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. October 17, 2023

Open Informatics Master's Course

►cont'd

	configuration	equation	# constraints
(3)	orthogonal vanishing points	$\underline{\mathbf{v}}_i^\top \boldsymbol{\omega}\, \underline{\mathbf{v}}_j = 0$	1
(4)	orthogonal vanishing lines	$\underline{\mathbf{n}}_{ij}^\top \boldsymbol{\omega}^{-1} \underline{\mathbf{n}}_{ik} = 0$	1
(5)	vanishing points orthogonal to vanishing lines	$\underline{\mathbf{n}}_{ij} = \varkappa\, \boldsymbol{\omega}\, \underline{\mathbf{v}}_k$	2
(6)	orthogonal image raster $\theta = \pi/2$	$\omega_{12} = \omega_{21} = 0$	1
(7)	unit aspect $a = 1$ when $\theta = \pi/2$	$\omega_{11} - \omega_{22} = 0$	1
(8)	known principal point $u_0 = v_0 = 0$	$\omega_{13} = \omega_{31} = \omega_{23} = \omega_{32} = 0$	2

ORUA

- These are homogeneous linear equations for the 5 parameters in $\boldsymbol{\omega}$ or $\boldsymbol{\omega}^{-1}$ — $\varkappa$ can be eliminated from (5)
- When $\mathbf{w} = \mathrm{vec}(\boldsymbol{\omega}) \in \mathbb{R}^6$, it has the form of $\mathbf{D}\mathbf{w} = \mathbf{0}$, $\mathbf{D} \in \mathbb{R}^{k \times 6}$
- With $k = 5$ constraints, we have $\mathrm{rank}(\mathbf{D}) = 5$, hence there is a unique solution for the homogeneous $\mathbf{w}$.
- We get $\mathbf{K}$ from $\boldsymbol{\omega}^{-1} = \mathbf{K}\mathbf{K}^\top$ by Choleski decomposition

the decomposition returns a positive definite upper triangular matrix

one avoids solving an explicit set of quadratic equations for the parameters in $\mathbf{K}$

Examples

Assuming orthogonal raster, unit aspect (ORUA): $\theta = \pi/2$, $a = 1$

$$\boldsymbol{\omega} \simeq \begin{bmatrix} 1 & 0 & -u_0 \\ 0 & 1 & -v_0 \\ -u_0 & -v_0 & f^2 + u_0^2 + v_0^2 \end{bmatrix}$$

Ex 1:

Assuming ORUA and known $m_0 = (u_0, v_0)$, two finite orthogonal vanishing points give f

$$\underline{\mathbf{v}}_1^\top \boldsymbol{\omega}\, \underline{\mathbf{v}}_2 = 0 \quad \Rightarrow \quad f^2 = \left| (\mathbf{v}_1 - \mathbf{m}_0)^\top (\mathbf{v}_2 - \mathbf{m}_0) \right|$$

in this formula, $\mathbf{v}_{1,2}$, $\mathbf{m}_0$ are Cartesian (not homogeneous)!

Ex 2:

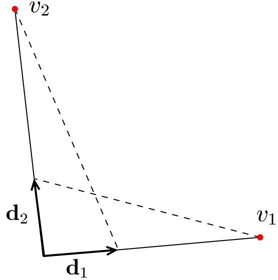
Non-orthogonal vanishing points $\mathbf{v}_i$, $\mathbf{v}_j$, known angle ϕ: $\cos\phi = \dfrac{\underline{\mathbf{v}}_i^\top \boldsymbol{\omega} \underline{\mathbf{v}}_j}{\sqrt{\underline{\mathbf{v}}_i^\top \boldsymbol{\omega} \underline{\mathbf{v}}_i}\sqrt{\underline{\mathbf{v}}_j^\top \boldsymbol{\omega} \underline{\mathbf{v}}_j}}$

- leads to polynomial equations
- e.g. ORUA and $u_0 = v_0 = 0$ gives

$$(f^2 + \mathbf{v}_i^\top \mathbf{v}_j)^2 = (f^2 + \|\mathbf{v}_i\|^2) \cdot (f^2 + \|\mathbf{v}_j\|^2) \cdot \cos^2\phi$$

▶Camera Orientation from Two Finite Vanishing Points

Problem: Given $\mathbf{K}$ and two vanishing points corresponding to two known orthogonal directions $\mathbf{d}_1$, $\mathbf{d}_2$, compute camera orientation $\mathbf{R}$ with respect to the plane.



