

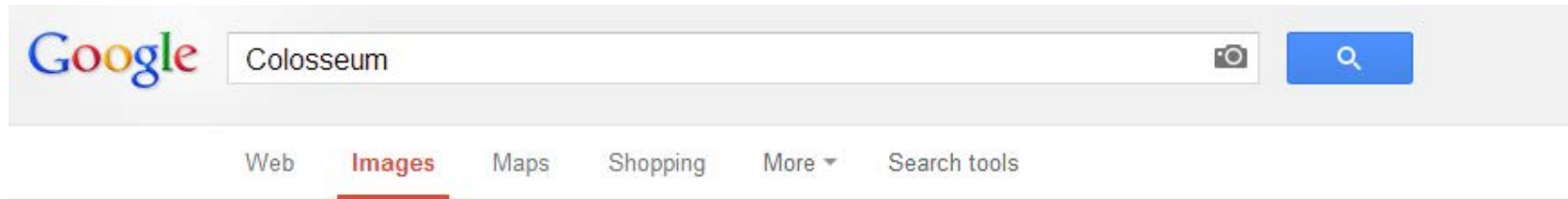
第7回測位技術懇話会

画像センサによる自己位置推定の技術動向

2018年 1月 26日
オムロン株式会社
諏訪正樹



“Building Rome in a Day”



Related searches: [colosseum rome](#) [colosseum inside](#) [colosseum wallpaper](#) [colosseum interior](#) [great wall of china](#)



“ ”

<http://grail.cs.washington.edu/rome/>

Structure from Motion (SfM)

カメラ位置とシーンの3次元復元を同時におこなう

2,106 images, 819,242 points

SLAM : Simultaneous Localization and Mapping

ロボットの位置姿勢 (3DOF/6DOF) + 地図 (2D/3D)

Vacuum robot



<https://www.smartrobotcleaners.com/lg-robotic.html>

Articulated Robot Motion for SLAM



<http://robotglobe.org/robot-with-in-hand-eye-solves-slam-problem-in-better-way/>

ロボットの位置姿勢（3DOF/6DOF） + 地図（2D/3D）

状態遷移モデル

$$P(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t, \mathbf{m})$$

センサ観測モデル

$$P(\mathbf{z}_t | \mathbf{x}_t, \mathbf{m})$$

地図更新モデル

$$P(\mathbf{m} | \{\mathbf{z}_s, \mathbf{u}_s : s = 1, \dots, t\})$$

$\mathbf{z}_t$: 観測量

$\mathbf{x}_t$: 状態

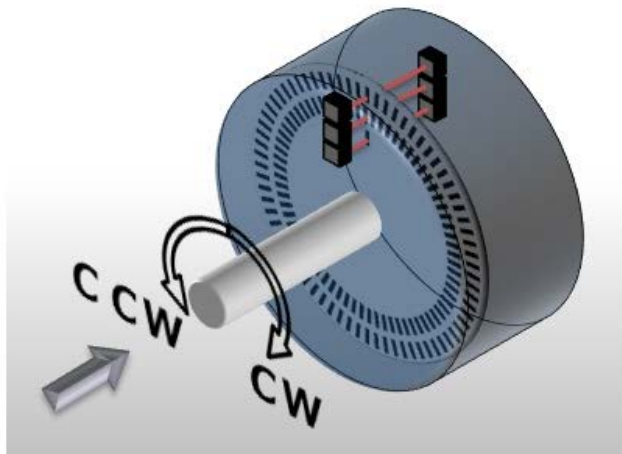
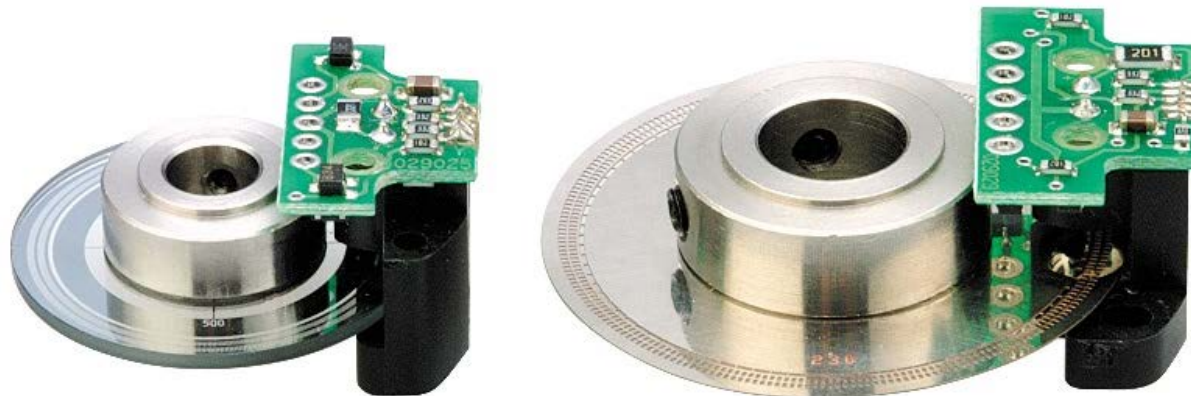
$\mathbf{m}$: 地図

$\mathbf{u}_t$: ロボットの制御量



オドメトリ

- ・ 自走ロボットに取り付けられた車輪回転量から位置推定
- ・ エンコーダをジャイロなどのセンサ情報と足し合わせて、逐次の位置・姿勢を推定



Visual Odometry



SfMとv-SLAM

基本原理に本質的な違いはない。実用面での制約をどこまでいれるか？の違い。

SfM

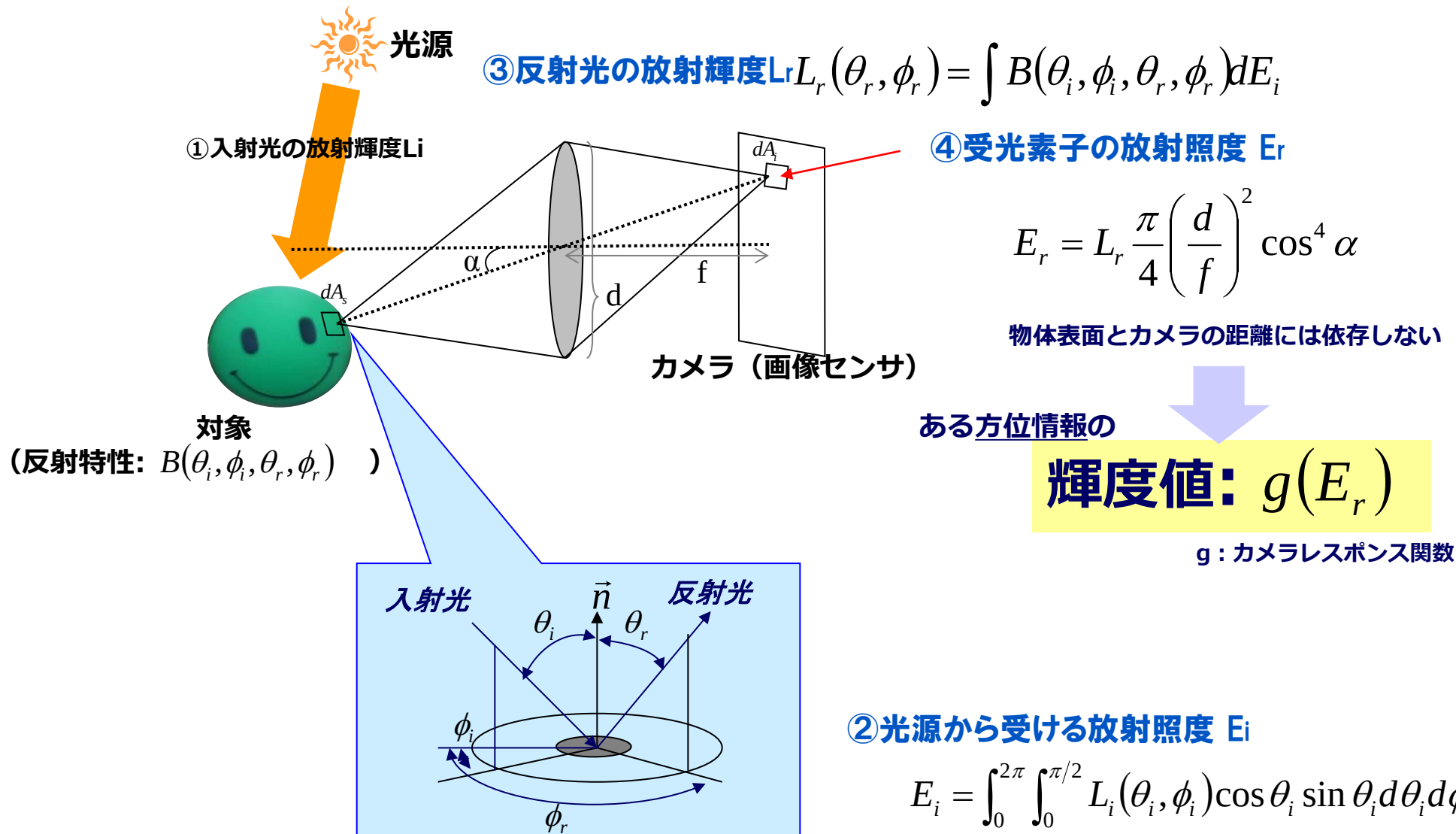
- ・ 入力：多視点画像（時間情報は考慮せず）
- ・ オフライン処理（リアルタイム性不要）
- ・ 出力：一括

