間隔に横幅  $w$  縦幅  $h$  の画像にマッピングされる (図 1d)。画像座標系  $(u, v)^T$  と緯度経度座標系  $(\theta, \phi)^T$  の変換は次のように表される。

$$\begin{pmatrix} \theta \\ \phi \end{pmatrix} = \begin{pmatrix} (u - \frac{w}{2}) \frac{2\pi}{w} \\ (v - \frac{h}{2}) \frac{\pi}{h} \end{pmatrix} \quad (10)$$

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \theta \cdot \frac{w}{2\pi} + \frac{w}{2} \\ \phi \cdot \frac{h}{\pi} + \frac{h}{2} \end{pmatrix} \quad (11)$$

逆投影関数  $q$  は、式 10 を用いて以下のように表される。

$$q(\mathbf{u}) = \begin{pmatrix} \cos \phi \sin \theta \\ \sin \phi \\ \cos \phi \cos \theta \end{pmatrix} \quad (12)$$

ここで、 $\theta$  と  $\phi$  は画像上の点  $\mathbf{u}$  と既知である画像のサイズ  $(w, h)$  で表されるため、 $\mathbf{u}$  から光線方向  $\mathbf{x}$  への変換を行うことができる。

反対に、投影関数  $p$  は式 11 と式 12 から以下のように表される。

$$p(\mathbf{X}) = \begin{pmatrix} \arctan2(\mathbf{X}, Z) \cdot \frac{w}{2\pi} + \frac{w}{2} \\ \arcsin\left(\frac{Y}{\sqrt{X^2 + Y^2 + Z^2}}\right) \cdot \frac{h}{\pi} + \frac{h}{2} \end{pmatrix} \quad (13)$$

ここで、画像のサイズ  $(w, h)$  は既知であり、空間点  $\mathbf{X}$  から画像上の点  $\mathbf{u}$  への変換を行うことができる。ただし、 $\arctan2$  は次のように定義する。

$$\arctan2(\mathbf{X}, Z) = 2 \arctan \frac{X}{\sqrt{Z^2 + X^2} + Z} \quad (14)$$

### 3. オンライン SfM システム

#### 3.1 提案システムの概要

提案システムでは、シーンの撮影をしながら画像を入力して復元を逐次的に行う。撮影された画像は、カメラの Wi-Fi 転送機能や FlashAir などのカメラに転送機能を付加する SD カードなどを用いて計算機に転送され、システムに画像が 1 枚入力されるごとに以下に説明する処理を行い、逐次的に復元を行う。ただし、入力画像は 2 章で述べた手法によりキャリブレーション済みであるとする。

1. 画像が入力されると、同じ空間点が投影された画像上の点を画像間に対応付けるために、特徴点の検出と特徴量の記述を行う。本システムでは、回転・拡大縮小に不変かつ微小な視点変化に不変な特徴量を抽出できる SIFT [9] または SURF [3] アルゴリズムを用いて特徴点の検出・記述を行う。
2. 画像の特徴を 1 つのベクトルで表した Bag of Feature (BoF) [4] ベクトルを計算する。似ている画像間では BoF ベクトルの類似度が高くなるため、BoF ベクトルを用いて画像類似度を計算することができる。
3. 入力画像とすでに復元された画像との特徴点マッチングを行う。このとき、正しい特徴点对応を得られる画像は入力画像との共通視野を十分に持つ画像である。そこで、特徴点マッチングを行う画像を BoF ベクトルにもとづく画像類似度でフィルタリングし、入力画像との共通視野を持たない可能性が高い画像を取り除くことで、特徴点マッチングを効率的に行う。画像間で特徴量を近似最近傍探索で対応づけた後、5 点アルゴリズム [11] により特徴点对応の検証を行い誤対応を取り除く。
4. カメラの位置姿勢の復元を行う。入力画像のカメラの位置姿勢が復元できると、特徴点の三次元位置を三角測量の原理で推定し、空間点を復元する。
5. カメラの位置姿勢と空間点をバンドル調整によって最適化する。バンドル調整では、空間点を各画像に再投

影したときの再投影誤差を最小化することで、復元したカメラの位置姿勢と空間点の最適化を行う。

上述した処理のうち特に、特徴点对応の検証、カメラの位置姿勢の復元、バンドル調整はカメラの投影モデルを用いた処理である。提案システムでは、投影関数  $p$  と逆投影関数  $q$  を一様な形で表現することによりこれらの処理においてカメラの投影の違いに依らない処理を行っており、その詳細について述べる。また、効率のよい実装の方法についても述べる。