- 3D coordinate system choice, e.g.:

$$\mathbf{d}_1 = (1,0,0), \quad \mathbf{d}_2 = (0,1,0)$$

- we know that

$$\mathbf{d}_i \simeq \mathbf{Q}^{-1}\underline{\mathbf{v}}_i = (\mathbf{K}\mathbf{R})^{-1}\underline{\mathbf{v}}_i = \mathbf{R}^{-1}\underbrace{\mathbf{K}^{-1}\underline{\mathbf{v}}_i}_{\underline{\mathbf{w}}_i}$$

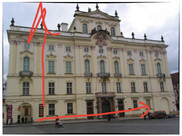
$$\mathbf{R}\mathbf{d}_i \simeq \underline{\mathbf{w}}_i \qquad i \in \{1,2\}$$

- knowing $\mathbf{d}_{1,2}$ we conclude that $\underline{\mathbf{w}}_i/\|\underline{\mathbf{w}}_i\|$ is the i-th column $\mathbf{r}_i$ of $\mathbf{R}$
- the third column is orthogonal: $\mathbf{r}_3 \simeq \mathbf{r}_1 \times \mathbf{r}_2$

$$\mathbf{R} = \begin{bmatrix} \frac{\underline{\mathbf{w}}_1}{\|\underline{\mathbf{w}}_1\|} & \frac{\underline{\mathbf{w}}_2}{\|\underline{\mathbf{w}}_2\|} & \frac{\underline{\mathbf{w}}_1 \times \underline{\mathbf{w}}_2}{\|\underline{\mathbf{w}}_1 \times \underline{\mathbf{w}}_2\|} \end{bmatrix}$$

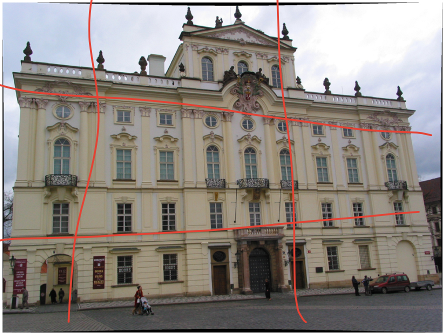
- we have to care about the signs $\pm\underline{\mathbf{w}}_i$ (such that $\det \mathbf{R} = 1$)

some suitable scenes



Application: Planar Rectification

Principle: Rotate camera (image plane) parallel to the plane of interest.



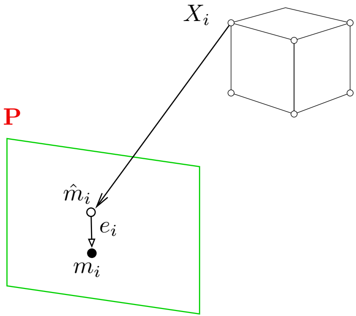
$$\underline{\mathbf{m}} \simeq \mathbf{K}\mathbf{R}\,[\mathbf{I} \quad -\mathbf{C}]\,\underline{\mathbf{X}} \qquad\qquad \underline{\mathbf{m}}' \simeq \mathbf{K}\,[\mathbf{I} \quad -\mathbf{C}]\,\underline{\mathbf{X}}$$

$$\underline{\mathbf{m}}' \simeq \mathbf{K}(\mathbf{K}\mathbf{R})^{-1}\,\underline{\mathbf{m}} = \mathbf{K}\mathbf{R}^{\top}\mathbf{K}^{-1}\,\underline{\mathbf{m}} = \mathbf{H}\,\underline{\mathbf{m}}$$