v-SLAM

- ・ 入力：動画像（連続時系列画像）
- ・ オンライン処理
- ・ 出力：逐次出力

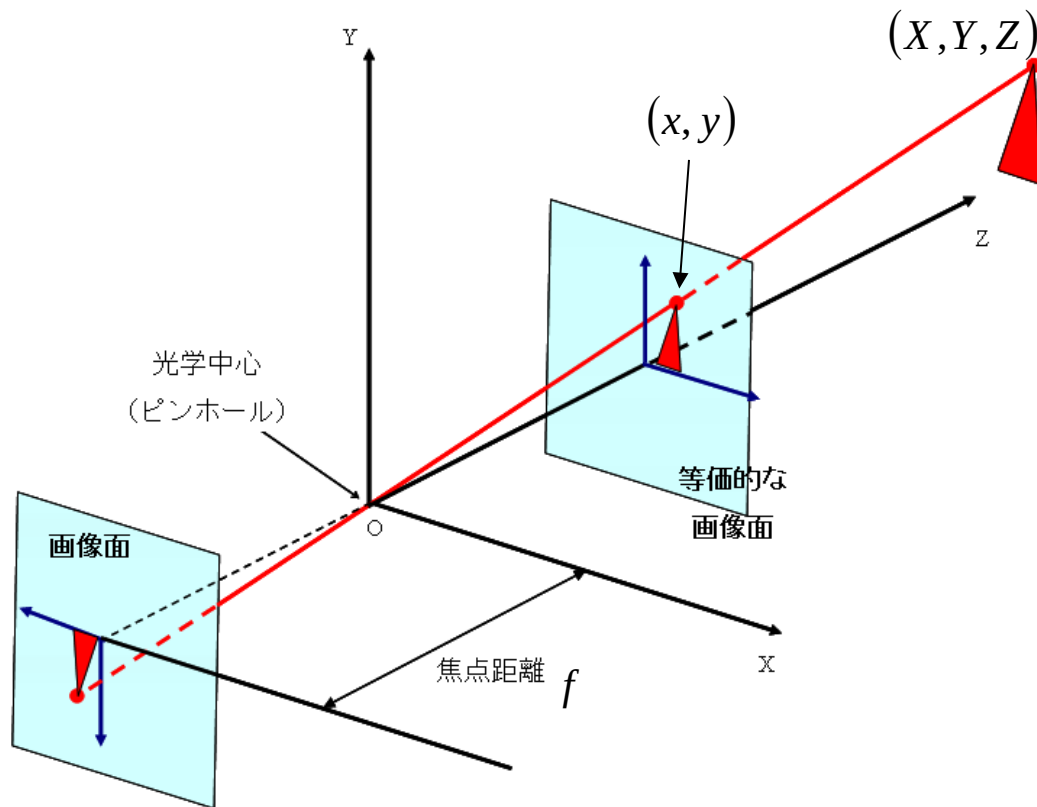
カメラは光の何を記録しているのか？

光の持つ多様な情報のうち、かなり限定された部分のみを切り取っている

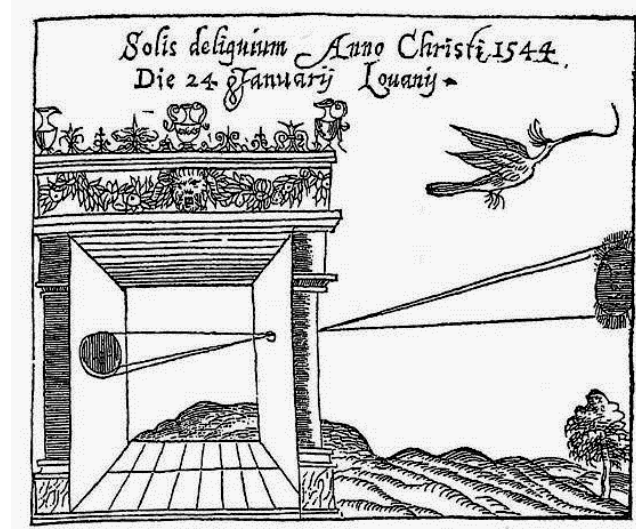


カメラのモデル化

ピンホールカメラ

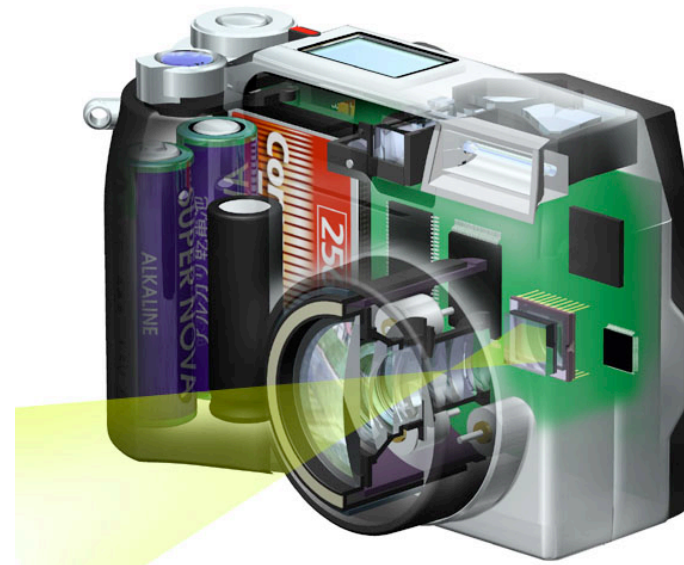


$$(X, Y, Z) \Rightarrow (x, y) = \left(f \frac{X}{Z}, f \frac{Y}{Z} \right)$$



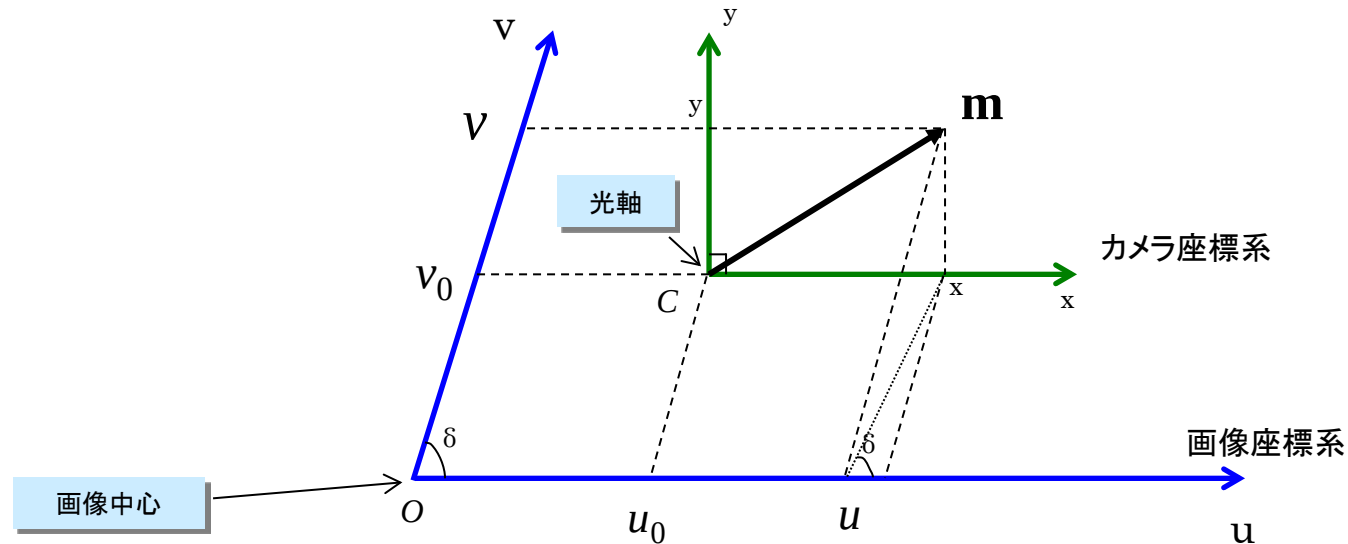
Camera Obscura, Gemma Frisius (1558)

ED2



<http://www.sugilab.net/jk/joho-kiki/1307/index.html>

カメラの内部パラメータ



$$\tilde{\mathbf{m}} \equiv \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} \varepsilon_u x - (\varepsilon_u \cot \delta) y + u_0 \\ \frac{\varepsilon_v}{\sin \delta} y + v_0 \\ 1 \end{pmatrix} \equiv \mathbf{K} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

$\varepsilon_u, \varepsilon_v$ はそれぞれ縦方向, 横方向のスケール因子

透視カメラモデル

透視カメラモデル
(ピンホールカメラモデル)

$$\lambda \tilde{\mathbf{m}} = \mathbf{P} \tilde{\mathbf{X}} = \mathbf{K} \mathbf{M} \tilde{\mathbf{X}}$$

内部パラメータ $\mathbf{K} = \begin{bmatrix} f k_u & f k_s & u_0 \\ 0 & f k_u & v_0 \\ 0 & 0 & 1 \end{bmatrix}$

外部パラメータ $\mathbf{M} = [\mathbf{R} \quad | \quad \mathbf{t}]$

$$\tilde{\mathbf{m}} = \mathbf{K} \tilde{\mathbf{x}}$$

物理的座標 $\tilde{\mathbf{x}}$



画像座標 $\tilde{\mathbf{m}}$

$$\tilde{\mathbf{m}} = \begin{bmatrix} m_1 \\ m_2 \\ m_3 \end{bmatrix} \sim \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

画像座標 $\mathbf{m} = (u, v)$ の斉次座標

一般化したものが

射影カメラモデル

$$\lambda \tilde{\mathbf{m}} = \mathbf{P}_p \tilde{\mathbf{X}}$$

