

3D Computer Vision

Radim Šára Martin Matoušek

Center for Machine Perception
Department of Cybernetics
Faculty of Electrical Engineering
Czech Technical University in Prague

<https://cw.fel.cvut.cz/wiki/courses/tdv/start>

<http://cmp.felk.cvut.cz>

<mailto:sara@cmp.felk.cvut.cz>

phone ext. 7203

rev. December 7, 2021

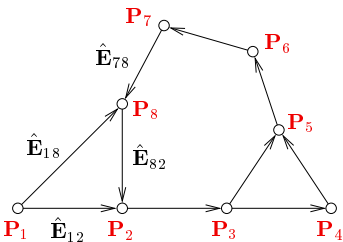


Open Informatics Master's Course

► Reconstructing Camera System from Pairs (Correspondence-Free)

Problem: Given a set of p decomposed pairwise essential matrices $\hat{\mathbf{E}}_{ij} = [\hat{\mathbf{t}}_{ij}]_{\times} \hat{\mathbf{R}}_{ij}$ and calibration matrices \mathbf{K}_i reconstruct the camera system $\mathbf{P}_i, i = 1, \dots, k$

→81 and →151 on representing \mathbf{E}



We construct calibrated camera pairs $\hat{\mathbf{P}}_{ij} \in \mathbb{R}^{6,4}$ see (17)

$$\hat{\mathbf{P}}_{ij} = \begin{bmatrix} \mathbf{K}_i^{-1} \hat{\mathbf{P}}_i \\ \mathbf{K}_j^{-1} \hat{\mathbf{P}}_j \end{bmatrix} = \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \hat{\mathbf{R}}_{ij} & \hat{\mathbf{t}}_{ij} \end{bmatrix} \in \mathbb{R}^{6,4}$$

- singletons i, j correspond to graph nodes k nodes
- pairs ij correspond to graph edges p edges

$\hat{\mathbf{P}}_{ij}$ are in different coordinate systems but these are related by similarities $\hat{\mathbf{P}}_{ij} \mathbf{H}_{ij} = \mathbf{P}_{ij}$

$$\begin{array}{c} R_{ij} | t_{ij} \\ \hline R_{ij} R_{ij} \end{array} \quad \underbrace{\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \hat{\mathbf{R}}_{ij} & \hat{\mathbf{t}}_{ij} \end{bmatrix}}_{\mathbb{R}^{6,4}} \underbrace{\begin{bmatrix} \mathbf{R}_{ij} & \mathbf{t}_{ij} \\ \mathbf{0}^{\top} & s_{ij} \end{bmatrix}}_{\mathbf{H}_{ij} \in \mathbb{R}^{4,4}} \stackrel{!}{=} \underbrace{\begin{bmatrix} \mathbf{R}_i & \mathbf{t}_i \\ \mathbf{R}_j & \mathbf{t}_j \end{bmatrix}}_{\mathbb{R}^{6,4}} \quad (28)$$

- (28) is a linear system of $24p$ eqs. in $7p + 6k$ unknowns $7p \sim (\mathbf{t}_{ij}, \mathbf{R}_{ij}, s_{ij}), 6k \sim (\mathbf{R}_i, \mathbf{t}_i)$
- each $\hat{\mathbf{P}}_i = (\mathbf{R}_i, \mathbf{t}_i)$ appears on the RHS as many times as is the degree of node \mathbf{P}_i eg. $\mathbf{P}_5 \ 3 \times$

► cont'd

Eq. (28) implies
$$\begin{bmatrix} \mathbf{R}_{ij} \\ \hat{\mathbf{R}}_{ij} \mathbf{R}_{ij} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_i \\ \mathbf{R}_j \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} \mathbf{t}_{ij} \\ \hat{\mathbf{R}}_{ij} \mathbf{t}_{ij} + s_{ij} \hat{\mathbf{t}}_{ij} \end{bmatrix} = \begin{bmatrix} \mathbf{t}_i \\ \mathbf{t}_j \end{bmatrix}$$