- $\mathbf{H}$ is the rectifying homography
- both $\mathbf{K}$ and $\mathbf{R}$ can be calibrated from two finite vanishing points — assuming ORUA →57
- not possible when one of them is (or both are) infinite
- without ORUA we would need 4 additional views to calibrate $\mathbf{K}$ as on →54

►Camera Resection

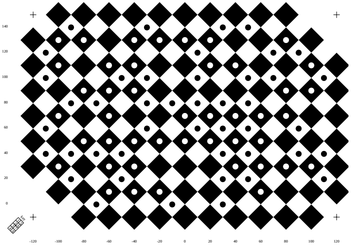
Camera <u>calibration</u> and <u>orientation</u> from a known set of $k \geq 6$ reference points and their images $\{(X_i, m_i)\}_{i=1}^{6}$.



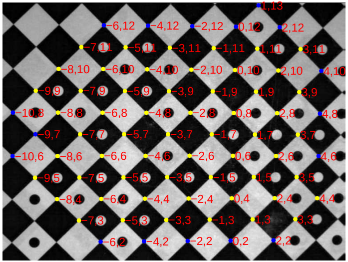
- X_i are considered exact
- m_i is a measurement subject to detection error

$$\mathbf{m}_i = \hat{\mathbf{m}}_i + \mathbf{e}_i \qquad \text{Cartesian}$$

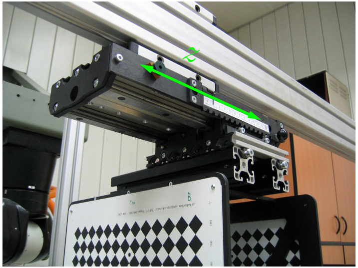
- where $\lambda_i \, \underline{\hat{\mathbf{m}}}_i = \mathbf{P}\underline{\mathbf{X}}_i$



calibration chart



automatic calibration point detection
based on a distributed bitcode ($2 \times 4 = 8$ bits)



resection target with translation stage

- target translated at least once
- by a calibrated (known) translation
- X_i point locations looked up in a table based on their bitcode

▶The Minimal Problem for Camera Resection

Problem: Given $k = 6$ corresponding pairs $\{(X_i, m_i)\}_{i=1}^{k}$, find $\mathbf{P}$

$$\lambda_i \underline{\mathbf{m}}_i = \mathbf{P}\underline{\mathbf{X}}_i, \qquad \mathbf{P} = \begin{bmatrix} \mathbf{q}_1^\top & q_{14} \\ \mathbf{q}_2^\top & q_{24} \\ \mathbf{q}_3^\top & q_{34} \end{bmatrix}$$

$$\underline{\mathbf{X}}_i = (x_i, y_i, z_i, 1), \quad i = 1, 2, \dots, k, \ k = 6$$

$$\underline{\mathbf{m}}_i = (u_i, v_i, 1), \quad \lambda_i \in \mathbb{R}, \ \lambda_i \neq 0, \ |\lambda_i| < \infty$$

easily modifiable for infinite points X_i but be aware of →64

expanded: $\lambda_i u_i = \mathbf{q}_1^\top \mathbf{X}_i + q_{14}, \quad \lambda_i v_i = \mathbf{q}_2^\top \mathbf{X}_i + q_{24}, \quad \lambda_i = \mathbf{q}_3^\top \mathbf{X}_i + q_{34}$

after elimination of λ_i: $(\mathbf{q}_3^\top \mathbf{X}_i + q_{34})u_i = \mathbf{q}_1^\top \mathbf{X}_i + q_{14}, \quad (\mathbf{q}_3^\top \mathbf{X}_i + q_{34})v_i = \mathbf{q}_2^\top \mathbf{X}_i + q_{24}$