$$\mathbf{P}_p = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix}$$

画像による3D計測

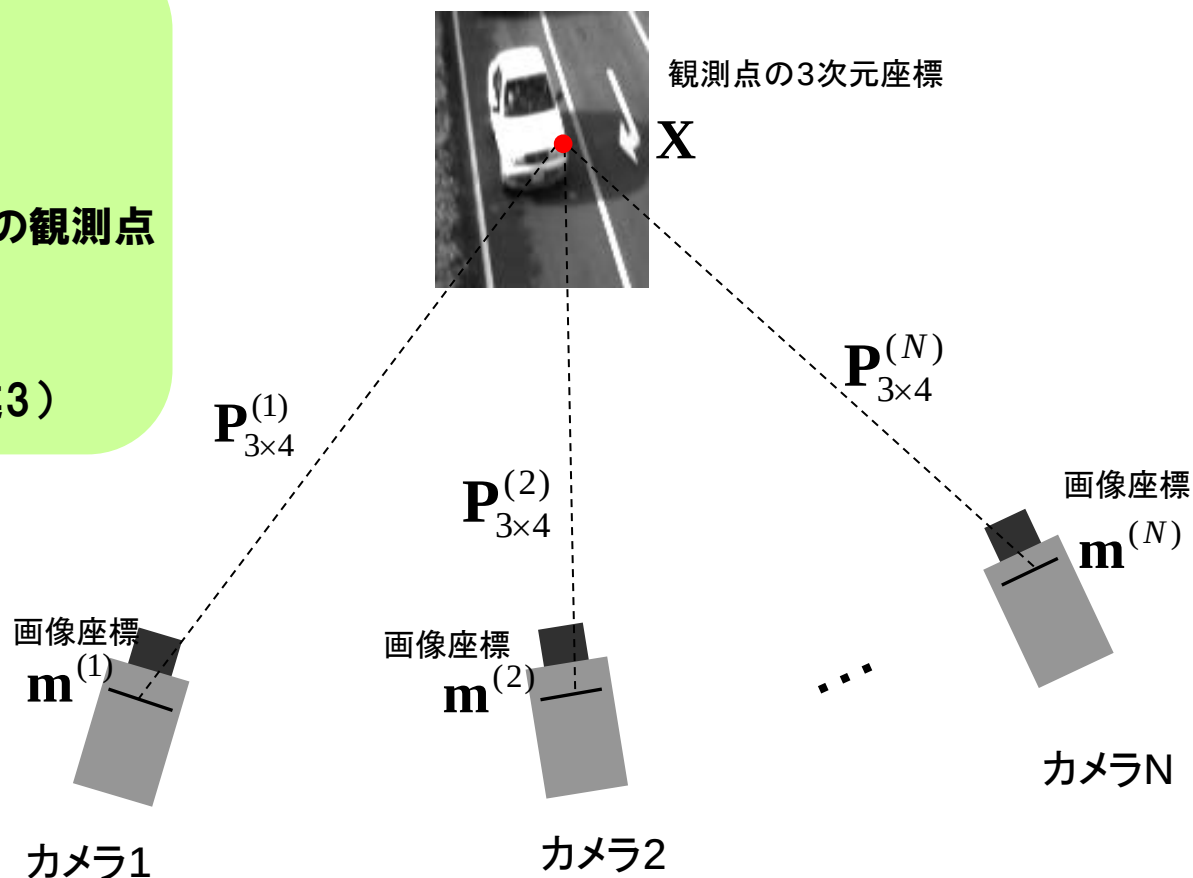
複数台(≥ 2)のカメラがあれば3D計測可能

$$\tilde{\mathbf{m}} \cong \mathbf{P}_{3 \times 4} \tilde{\mathbf{X}}$$

↑
画像座標

↓
3次元空間中の観測点

11自由度
(内部パラメータ5 + 回転3 + 並進3)



3D計測の具体的方法

線形解法の例

$$\begin{aligned} \tilde{\mathbf{m}}^{(i)} &\cong \mathbf{P}_{3 \times 4}^{(i)} \tilde{\mathbf{X}} \quad (i=1,2,\dots,N) \\ &\Updownarrow \\ \begin{cases} u^{(i)} = \frac{(\mathbf{p}_1^{(i)}, \tilde{\mathbf{X}})}{(\mathbf{p}_3^{(i)}, \tilde{\mathbf{X}})} \\ v^{(i)} = \frac{(\mathbf{p}_2^{(i)}, \tilde{\mathbf{X}})}{(\mathbf{p}_3^{(i)}, \tilde{\mathbf{X}})} \end{cases} &\text{ただし } \mathbf{P} \equiv \begin{bmatrix} \mathbf{p}_1^T \\ \mathbf{p}_2^T \\ \mathbf{p}_3^T \end{bmatrix} \end{aligned} \quad \Rightarrow \quad \begin{aligned} &\begin{matrix} \xleftarrow{4} \\ \begin{bmatrix} (\mathbf{p}_1^{(1)})^T - (\mathbf{p}_3^{(1)})^T u^{(1)} \\ (\mathbf{p}_2^{(1)})^T - (\mathbf{p}_3^{(1)})^T v^{(1)} \\ (\mathbf{p}_1^{(2)})^T - (\mathbf{p}_3^{(2)})^T u^{(2)} \\ (\mathbf{p}_2^{(2)})^T - (\mathbf{p}_3^{(2)})^T v^{(2)} \\ \vdots \\ (\mathbf{p}_1^{(N)})^T - (\mathbf{p}_3^{(N)})^T u^{(N)} \\ (\mathbf{p}_2^{(N)})^T - (\mathbf{p}_3^{(N)})^T v^{(N)} \end{bmatrix} \end{matrix} \\ &\begin{matrix} \begin{matrix} \uparrow \\ \downarrow \end{matrix} 2N \end{matrix} \end{aligned} \quad \tilde{\mathbf{X}} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix} \end{aligned}$$