- \mathbf{R}_{ij} and \mathbf{t}_{ij} can be eliminated:

$$\hat{\mathbf{R}}_{ij} \mathbf{R}_i = \mathbf{R}_j, \quad \hat{\mathbf{R}}_{ij} \mathbf{t}_i + s_{ij} \hat{\mathbf{t}}_{ij} = \mathbf{t}_j, \quad s_{ij} > 0 \quad (29)$$

- note transformations that do not change these equations assuming no error in $\hat{\mathbf{R}}_{ij}$
 1. $\mathbf{R}_i \mapsto \mathbf{R}_i \mathbf{R}$,
 2. $\mathbf{t}_i \mapsto \sigma \mathbf{t}_i$ and $s_{ij} \mapsto \sigma s_{ij}$,
 3. $\mathbf{t}_i \mapsto \mathbf{t}_i + \mathbf{R}_i \mathbf{t}$

- the global frame is fixed, e.g. by selecting

$$\mathbf{R}_1 = \mathbf{I}, \quad \sum_{i=1}^k \mathbf{t}_i = \mathbf{0}, \quad \frac{1}{p} \sum_{i,j} s_{ij} = 1 \quad (30)$$

- rotation equations are decoupled from translation equations
- in principle, s_{ij} could correct the sign of $\hat{\mathbf{t}}_{ij}$ from essential matrix decomposition $\rightarrow 81$
but \mathbf{R}_i cannot correct the α sign in $\hat{\mathbf{R}}_{ij}$
 \Rightarrow therefore make sure all points are in front of cameras and constrain $s_{ij} > 0$; $\rightarrow 83$

+ pairwise correspondences are sufficient

- suitable for well-distributed cameras only (dome-like configurations)

otherwise intractable or numerically unstable

Finding The Rotation Component in Eq. (29)

1. Poor Man's Algorithm:

- create a Minimum Spanning Tree of \mathcal{G} from $\rightarrow 133$
- propagate rotations from $\mathbf{R}_1 = \mathbf{I}$ via $\hat{\mathbf{R}}_{ij}\mathbf{R}_i = \mathbf{R}_j$ from (29)

quality = ep1 error

2. Rich Man's Algorithm:

Consider $\hat{\mathbf{R}}_{ij}\mathbf{R}_i = \mathbf{R}_j$, $(i, j) \in E(\mathcal{G})$, where \mathbf{R} are a 3×3 rotation matrices
Errors per columns $c = 1, 2, 3$ of \mathbf{R}_j :

$$\mathbf{e}_{ij}^c = \hat{\mathbf{R}}_{ij}\mathbf{r}_i^c - \mathbf{r}_j^c, \quad \text{for all } i, j$$

Solve

$$\arg \min \sum_{(i,j) \in E(\mathcal{G})} \sum_{c=1}^3 (\mathbf{e}_{ij}^c)^\top \mathbf{e}_{ij}^c \quad \text{s.t.} \quad (\mathbf{r}_i^k)^\top (\mathbf{r}_j^l) = \begin{cases} 1 & i = j \wedge k = l \\ 0 & i \neq j \wedge k = l \\ 0 & i = j \wedge k \neq l \end{cases}$$

this is a quadratic programming problem

3. SVD-Lover's Algorithm:

Ignore the constraints and project the solution onto rotation matrices

[see next](#)

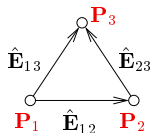
SVD Algorithm (cont'd)

Per columns $c = 1, 2, 3$ of \mathbf{R}_j :

$$\hat{\mathbf{R}}_{ij} \mathbf{r}_i^c - \mathbf{r}_j^c = \mathbf{0}, \quad \text{for all } i, j \quad (31)$$

- fix c and denote $\mathbf{r}^c = [\mathbf{r}_1^c, \mathbf{r}_2^c, \dots, \mathbf{r}_k^c]^\top$ c -th columns of all rotation matrices stacked; $\mathbf{r}^c \in \mathbb{R}^{3k}$
- then (31) becomes $\mathbf{D} \mathbf{r}^c = \mathbf{0}$ $\mathbf{D} \in \mathbb{R}^{3p, 3k}$
- $3p$ equations for $3k$ unknowns $\rightarrow p \geq k$ in a 1-connected graph we have to fix $\mathbf{r}_1^c = [1, 0, 0]$