Then

$$\mathbf{A}\,\mathbf{q} = \begin{bmatrix} \mathbf{X}_1^\top & 1 & \mathbf{0}^\top & 0 & -u_1\mathbf{X}_1^\top & -u_1 \\ \mathbf{0}^\top & 0 & \mathbf{X}_1^\top & 1 & -v_1\mathbf{X}_1^\top & -v_1 \\ \vdots & & & & & \vdots \\ \mathbf{X}_k^\top & 1 & \mathbf{0}^\top & 0 & -u_k\mathbf{X}_k^\top & -u_k \\ \mathbf{0}^\top & 0 & \mathbf{X}_k^\top & 1 & -v_k\mathbf{X}_k^\top & -v_k \end{bmatrix} \cdot \begin{bmatrix} \mathbf{q}_1 \\ q_{14} \\ \mathbf{q}_2 \\ q_{24} \\ \mathbf{q}_3 \\ q_{34} \end{bmatrix} = \mathbf{0} \qquad (9)$$

2k × 12

- we need 11 indepedent parameters for $\mathbf{P}$
- $\mathbf{A} \in \mathbb{R}^{2k,12}$, $\mathbf{q} \in \mathbb{R}^{12}$
- 6 points in a general position give $\operatorname{rank} \mathbf{A} = 12$ and there is no (non-trivial) null space
- drop one row to get rank-11 matrix, then the basis vector of the null space of $\mathbf{A}$ gives $\mathbf{q}$

▶The Jack-Knife Solution for $k = 6$

- given the 6 correspondences, we have 12 equations for the 11 parameters
- can we use all the information present in the 6 points?

Jack-knife estimation

1. $n := 0$
2. for $i = 1, 2, \ldots, 2k$ do
   a) delete i-th row from $\mathbf{A}$, this gives $\mathbf{A}_i$
   b) if $\dim \operatorname{null} \mathbf{A}_i > 1$ continue with the next i
   c) $n := n + 1$
   d) compute the right null-space $\mathbf{q}_i$ of $\mathbf{A}_i$ — e.g. by 'economy-size' SVD
   e) $\hat{\mathbf{q}}_i := \mathbf{q}_i$ normalized to $q_{34} = 1$ and dimension-reduced — assuming finite cam. with $P_{3,4} = 1$
3. from all n vectors $\hat{\mathbf{q}}_i$ collected in Step 2.e compute

$$\mathbf{q} = \frac{1}{n} \sum_{i=1}^{n} \hat{\mathbf{q}}_i, \qquad \operatorname{var}[\mathbf{q}] = \frac{n-1}{n} \operatorname{diag} \sum_{i=1}^{n} (\hat{\mathbf{q}}_i - \mathbf{q})(\hat{\mathbf{q}}_i - \mathbf{q})^\top$$

regular for $n \geq 11$
variance of the sample mean

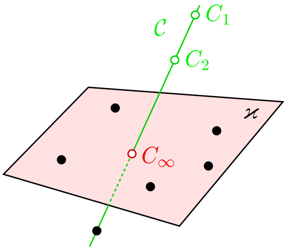
- have a solution + an error estimate, per individual elements of $\mathbf{P}$ (except P_{34})
- at least 5 points must be in a general position (→64)
- large error indicates near degeneracy
- computation not efficient with $k > 6$ points, needs $\binom{2k}{11}$ draws, e.g. $k = 7 \Rightarrow 364$ draws
- better error estimation method: decompose $\mathbf{P}_i$ to $\mathbf{K}_i$, $\mathbf{R}_i$, $\mathbf{t}_i$ (→33), represent $\mathbf{R}_i$ with 3 parameters (e.g. Euler angles, or in exponential map representation →144) and compute the errors for the parameters
- even better: use the $\mathrm{SE}(3)$ Lie group for $(\mathbf{R}_i, \mathbf{t}_i)$ and average its group-theoretic representations (the procedure is iterative)