### 3.2 特徴点对応の検証

近似最近傍探索ライブラリである FLANN [10] により、画像間で特徴量を比較して一番近い特徴量を持つ特徴点を高速に対応付けた後、Distance ratio test [9] により信頼性が低い対応を除去する。最近傍探索では誤対応であっても特徴量が近い特徴点が必ず対応付けられてしまうため、この時点で得られる特徴点对応は誤対応を多く含んでいる。誤対応を多く含む特徴点对応を元に復元を行うと復元が破綻する可能性があるため、安定した復元を行うためには誤対応を除去することが重要である。そこで、5点アルゴリズム [11] と RANSAC [5] を用いて特徴点对応のエピポーラ拘束を検証し誤対応を除去することでロバストな特徴点对応を得て安定した復元を行う。

ある空間点  $\mathbf{X}_i$  を2つのカメラ（投影中心  $\mathbf{c}_i, \mathbf{c}'_i$ ）に投影したときの光線  $(\mathbf{x}_i, \mathbf{x}'_i)$  との間には、それらが同一の平面上に存在するエピポーラ拘束が存在し、この拘束は基本行列  $\mathbf{E}$  を用いて次のように表される。

$$\mathbf{x}'_i{}^T \mathbf{E} \mathbf{x}_i = 0 \quad (15)$$

5点アルゴリズムは、2つの画像間の5組の  $\{\mathbf{x}_i \leftrightarrow \mathbf{x}'_i\}$  対応から基本行列  $\mathbf{E}$  を推定する手法である。提案システムでは、逆投影関数  $\mathbf{x}_i = q(\mathbf{u}_i)$  で特徴点  $\mathbf{u}_i$  を光線  $\mathbf{x}_i$  に変換し、5点アルゴリズムを用いる。

RANSAC はロバストなモデル推定手法で、繰り返しランダムサンプリングしたデータからモデルを仮定し、仮定したモデルを支持するデータ（インライア）の数が最大になるモデルを解とする。

1. 特徴点对応の中から5組の対応をランダムサンプリングする。
2. 5点アルゴリズムを用いて  $\mathbf{E}$  を仮定する。
3. 各特徴点对応のうち、仮定した  $\mathbf{E}$  により表されるエピポーラ拘束を満たすインライアを数えて  $\mathbf{E}$  の検証をする。

以上の処理を繰り返し行い、推定された  $\mathbf{E}$  を満たさない特徴点对応を誤対応として除去する。ある特徴点对応がエピポーラ拘束を満たさかどうかの検証はエピポーラ平面の角度に対する閾値処理によって行う。すなわち次の条件を満たす特徴点对応を、エピポーラ拘束を満たすインライア

とする。

$$\left| \frac{\pi}{2} - \arccos \frac{\mathbf{x}'_i{}^T \mathbf{E} \mathbf{x}_i}{\|\mathbf{x}'_i\| \|\mathbf{E} \mathbf{x}_i\|} \right| < T_{5pt} \quad (16)$$

ただし、 $\|\cdot\|$  は  $L_2$  ノルムを表す。閾値  $T_{5pt}$  は調整パラメータで、カメラの画角やキャリブレーションの精度により最適な値が異なる。4章の実験では経験的に値を決定して実験を行った。

### 3.3 カメラの位置姿勢の復元

カメラの位置姿勢の復元では、最初の2枚の画像で行われるカメラの位置姿勢の復元（初期復元）と初期復元後に入力された画像のカメラの位置姿勢の復元（追加復元）とで復元手法が異なる。