$\mathbf{A}$

カメラNのデータ

$\mathbf{A} \tilde{\mathbf{X}} = \mathbf{0} \Rightarrow \tilde{\mathbf{X}}$ は $\mathbf{A}^T \mathbf{A}$ の最小固有値に対する
固有ベクトルとして求めればよい

ただし $N \geq 2$

線形解法が実用的でない理由

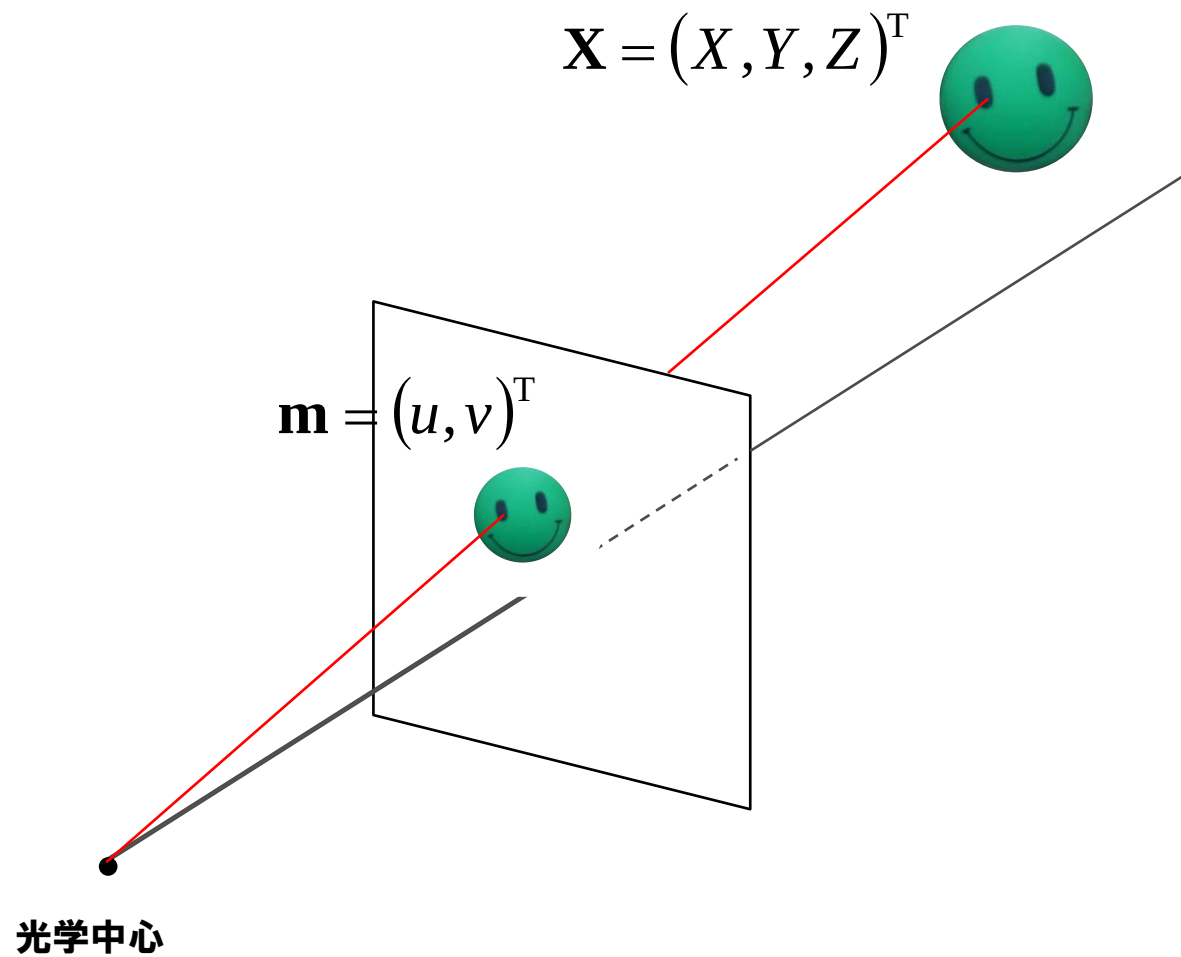
ノイズを考慮していないため不安定

- 光学～信号～アルゴリズムまで多様なノイズ形態
- 外挿の場合、誤差の挙動がまったく読めない

何を最小化しているのかが不明瞭

- 代数的最小化は幾何学的・物理的意味をもたない
 - 単なる12次元パラメータ空間
- 回転行列のベクトル要素は正規直交基底ではない
- 観測空間での誤差ではなく、画像空間での誤差を考えた方が実用上は精度が出やすい
 - 非線形最適化問題

何を最小化するか？

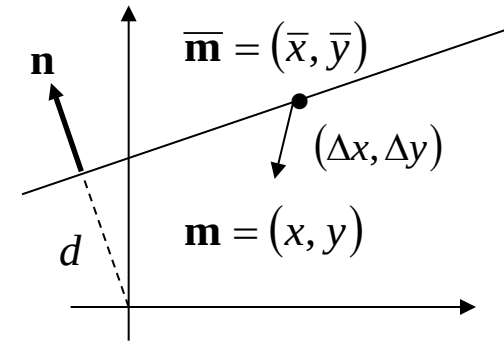


幾何学的あてはめ問題

真の位置 $\bar{\mathbf{m}}_\alpha, \alpha = 1, \dots, N$ が

$$(\mathbf{n}, \bar{\mathbf{m}}_\alpha) = d, \quad \alpha = 1, \dots, N$$

を満たすようなパラメータ (単位ベクトル $\mathbf{n}$, 定数 d)
を観測データ $\mathbf{m}_\alpha, \alpha = 1, \dots, N$ から推定せよ



前提

$$p(\mathbf{m}_\alpha) = \frac{1}{2\pi\sqrt{|\Sigma_\alpha|}} \exp\left\{-\frac{(\mathbf{m}_\alpha - \bar{\mathbf{m}}_\alpha, \Sigma_\alpha^{-1}(\mathbf{m}_\alpha - \bar{\mathbf{m}}_\alpha))}{2}\right\}$$

誤差はデータごとに独立で
正規分布に従う

未知パラメータは2種類ある

$\{\mathbf{n}, d\}$: 構造母数 (最終的に知りたいパラメータ)

$\{\mathbf{m}_\alpha\}$: 局外(攪乱)母数

SfMも幾何学的あてはめ問題の一種

SfMの問題の定式化

3次元空間内に適当に配置されたカメラ L 台で、空間内の適当な N 点を撮影し、 L 枚の画像上で $\mathbf{m}_\alpha^{(i)} = (u_\alpha^{(i)}, v_\alpha^{(i)})^T, i=1, \dots, L, \alpha=1, \dots, N$ が得られたとする。これから点の3次元座標 $\mathbf{X}_\alpha$ 、カメラの位置姿勢 $\mathbf{M}^{(i)}$ 、カメラの内部パラメータ $\mathbf{K}^{(i)}$ を推定せよ

$$\mathbf{P}^{(i)} = \mathbf{K}^{(i)} \mathbf{M}^{(i)} = \mathbf{K}^{(i)} \left[\mathbf{R}^{(i)} \mid \mathbf{t}^{(i)} \right]$$