▶Degenerate (Critical) Configurations for Camera Resection

Let $\mathcal{X} = \{X_i;\ i = 1, \ldots\}$ be a set of points and $\mathbf{P}_1 \not\simeq \mathbf{P}_j$ be two regular (rank-3) cameras. Then two configurations $(\mathbf{P}_1, \mathcal{X})$ and $(\mathbf{P}_j, \mathcal{X})$ are image-equivalent if

$$\mathbf{P}_1 \underline{\mathbf{X}}_i \simeq \mathbf{P}_j \underline{\mathbf{X}}_i \quad \text{for all} \quad X_i \in \mathcal{X}$$

i.e. there is a non-trivial set of other cameras that see the same image



Case 4

Results

- importantly: If all calibration points $X_i \in \mathcal{X}$ lie on a plane $\varkappa$ then camera resection is non-unique and all image-equivalent camera centers lie on a spatial line $\mathcal{C}$ with the $C_\infty = \varkappa \cap \mathcal{C}$ excluded

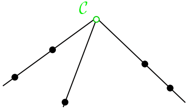
  this also means we cannot resect if all X_i are infinite
- and more: by adding points $X_i \in \mathcal{X}$ to $\mathcal{C}$ we gain nothing
- there are additional image-equivalent configurations, see next

Proof sketch: If $\mathbf{Q}$, $\mathbf{T}$ are suitable homographies then $\mathbf{P}_1 \simeq \mathbf{Q}\mathbf{P}_0\mathbf{T}$, where $\mathbf{P}_0$ is canonical and the analysis can be made with $\hat{\mathbf{P}}_j \simeq \mathbf{Q}^{-1}\mathbf{P}_j$

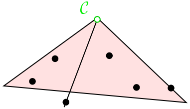
$$\mathbf{P}_0 \underbrace{\mathbf{T}\underline{\mathbf{X}}_i}_{\underline{\mathbf{Y}}_i} \simeq \hat{\mathbf{P}}_j \underbrace{\mathbf{T}\underline{\mathbf{X}}_i}_{\underline{\mathbf{Y}}_i} \quad \text{for all} \quad Y_i \in \mathcal{Y}$$

see [H&Z, Sec. 22.1.2] for a full prof

cont'd (all cases)

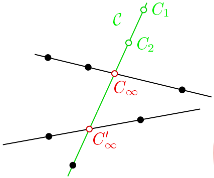


Case 5

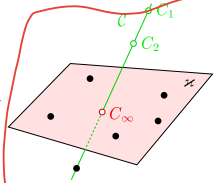


Case 6

- points lie on three optical rays or one optical ray and one optical plane
- cameras C_1, C_2 co-located at point $\mathcal{C}$
- Case 5: camera sees 3 isolated point images
- Case 6: cam. sees a line of points and an isolated point



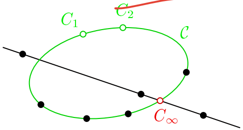
Case 3



Case 4

- points lie on a line $\mathcal{C}$ and
  1. on two lines meeting $\mathcal{C}$ at C_∞, C'_∞
  2. or on a plane meeting $\mathcal{C}$ at C_∞
- cameras lie on a line $\mathcal{C} \setminus \{C_\infty, C'_\infty\}$
- Case 3: camera sees 2 lines of points
- Case 4: **dangerous!**

Case 2

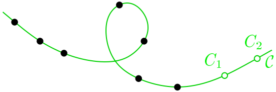


- points lie on a planar conic $\mathcal{C}$ and an additional line meeting $\mathcal{C}$ at C_∞
- cameras lie on $\mathcal{C} \setminus \{C_\infty\}$

not necessarily an ellipse

- Case 2: camera sees 2 lines of points

Case 1



- points and cameras all lie on a twisted cubic $\mathcal{C}$
- Case 1: camera sees points on a conic

dangerous but unlikely to occur

▸Three-Point Exterior Orientation Problem (P3P)

Calibrated camera rotation and translation from Perspective images of 3 reference Points.