- 初期復元では、基本行列  $\mathbf{E}$  を特異値分解により並進移動ベクトルと回転行列に分解する手法 [7] でカメラの位置姿勢を推定する。特異値分解によりカメラの位置姿勢の候補が4つ得られるが、それぞれの位置姿勢で仮の空間点の復元を行い、正しい空間点を復元できた位置姿勢をカメラの位置姿勢として復元する。透視投影カメラなど撮影できる範囲がカメラ前方に限定されているカメラの場合は、最もカメラ前方に復元された空間点が多い位置姿勢を正しいものとして選択する。全方位ミラーカメラなどの撮影できる範囲がカメラ前方に限定されないカメラの場合は、空間点の復元を行った際の再投影誤差が最も小さい位置姿勢を正しいものとして選択する。
- 追加復元では、P3P アルゴリズム [6] と RANSAC [5] を用いてカメラの位置姿勢をロバストに推定する。P3P アルゴリズムは、画像上の点と空間点の3組の対応からカメラの位置姿勢を推定する手法である。提案システムでは、逆投影関数  $\mathbf{x}_i = q(\mathbf{u}_i)$  で特徴点  $\mathbf{u}_i$  を光線  $\mathbf{x}_i$  に変換し、光線と空間点の対応に対して P3P アルゴリズムを用いてカメラの位置  $\mathbf{t}$  と姿勢  $\mathbf{R}$  を推定する。RANSAC の繰り返し処理におけるカメラの位置姿勢の検証には投影中心から  $\mathbf{X}_i$  への光線と  $\mathbf{x}_i$  の角度を用いる。すなわち、次の条件を満たす空間点・光線対応をインライアとする。

$$\arccos \frac{\mathbf{x}_i{}^T (\mathbf{R} \mathbf{X}_i + \mathbf{t})}{\|\mathbf{x}_i\| \|\mathbf{R} \mathbf{X}_i + \mathbf{t}\|} < T_{p3p}$$

閾値  $T_{p3p}$  は調整パラメータで、4章の実験では経験的に値を決定して実験を行った。

### 3.4 バンドル調整

バンドル調整は非線形最小二乗問題として投影関数  $p$  を用いて次のように定式化され、ガウス・ニュートン法やレベンバーグ・マーカート法などを用いて解くことができる。

$$\operatorname{argmin}_{\mathbf{X}_i, \mathbf{R}_j, \mathbf{t}_j} \sum_{i \in A} \sum_{j \in T_i} \|p(\mathbf{R}_j \mathbf{X}_i + \mathbf{t}_j) - \mathbf{u}_{ij}\|^2 \quad (17)$$

$A$  はバンドル調整を行う空間点の添字集合,  $T_i$  は空間点  $\mathbf{X}_i$  と対応する特徴点を持っている画像の添字集合である.  $\mathbf{R}_j$  と  $\mathbf{t}_j$  はそれぞれカメラ  $j$  の回転と並進を表し,  $\mathbf{u}_{ij}$  は空間点  $\mathbf{X}_i$  に対応するカメラ  $j$  における特徴点である.

本システムでは Ceres Solver [1] によりレベンバーグ・マーカート法でバンドル調整を行う. Ceres Solver は非線形最小二乗問題ソルバーであり, 対象の問題における残差を与えることで様々な非線形最小二乗問題を解くことができる.

SfM において画像が入力される度に全ての空間点に対してバンドル調整を行うと, カメラと空間点の数の増加にともなって処理時間が増加してしまう. そこで提案システムでは, 誤差の大きさに応じてバンドル調整の適用範囲を切り替えることで計算の効率化を図る.

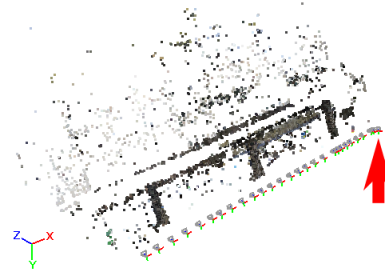
- 画像が入力され空間点が復元された時点で,  $A$  を時系列近傍の画像に投影されている空間点のみに限定して局所的にバンドル調整を行う (ローカルバンドル調整).
- ローカルバンドル調整を行ったにも関わらず再投影誤差が閾値よりも大きい場合は,  $A$  を全ての空間点としてバンドル調整を行う (グローバルバンドル調整). また, 蓄積誤差を解消するために定期的にグローバルバンドル調整を行う.