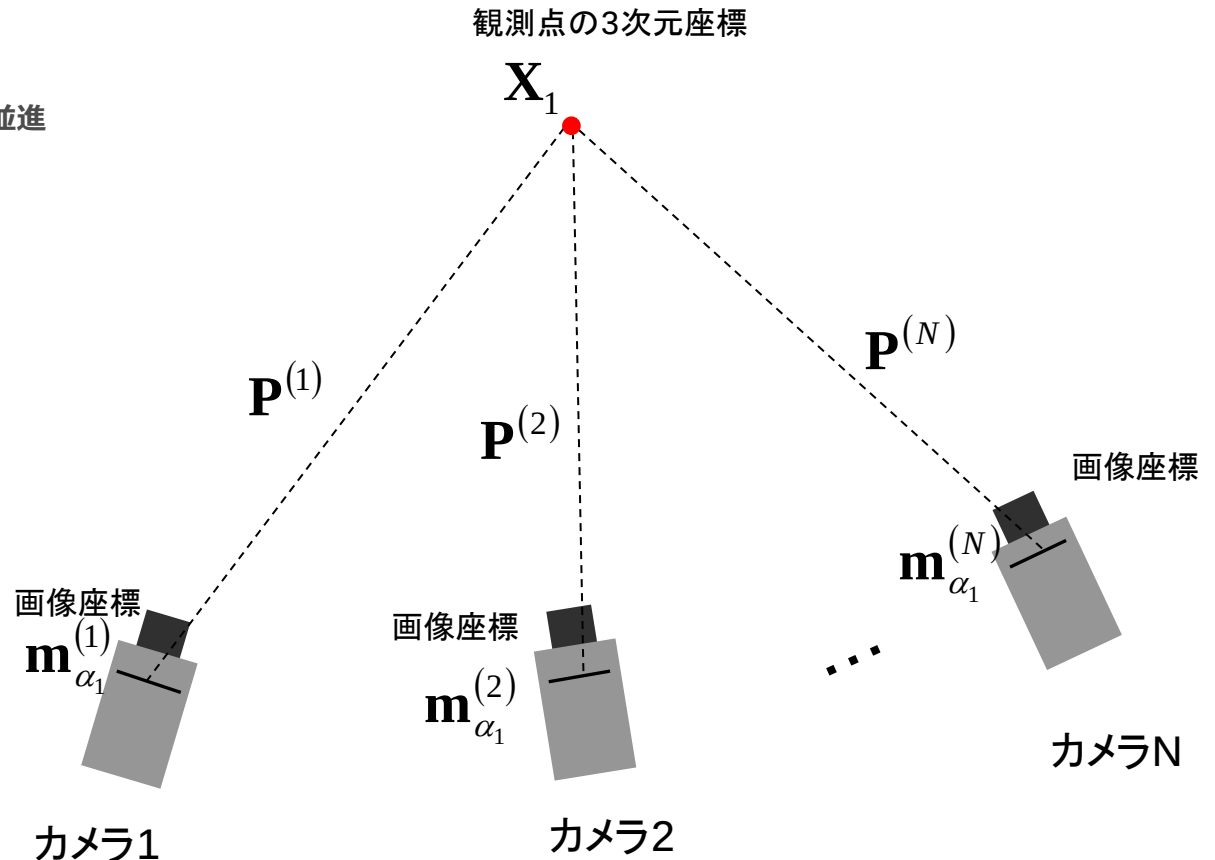
カメラ内部パラメータ カメラ外部パラメータ

カメラ回転 カメラ並進

画像座標 $\Leftrightarrow$ 空間座標

$$\begin{bmatrix} \mathbf{m}_\alpha^{(i)} \\ 1 \end{bmatrix} \propto \mathbf{P}^{(i)} \begin{bmatrix} \mathbf{X}_\alpha \\ 1 \end{bmatrix}$$

↑
ベクトルの長さを
除いて等しい



推定に制約は必要か？

- $P^{(i)}, X_\alpha$ のすべてが未知：射影的不定性 (projective ambiguity) が残る
⇒ 空間中の2直線の**角度や平行性が保存されない**
- カメラ内部パラメータが既知、あるいは一部が未知であれば、**相似変換の不定性**を除いて復元が可能
⇒ **カメラ回転・並進とスケール倍の座標変換に相当する自由度は定められない**
3次元空間の基準座標系に依存 ピンホールカメラの射影変換
- 絶対的な長さ尺度が入れば、ユークリッド変換の不定性を除いて復元可能
⇒ **3次元座標の回転および並進**



・ 実用上は、カメラ内部パラメータのうち**焦点距離 f のみを**
未知とし、残りを既知（校正済み）とすることが多い

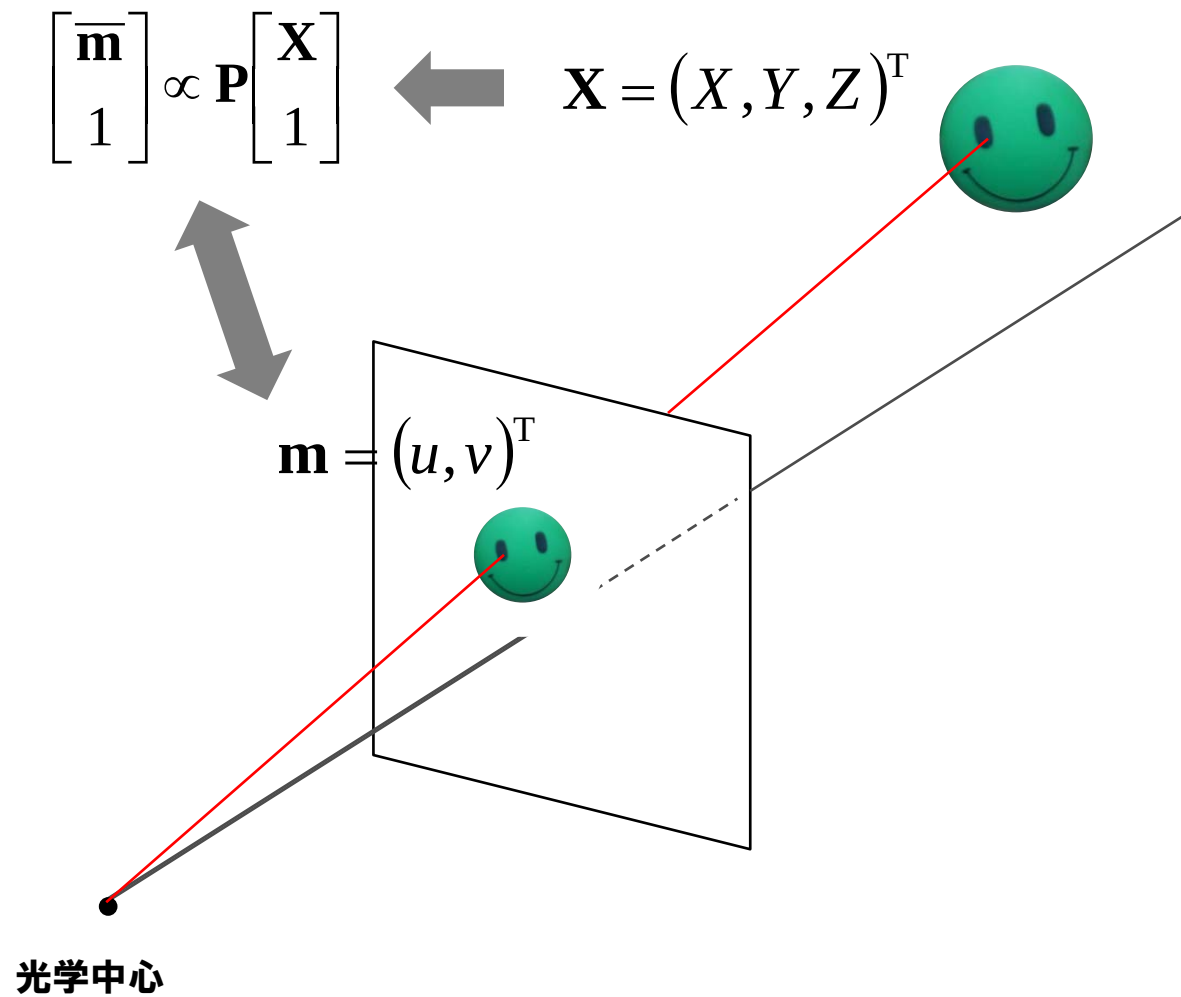
フォーカスチェンジの可能性

$$K = \begin{bmatrix} fk_u & fk_s & u_0 \\ 0 & fk_u & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