Ex: ($k = p = 3$)



$$\begin{aligned} \hat{\mathbf{R}}_{12} \mathbf{r}_1^c - \mathbf{r}_2^c &= \mathbf{0} \\ \hat{\mathbf{R}}_{23} \mathbf{r}_2^c - \mathbf{r}_3^c &= \mathbf{0} \\ \hat{\mathbf{R}}_{13} \mathbf{r}_1^c - \mathbf{r}_3^c &= \mathbf{0} \end{aligned}$$

$$\mathbf{D} \mathbf{r}^c = \begin{bmatrix} \hat{\mathbf{R}}_{12} & -\mathbf{I} & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{R}}_{23} & -\mathbf{I} \\ \hat{\mathbf{R}}_{13} & \mathbf{0} & -\mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{r}_1^c \\ \mathbf{r}_2^c \\ \mathbf{r}_3^c \end{bmatrix} = \mathbf{0}$$

$$\mathbf{R}_i = [\mathbf{r}_1^1, \mathbf{r}_2^1, \mathbf{r}_3^1]$$

- must hold for any c

Idea:

[Martinec & Pajdla CVPR 2007]

1. find the space of all $\mathbf{r}^c \in \mathbb{R}^{3k}$ that solve (31) \mathbf{D} is sparse, use $[V, E] = \text{eigs}(\mathbf{D}^* \mathbf{D}, 3, 0)$; (Matlab)
 2. choose 3 unit orthogonal vectors in this space 3 smallest eigenvectors
 3. find closest rotation matrices per cam. using SVD because $\|\mathbf{r}^c\| = 1$ is necessary but insufficient
- global world rotation is arbitrary $\mathbf{R}_i^* = \mathbf{U}\mathbf{V}^\top$, where $\mathbf{R}_i = \mathbf{U}\mathbf{D}\mathbf{V}^\top$

correct def (\mathbf{R}_i^*)

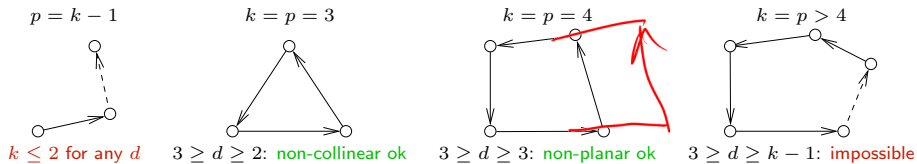
Finding The Translation Component in Eq. (29)

From (29) and (30): $0 < d \leq 3$ rank of camera center set, p - #pairs, k - #cameras

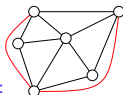
$$\hat{\mathbf{R}}_{ij} \mathbf{t}_i + s_{ij} \hat{\mathbf{t}}_{ij} - \mathbf{t}_j = \mathbf{0}, \quad \sum_{i=1}^k \mathbf{t}_i = \mathbf{0}, \quad \sum_{i,j} s_{ij} = p, \quad s_{ij} > 0, \quad \mathbf{t}_i \in \mathbb{R}^d$$

- in rank d : $d \cdot p + d + 1$ indep. eqns for $d \cdot k + p$ unknowns $\rightarrow p \geq \frac{d(k-1)-1}{d-1} \stackrel{\text{def}}{=} Q(d, k)$

Ex: Chains and circuits construction from sticks of known orientation and unknown length?