### 3.5 実装上の工夫

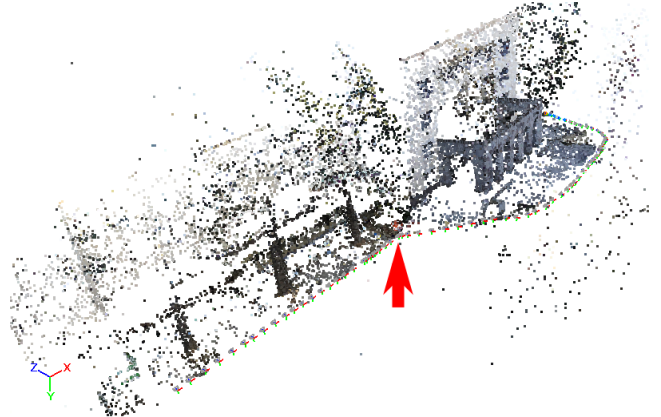
これまで述べた処理は, 並列処理可能なグループに分けることができる. 特徴点検出と BoF ベクトル計算処理は画像ごとに行う処理であり, それ以降の処理は, 複数の画像の特徴点や BoF ベクトルをもとにして行う処理である. したがって, 特徴点検出と BoF ベクトル計算処理はそれ以降の処理と独立して行うことができ, 特徴点検出と BoF ベクトル計算を行うグループとそれ以降の処理を行うグループに分けて並列化して処理することで処理効率を高めることができる.

また, 特徴点マッチングにおいて近似最近傍探索による対応付けや特徴点对応の検証は, 特徴点マッチングを行う画像ペアごとに独立した処理になっている. 特徴点マッチングを行う画像ペアごとに処理を並列化することで特徴点マッチングの処理効率を高めることができる.

特徴点検出や特徴点マッチングはシステム全体の処理時間に対して大きな割合を占める処理であり, これらを並列化することで効率のよい実装を行った.



(a) 透視投影カメラ. 28 台のカメラと 6,186 点の空間点を復元した.



(b) 魚眼レンズカメラ. 73 台のカメラと 24,983 点の空間点を復元した.

図 2: 透視投影カメラと魚眼レンズカメラを用いた建物の復元結果. 色の付いた点は空間点, 灰色の四角錐はカメラを表している. 透視投影カメラによる復元では赤矢印で示した 28 フレーム目より後のカメラの位置姿勢の復元に失敗している.

## 4. 実験

### 4.1 魚眼レンズカメラによる実験

同一視点から単焦点レンズで透視投影カメラ画像を撮影し, 魚眼コンバーターを装着して魚眼レンズカメラ画像を撮影して実験を行った. 建物の壁面を移動しながら撮影した 73 枚ずつの画像 (3.5M px) をそれぞれ入力して復元を行い, 透視投影カメラと魚眼レンズカメラでの復元結果を示す (図 2). パラメータは  $T_{5pt} = 0.1^\circ$ ,  $T_{p3p} = 0.2^\circ$  を用いた.

透視投影カメラで撮影した画像による復元では, カメラモーションが大きいフレーム (図 3) において共通視野が少なくなり, 特徴点对応を十分に得ることができなかったためにカメラの位置姿勢の復元に失敗している. カメラの位置姿勢の復元の失敗により, それ以降のフレームの復元も連鎖的に失敗して復元が途切れてしまっている.

一方, 魚眼レンズカメラで撮影した画像による復元では, 透視投影カメラを用いた復元でカメラの位置姿勢の復元に失敗したフレームにおいてもカメラの位置姿勢の復元を行





図 3: カメラモーションが大きいフレーム. 図 2 中の赤矢印の箇所.

うことができ, 入力したすべての画像について復元できた. 魚眼レンズカメラは透視投影カメラよりも広範囲を撮影できるため, カメラモーションが大きい画像間でも共通視野を十分に確保し特徴点对応を得ることができ, 安定して復元が行えることを示している.

#### 4.2 全方位ミラーカメラでの実験

本実験では, 双曲面ミラーである GoPano plus を単焦点レンズに装着して全方位ミラーカメラ画像 (14M px) を撮影して実験を行った. 全方位ミラーカメラで四方を壁に囲まれた部屋の中央を一直線に進みながらから内装を撮影した 16 枚の画像を用いて復元を行った結果を示す (図 4). パラメータは  $T_{5pt} = 0.3^\circ$ ,  $T_{p3p} = 0.3^\circ$  を用いた.

全方位ミラーカメラでは周囲  $360^\circ$  の広範囲を撮影できるため, 部屋の四方の壁や天井, 床までくまなく撮影し復元することができる.

#### 4.3 Ladybug カメラ (パノラマ画像) での実験

Ladybug カメラで道路上を一直線に進みながらから左右に見える建物を撮影した 55 枚のパノラマ画像 (12.5M px) データセット [12] を用いて復元を行った結果を示す (図 5). パラメータは  $T_{5pt} = 0.3^\circ$ ,  $T_{p3p} = 0.3^\circ$  を用いた.