- ・ 未知パラメータは**7次元ベクトル**で表すことができる
(回転・並進の自由度 6 + 焦点距離)
- ・ 座標系の選択

⇒ 例えば $R^{(1)} = I, t^{(1)} = 0, t^{(2)}$ の1つの成分を定数 (1) とする

再投影误差



Bundle Adjustmentの統計的推定手法としての側面

3次元空間内に適当に配置されたカメラ L 台で、空間内の適当な N 点を撮影し、 L 枚の画像上で $\mathbf{m}_\alpha^{(i)} = (u_\alpha^{(i)}, v_\alpha^{(i)})^T, i=1, \dots, L, \alpha=1, \dots, N$ が得られたとする。これから点の3次元座標 $\mathbf{X}_\alpha, \alpha=1, \dots, N$ およびパラメータ $\mathbf{p}_i, i=1, \dots, L$ を推定せよ



$$\mathbf{m}_\alpha^{(i)} = \overline{\mathbf{m}}_\alpha^{(i)} + \mathbf{n}^{(i)}$$

観測ノイズ $\sim N(\mathbf{0}, \sigma^2 \mathbf{I})$

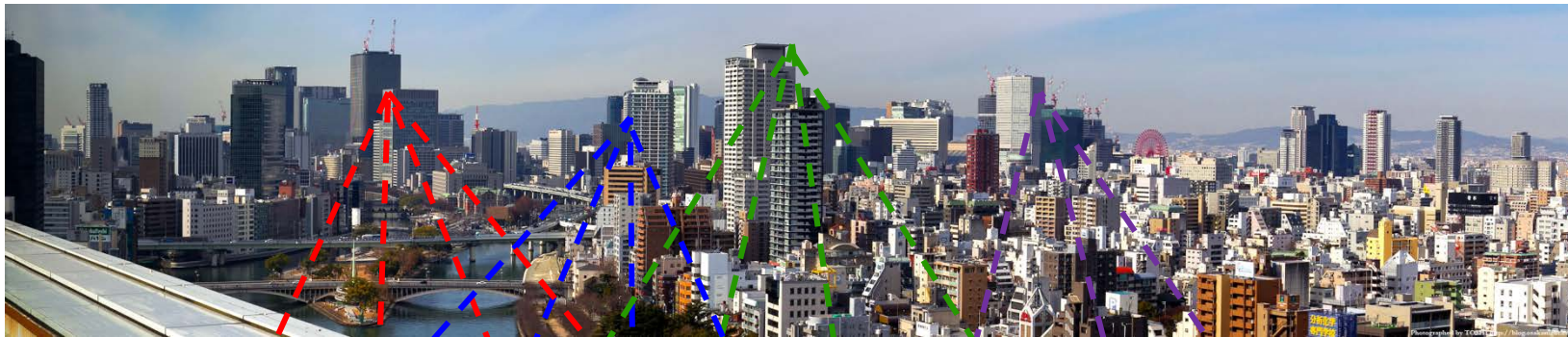
Bundle Adjustmentにおける再投影誤差最小化 = 最尤推定

最尤推定量は $\sigma^2 \rightarrow 0$ の極限においてクラメール・ラオの下界を達成する不偏推定量

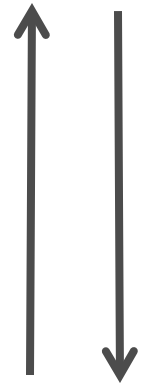
⇒ SfMのような画像センシングでは誤差分散は十分小さくできる

v-SLAM

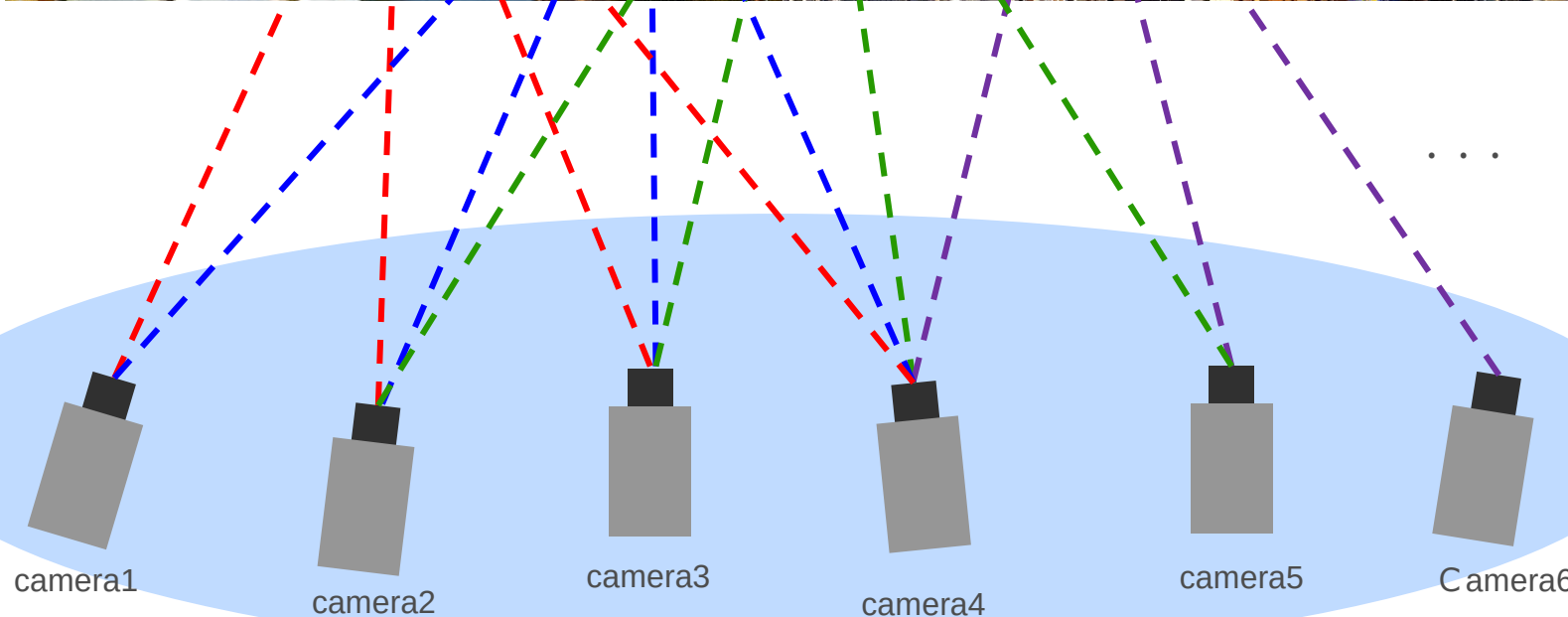
カメラ位置の初期値が与えて、localization, mappingの繰り返りで逐次推定