- equations insufficient for chains, trees, or when $d = 1$ collinear cameras
- 3-connectivity implies sufficient equations for $d = 3$ cams. in general pos. in 3D
 - s -connected graph has $p \geq \lceil \frac{sk}{2} \rceil$ edges for $s \geq 2$, hence $p \geq \lceil \frac{3k}{2} \rceil \geq Q(3, k) = \frac{3k}{2} - 2$
- 4-connectivity implies sufficient eqns. for any k when $d = 2$ coplanar cams
 - since $p \geq \lceil 2k \rceil \geq Q(2, k) = 2k - 3$
 - maximal planar triangulated graphs have $p = 3k - 6$ and give a solution for $k \geq 3$ maximal planar triangulated graph example:



Linear equations in (29) and (30) can be rewritten to

$$\mathbf{D}\mathbf{t} = \mathbf{0}, \quad \mathbf{t} = [\mathbf{t}_1^\top, \mathbf{t}_2^\top, \dots, \mathbf{t}_k^\top, s_{12}, \dots, s_{ij}, \dots]^\top$$

assuming measurement errors $\mathbf{D}\mathbf{t} = \boldsymbol{\epsilon}$ and $d = 3$, we have

$$\mathbf{t} \in \mathbb{R}^{3k+p}, \quad \mathbf{D} \in \mathbb{R}^{3p, 3k+p} \quad \text{sparse}$$

and

$$\mathbf{t}^* = \arg \min_{\mathbf{t}, s_{ij} > 0} \mathbf{t}^\top \mathbf{D}^\top \mathbf{D} \mathbf{t}$$

- this is a quadratic programming problem (mind the constraints!)

```
z = zeros(3*k+p,1);
```

```
t = quadprog(D.'*D, z, diag([zeros(3*k,1); -ones(p,1)]), z);
```

- but check the rank first!

► Bundle Adjustment

Goal: Use a good (and expensive) error model and improve all estimated parameters

Given:

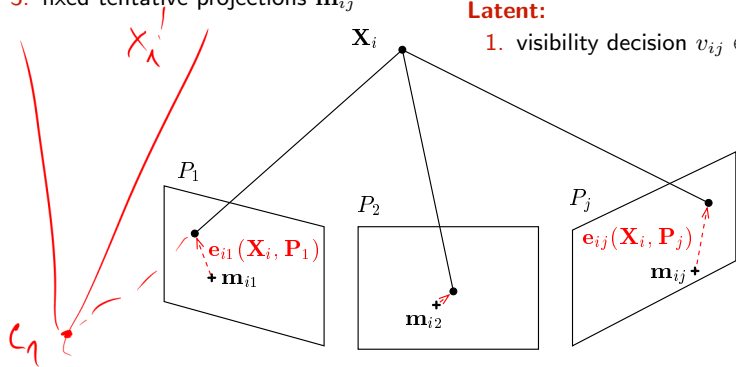
1. set of 3D points $\{\mathbf{X}_i\}_{i=1}^P$
2. set of cameras $\{\mathbf{P}_j\}_{j=1}^C$
3. fixed tentative projections \mathbf{m}_{ij}

Required:

1. corrected 3D points $\{\mathbf{X}'_i\}_{i=1}^P$
2. corrected cameras $\{\mathbf{P}'_j\}_{j=1}^C$

Latent:

1. visibility decision $v_{ij} \in \{0, 1\}$ per \mathbf{m}_{ij}



- for simplicity, \mathbf{X} , \mathbf{m} are considered Cartesian (not homogeneous)
- we have projection error $e_{ij}(\mathbf{X}_i, P_j) = \mathbf{x}_i - \mathbf{m}_i$ per image feature, where $\mathbf{x}_i = P_j \mathbf{X}_i$
- for simplicity, we will work with scalar error $e_{ij} = \|e_{ij}\|$

Robust Objective Function for Bundle Adjustment

The data model is constructed by marginalization over v_{ij} , as in the Robust Matching Model →114

$$p(\{e\} | \{\mathbf{P}, \mathbf{X}\}) = \prod_{\text{pts: } i=1}^p \prod_{\text{cams: } j=1}^c \left((1 - P_0) p_1(e_{ij} | \mathbf{X}_i, \mathbf{P}_j) + P_0 p_0(e_{ij} | \mathbf{X}_i, \mathbf{P}_j) \right)$$

marginalized negative log-density is (→115)