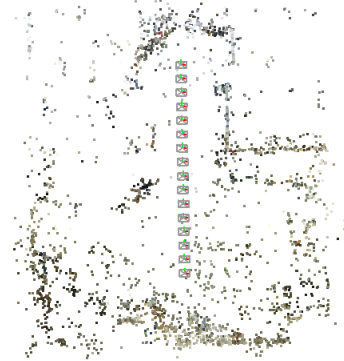
Ladybug カメラは一度の撮影でほぼ全球に近い範囲を撮影することができ, 天井までくまなく撮影し復元することができる.

### 5. おわりに

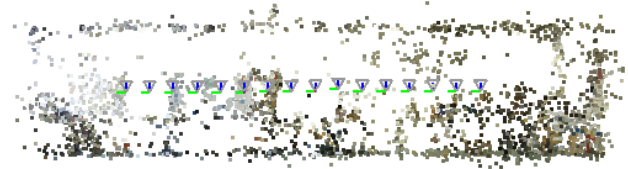
本論文では, 広視野カメラを用いて効率的に撮影をして三次元復元を行える実用的なシステムを提案した. 広視野カメラには様々な投影を持つカメラがあり, カメラごとに異なる投影を投影関数と逆投影関数で一樣に表現することで一般化して扱えるようにし, 個々の投影モデルに依らず様々な広視野カメラを利用可能なオンライン SfM システムを構築した. また, 処理時間の大部分を占める特徴点検出と特徴点マッチングを画像ごとに並列化することで効率



(a) 入力画像の一部.



(b) 部屋を上から見た結果.



(c) 部屋を横から見た結果.

図 4: 全方位ミラーカメラを用いた復元結果. 16 台のカメラと 3,611 点の空間点を復元した.

のよい実装を行った.

提案システムで用いた投影関数と逆投影関数の表現は, カメラごとに 1 つ持つ投影中心に空間点から向かってくる光線が画像上に投影されるというモデルであり, ステレオカメラなど 1 つの投影中心では表せない投影を持つカメラを表現することができない. そこで今後の課題として, 投影中心が複数あるような投影モデルへの対応が挙げられる. これにより, さらに多くの種類のカメラで SfM を行うことが可能になる. また, SfM で復元した三次元点群モデルでは物体の遮蔽関係やテクスチャを表現することができないが, サーフェイス復元手法でサーフェイスを復元することによりそれらの情報を表現することが可能になるため, サーフェイス復元手法との統合を行うことも今後の課題である.

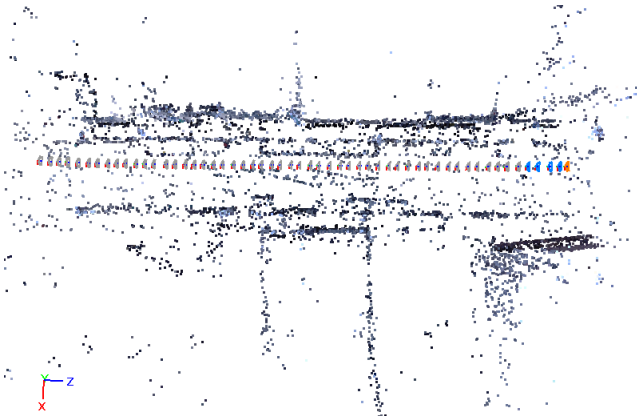
謝辞 本研究は科研費 25240025, 2470016 の一部助成を受けたものである.

#### 参考文献

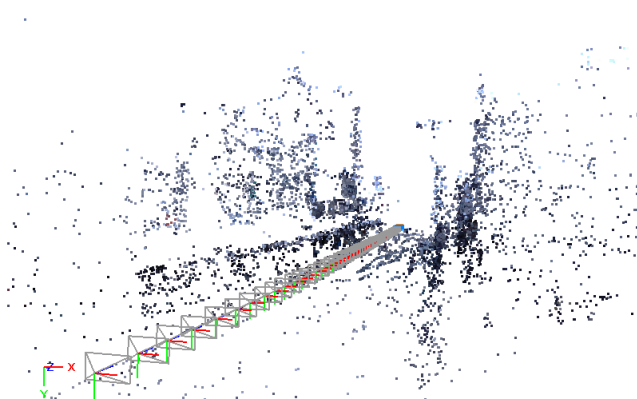
- [1] Agarwal, S., Mierle, K. and Others: Ceres Solver, <https://code.google.com/p/ceres-solver/>.
- [2] Agarwal, S., Snavely, N., Simon, I., Seitz, S. M. and Szeliski, R.: Building rome in a day, *Computer Vision, 2009 IEEE 12th International Conference on*, IEEE,



(a) 入力画像の一部.