Mapping



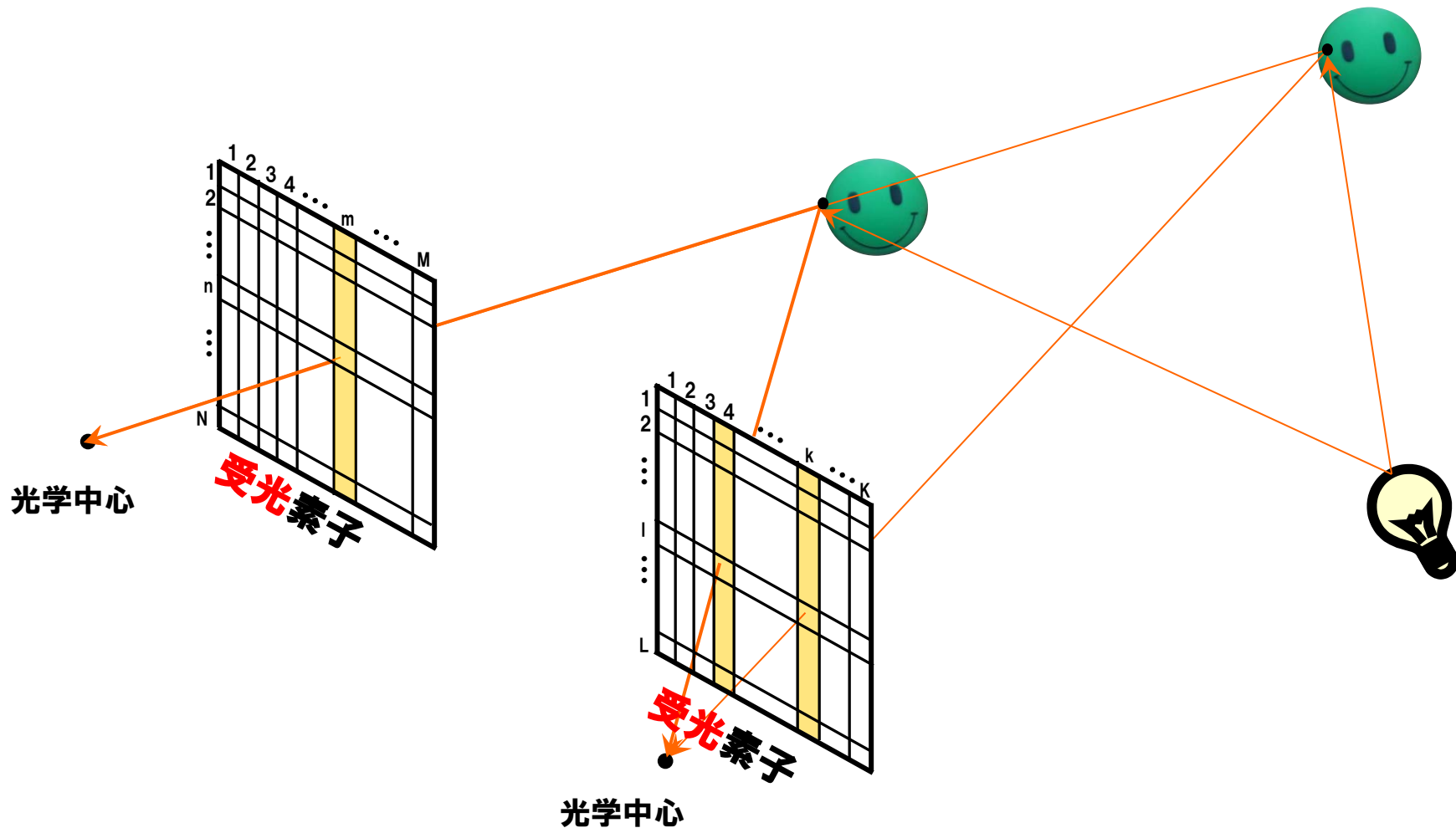
Localization



http://www.osakanight.com/skyscrapers/photo/title/osaka_skyscrapers_p14.htm

画像による距離計測の原理（三角測量法）

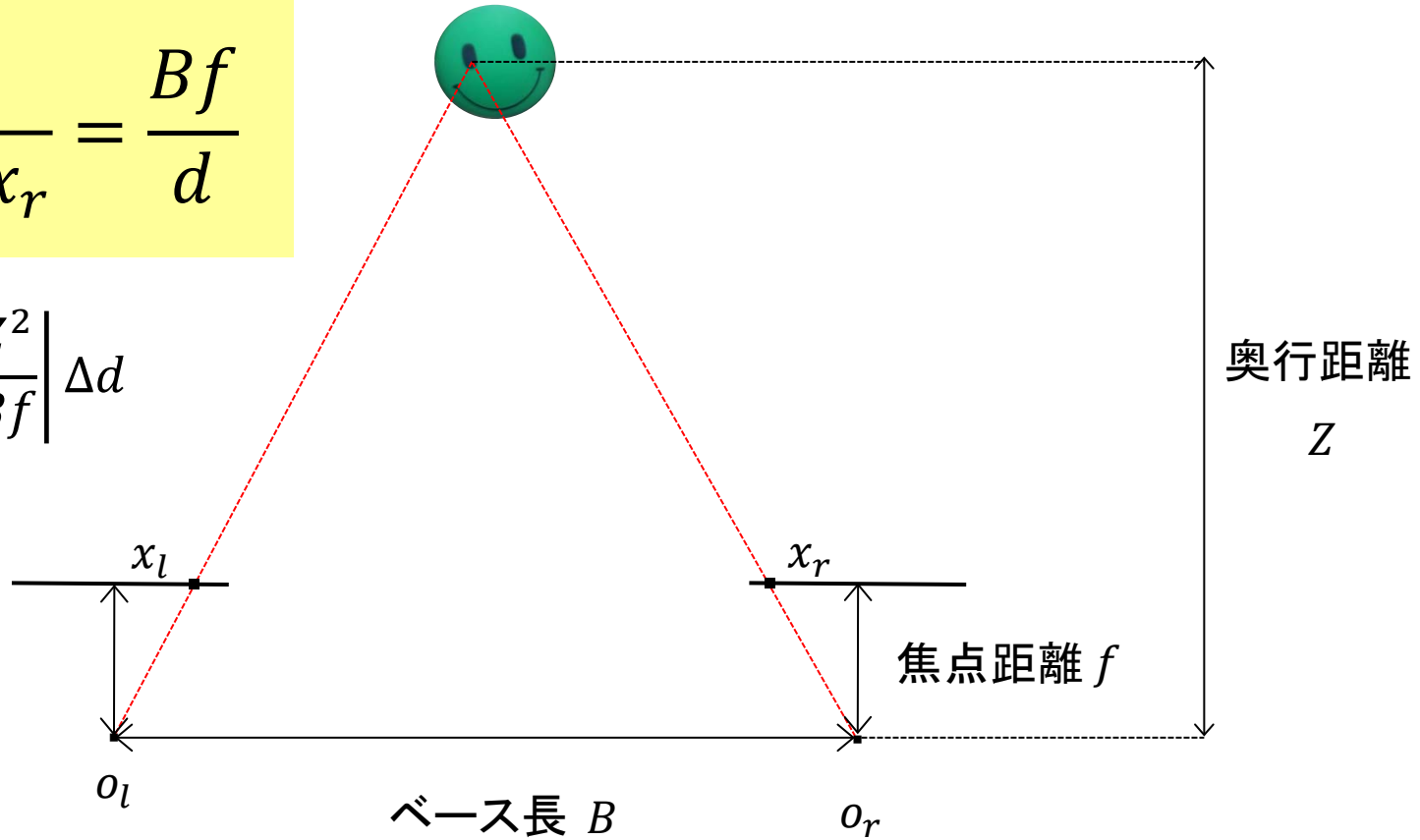
それぞれの**受光部**が「**光を受けた位置**」から距離を算出



三角測量の測距式（平行ステレオ）

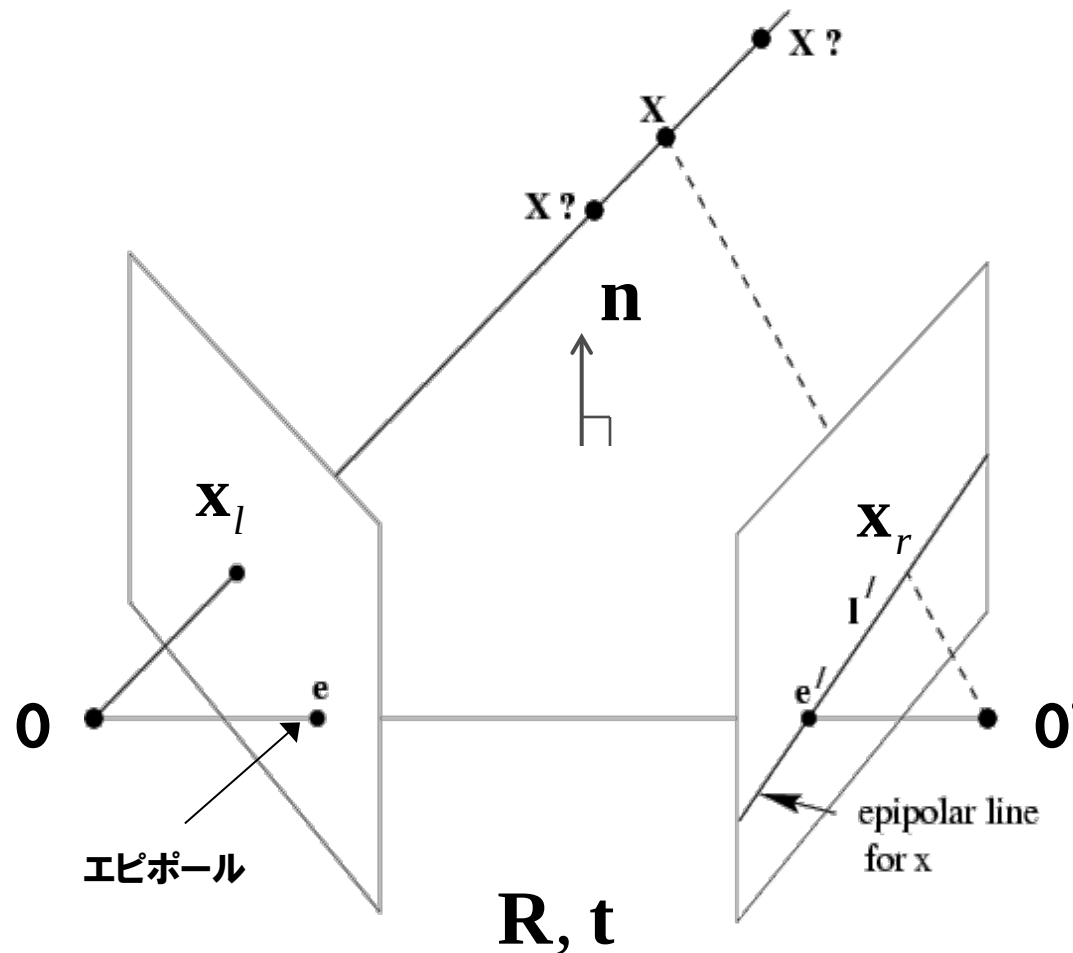
$$Z = \frac{Bf}{x_l - x_r} = \frac{Bf}{d}$$

$$\Delta Z = \left| \frac{Z^2}{Bf} \right| \Delta d$$



2台のカメラによる3次元復元

Epipolar拘束の利用による3D復元



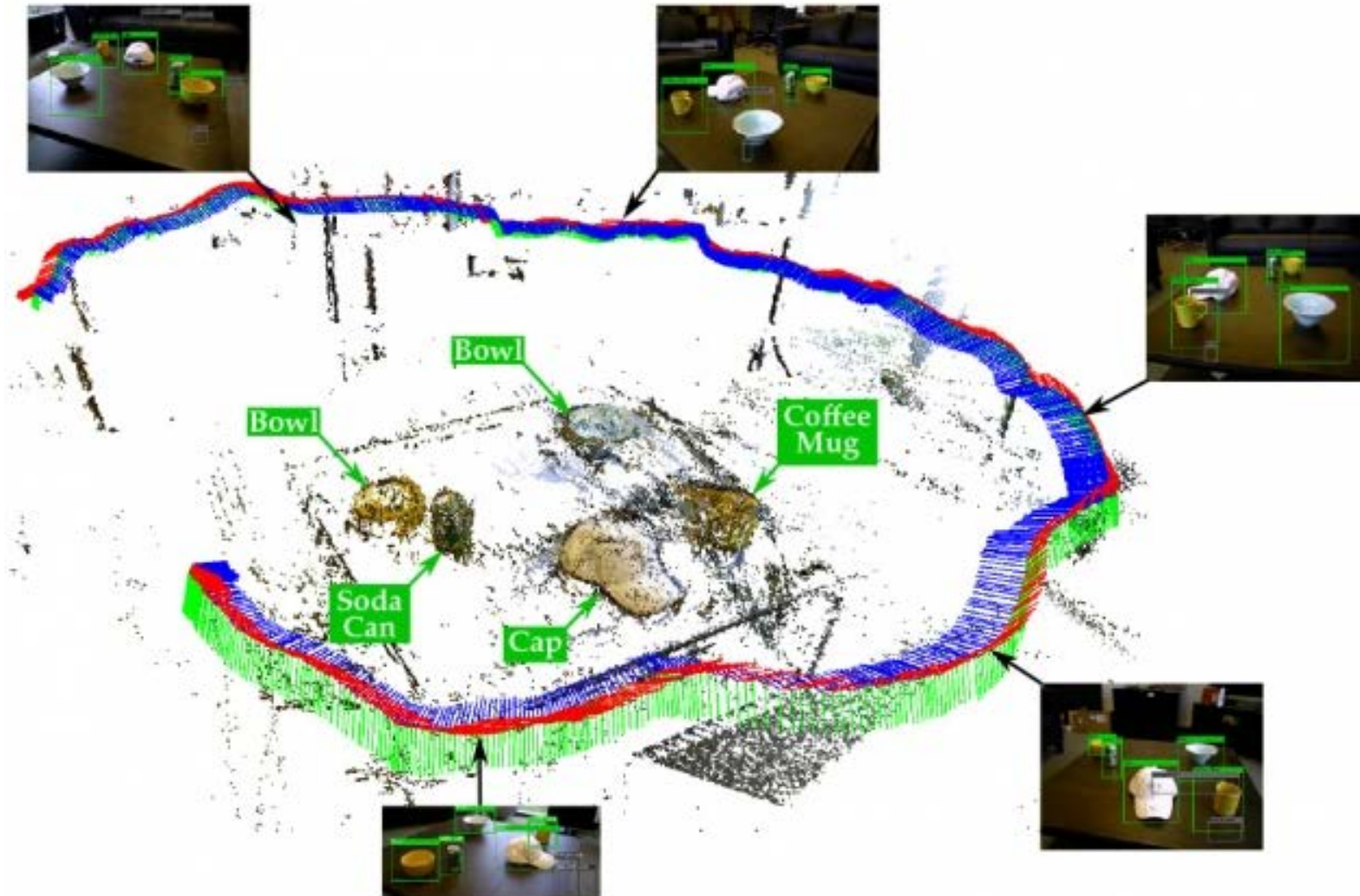
$$\mathbf{x}_r^T [\mathbf{t}]_{\times} \mathbf{R} \mathbf{x}_l = 0$$

$$\mathbf{E} = [\mathbf{t}]_{\times} \mathbf{R} \mathbf{x}_l \text{ とおくと}$$

$$\mathbf{x}_r^T \mathbf{E} \mathbf{x}_l = 0$$