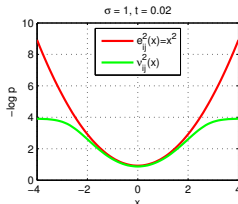
$$-\log p(\{e\} | \{\mathbf{P}, \mathbf{X}\}) = \sum_i \sum_j \underbrace{-\log \left(e^{-\frac{e_{ij}^2(\mathbf{X}_i, \mathbf{P}_j)}{2\sigma_1^2}} + t \right)}_{\rho(e_{ij}^2(\mathbf{X}_i, \mathbf{P}_j)) = \nu_{ij}^2(\mathbf{X}_i, \mathbf{P}_j)} \stackrel{\text{def}}{=} \sum_i \sum_j \nu_{ij}^2(\mathbf{X}_i, \mathbf{P}_j)$$

- we can use LM, e_{ij} is the projection error (not Sampson error)
- ν_{ij} is a 'robust' error fcn.; it is non-robust ($\nu_{ij} = e_{ij}$) when $t = 0$
- $\rho(\cdot)$ is a 'robustification function' often found in M-estimation
- the \mathbf{L}_{ij} in Levenberg-Marquardt changes to vector

$$(\mathbf{L}_{ij})_l = \frac{\partial \nu_{ij}}{\partial \theta_l} = \frac{1}{1 + t e^{\frac{e_{ij}^2(\theta)}{2\sigma_1^2}}} \cdot \frac{1}{\nu_{ij}(\theta)} \cdot \frac{1}{4\sigma_1^2} \cdot \frac{\partial e_{ij}^2(\theta)}{\partial \theta_l} \quad (32)$$

l-th ∂

small for $e_{ij} \gg \sigma_1$



but the LM method stays the same as before →108–109

- outliers (wrong v_{ij}): almost no impact on \mathbf{d}_s in normal equations because the red term in (32) scales contributions to both sums down for the particular ij

$$-\sum_{i,j} \mathbf{L}_{ij}^\top \nu_{ij}(\theta^s) = \left(\sum_{i,j} \mathbf{L}_{ij}^\top \mathbf{L}_{ij} \right) \mathbf{d}_s$$

► Sparsity in Bundle Adjustment

We have $q = 3p + 11k$ parameters: $\theta = (\mathbf{X}_1, \mathbf{X}_2, \dots, \mathbf{X}_p; \mathbf{P}_1, \mathbf{P}_2, \dots, \mathbf{P}_k)$ points, cameras

We will use a multi-index $r = 1, \dots, z$, $z = p \cdot k$. Then r correspond to point-cam pairs (i, j)

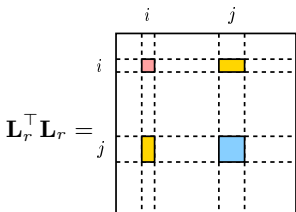
$$\theta^* = \arg \min_{\theta} \sum_{r=1}^z \nu_r^2(\theta), \quad \theta^{s+1} := \theta^s + \mathbf{d}_s, \quad \left(- \sum_{r=1}^z \mathbf{L}_r^\top \nu_r(\theta^s) = \left(\sum_{r=1}^z \mathbf{L}_r^\top \mathbf{L}_r + \lambda \text{diag}(\mathbf{L}_r^\top \mathbf{L}_r) \right) \mathbf{d}_s \right)$$

The block-form of \mathbf{L}_r in Levenberg-Marquardt ($\rightarrow 108$) is zero except in columns i and j :
 r -th error term is $\nu_r^2 = \rho(e_{ij}^2(\mathbf{X}_i, \mathbf{P}_j))$



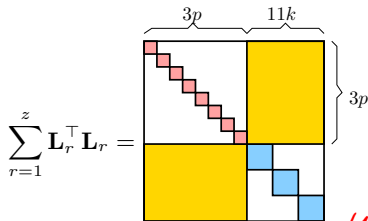
$r = (i, j)$ blocks:

- : $\mathbf{X}_i, 1 \times 3$
- : $\mathbf{P}_j, 1 \times 11$



blocks:

- : $\mathbf{X}_i - \mathbf{X}_i, 3 \times 3$
- : $\mathbf{X}_i - \mathbf{P}_j, 3 \times 11$
- : $\mathbf{P}_j - \mathbf{P}_j, 11 \times 11$



- "points-first-then-cameras" parameterization scheme

Thank You