(b) 上から見た結果.



(c) 斜めから見た結果.

図 5: Ladybug カメラ (パノラマ画像) データセット [12] を用いた復元結果. 55 台のカメラと 8,945 点の空間点を復元した.

pp. 72–79 (2009).

[3] Bay, H., Tuytelaars, T. and Van Gool, L.: Surf: Speeded up robust features, *Computer Vision–ECCV 2006*, Springer, pp. 404–417 (2006).

[4] Csurka, G., Dance, C., Fan, L., Willamowski, J. and Bray, C.: Visual categorization with bags of keypoints, *Workshop on statistical learning in computer vision, ECCV*, Vol. 1, No. 1–22, pp. 1–2 (2004).

[5] Fischler, M. A. and Bolles, R. C.: Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, *Communications of the ACM*, Vol. 24, No. 6, pp. 381–395 (1981).

[6] Gao, X.-S., Hou, X.-R., Tang, J. and Cheng, H.-F.: Complete solution classification for the perspective-three-point problem, *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, Vol. 25, No. 8, pp. 930–943 (2003).

[7] Hartley, R. I.: Estimation of relative camera positions for uncalibrated cameras, *Computer Vision–ECCV’92*, Springer, pp. 579–587 (1992).

[8] Hoppe, C., Klopschitz, M., Rimpler, M., Wendel, A., Kluckner, S., Bischof, H. and Reitmayr, G.: Online

Feedback for Structure-from-Motion Image Acquisition., *BMVC*, pp. 1–12 (2012).

[9] Lowe, D. G.: Distinctive image features from scale-invariant keypoints, *International journal of computer vision*, Vol. 60, No. 2, pp. 91–110 (2004).

[10] Muja, M. and Lowe, D. G.: Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration., *VISAPP (1)*, pp. 331–340 (2009).

[11] Nistér, D.: An efficient solution to the five-point relative pose problem, *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, Vol. 26, No. 6, pp. 756–770 (2004).

[12] Sakurada, K., Okatani, T. and Deguchi, K.: Detecting Changes in 3D Structure of a Scene from Multi-view Images Captured by a Vehicle-mounted Camera, *CVPR* (2013).

[13] Sato, T., Ikeda, S. and Yokoya, N.: Extrinsic camera parameter recovery from multiple image sequences captured by an omni-directional multi-camera system, *Computer Vision–ECCV 2004*, Springer, pp. 326–340 (2004).

[14] Sato, T., Pajdla, T. and Yokoya, N.: Epipolar geometry estimation for wide-baseline omnidirectional Street View images, *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, IEEE, pp. 56–63 (2011).

[15] Scaramuzza, D., Martinelli, A. and Siegwart, R.: A toolbox for easily calibrating omnidirectional cameras, *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, IEEE, pp. 5695–5701 (2006).

[16] Snavely, N., Seitz, S. M. and Szeliski, R.: Photo tourism: exploring photo collections in 3D, *ACM transactions on graphics (TOG)*, Vol. 25, No. 3, ACM, pp. 835–846 (2006).

[17] Snavely, N., Seitz, S. M. and Szeliski, R.: Modeling the world from internet photo collections, *International Journal of Computer Vision*, Vol. 80, No. 2, pp. 189–210 (2008).

[18] Torii, A., Havlena, M. and Pajdla, T.: From google street view to 3d city models, *Computer vision workshops (ICCV Workshops), 2009 IEEE 12th international conference on*, IEEE, pp. 2188–2195 (2009).

[19] Wu, C.: VisualSFM : A Visual Structure from Motion System, <http://ccwu.me/vsfm/>.

[20] Zhang, Z.: A flexible new technique for camera calibration, *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, Vol. 22, No. 11, pp. 1330–1334 (2000).

[21] 阿達大地: ビジョンベースのオンライン 3 次元復元システムの構築, 修士論文, 東京工業大学 (2013).

[22] 半澤悠樹: オンライン処理に適した Structure from Motion の実現, 修士論文, 東京工業大学 (2012).