v-SLAMで不可欠なLoop Closing

同じ場所を観測している2画像を検知するとループを閉じるよう最適化



<http://news.mit.edu/2015/object-recognition-robots-0724>

v-SLAMの代表的手法

Indirect Method (特徴点ベース)

MonoSLAM (Davison07)

- ・ 拡張カルマンフィルタをベース
- ・ 計算時間大(共分散行列が巨大)

PTAM (Klein07)

- ・ SfMをリアルタイムで実施
- ・ 全フレームをマップに反映することを諦める
- ・ 自己位置推定：毎フレーム
- ・ マッピング：視点が大きく変わるフレームのみ

ORB-SLAM (Mur-Artal15)

- ・ 局所的にキーフレームを多数取り込みあとから間引く適者生存戦略

Direct Method (画素ベース)

DTAM (Newcombe11)

- ・ フレーム内のすべてのピクセルの輝度についてフレーム対応を求める
- ・ photometric errorを考慮

LSD-STAM (Engel14)

- ・ 特徴点の利用はloop close検出のみ
- ・ マップ作成と自己位置推定(Tracking)を並列して別スレッド
- ・ 全体の整合性を保つためのマップ全体最適化も別スレッド

DSO (Engel16)

- ・ 特徴点の代わりに（推定に寄与する）勾配の大きい画素のみでTracking
- ・ 評価尺度は輝度値の整合性