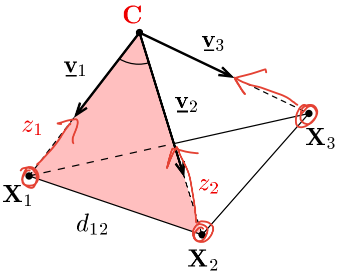
Problem: Given $\mathbf{K}$ and three corresponding pairs $\{(m_i, X_i)\}_{i=1}^{3}$, find $\mathbf{R}$, $\mathbf{C}$ by solving

$$\lambda_i \underline{\mathbf{m}}_i = \mathbf{K}\mathbf{R}\,(\mathbf{X}_i - \mathbf{C}), \qquad i = 1, 2, 3 \qquad \mathbf{X}_i \text{ Cartesian}$$

1. Transform $\underline{\mathbf{v}}_i \stackrel{\text{def}}{=} \mathbf{K}^{-1}\underline{\mathbf{m}}_i$. Then

$$\lambda_i \underline{\mathbf{v}}_i = \mathbf{R}\,(\mathbf{X}_i - \mathbf{C}). \tag{10}$$

2. If there was no rotation in (10), the situation would look like this



3. and we could shoot 3 lines from the given points $\mathbf{X}_i$ in given directions $\underline{\mathbf{v}}_i$ to get $\mathbf{C}$
4. given $\mathbf{C}$ we could solve (10) for λ_i

▶P3P cont'd

If there is rotation $\mathbf{R}$

1. Eliminate $\mathbf{R}$ by taking   rotation preserves length: $\|\mathbf{R}\mathbf{x}\| = \|\mathbf{x}\|$

$$|\lambda_i| \cdot \|\underline{\mathbf{v}}_i\| = \|\mathbf{X}_i - \mathbf{C}\| \overset{\text{def}}{=} z_i \qquad (11)$$

2. Consider only angles among $\underline{\mathbf{v}}_i$ and apply the Cosine Law per triangle $(\mathbf{C}, \mathbf{X}_i, \mathbf{X}_j)$ $i, j = 1, 2, 3,\ i \neq j$

$$d_{ij}^2 = z_i^2 + z_j^2 - 2\, z_i\, z_j\, c_{ij},$$

$$z_i = \|\mathbf{X}_i - \mathbf{C}\|,\ \ d_{ij} = \|\mathbf{X}_j - \mathbf{X}_i\|,\ \ c_{ij} = \cos(\angle \underline{\mathbf{v}}_i\, \underline{\mathbf{v}}_j)$$

4. Solve the system of 3 quadratic eqs in 3 unknowns z_i

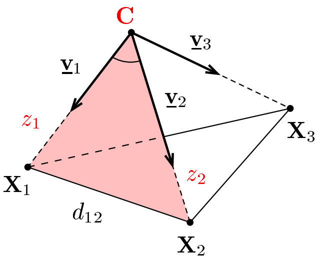
[Fischler & Bolles, 1981]

   there may be no real root

   there are up to 4 solutions that cannot be ignored   (verify on additional points)

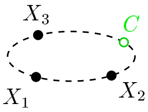
5. Compute $\mathbf{C}$ by trilateration (3-sphere intersection) from $\mathbf{X}_i$ and z_i; then λ_i from (11)

6. Compute $\mathbf{R}$ from (10)   we will solve this problem next →70



Similar problems (P4P with unknown f) at http://aag.ciirc.cvut.cz/minimal/ (papers, code)

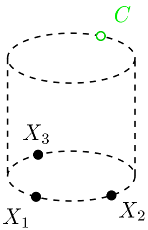
Degenerate (Critical) Configurations for Exterior Orientation



no solution

1. C cocyclic with (X_1, X_2, X_3) — camera sees points on a line

---



unstable solution

- center of projection C located on the orthogonal circular cylinder with base circumscribing the three points X_i

unstable: a small change of X_i results in a large change of C

can be detected by error propagation

degenerate

- camera C is coplanar with points (X_1, X_2, X_3) but is not on the circumscribed circle of (X_1, X_2, X_3) — camera sees points on a line

---

- additional critical configurations depend on the quadratic equations solver [Haralick et al. IJCV 1994]

►Populating A Little ZOO of Minimal Geometric Problems in CV

problem	given	unknown	slide
camera resection	6 world–image correspondences $\left\{(X_i, m_i)\right\}_{i=1}^{6}$	$\mathbf{P}$	→62
exterior orientation P3P	$\mathbf{K}$, 3 world–image correspondences $\left\{(X_i, m_i)\right\}_{i=1}^{3}$	$\mathbf{R}$, $\mathbf{C}$	→66
next: relative orientation	3 world-world correspondences $\left\{(X_i, Y_i)\right\}_{i=1}^{3}$	$\mathbf{R}$, $\mathbf{t}$	→70

- camera resection and exterior orientation are similar problems in a sense:
  - we do resectioning when our camera is uncalibrated
  - we do orientation when our camera is calibrated
- relative orientation involves no camera (see next) — it is a recurring problem in 3D vision
- more problems to come

▶The Relative Orientation Problem

Problem: Given point triples (X_1, X_2, X_3) and (Y_1, Y_2, Y_3) in a general position in $\mathbf{R}^3$ such that the correspondence $X_i \leftrightarrow Y_i$ is known, determine the relative orientation $(\mathbf{R}, \mathbf{t})$ that maps $\mathbf{X}_i$ to $\mathbf{Y}_i$, i.e.

$$\mathbf{Y}_i = \mathbf{R}\mathbf{X}_i + \mathbf{t}, \quad i = 1, 2, 3\,.$$

Applies to:

$\lambda_i m_i = R(X_i - C) = R X_i + t$

- 3D scanners
- merging partial reconstructions from different viewpoints
- generalization of the last step of P3P

Obs: Let the centroid be $\bar{\mathbf{X}} = \frac{1}{3}\sum_i \mathbf{X}_i$ and analogically for $\bar{\mathbf{Y}}$. Then

$$\bar{\mathbf{Y}} = \mathbf{R}\bar{\mathbf{X}} + \mathbf{t}.$$

Therefore

$$\mathbf{Z}_i \overset{\text{def}}{=} (\mathbf{Y}_i - \bar{\mathbf{Y}}) = \mathbf{R}(\mathbf{X}_i - \bar{\mathbf{X}}) \overset{\text{def}}{=} \mathbf{R}\mathbf{W}_i$$

If all dot products are equal, $\mathbf{Z}_i^\top \mathbf{Z}_j = \mathbf{W}_i^\top \mathbf{W}_j$ for $i, j = 1, 2, 3$, we have

$$\mathbf{R}^* = \begin{bmatrix}\mathbf{W}_1 & \mathbf{W}_2 & \mathbf{W}_3\end{bmatrix}^{-1} \begin{bmatrix}\mathbf{Z}_1 & \mathbf{Z}_2 & \mathbf{Z}_3\end{bmatrix}$$

Poor man's solver:

- normalize $\mathbf{W}_i$, $\mathbf{Z}_i$ to unit length, use the above formula, and then find the closest rotation matrix
- but this is equivalent to a non-optimal objective — it ignores errors in vector lengths

An Optimal Algorithm for Relative Orientation

We setup a minimization problem

$$\mathbf{R}^* = \arg\min_{\mathbf{R}} \sum_{i=1}^{3} \|\mathbf{Z}_i - \mathbf{R}\mathbf{W}_i\|^2 \quad \text{s.t.} \quad \mathbf{R}^\top \mathbf{R} = \mathbf{I}, \quad \det \mathbf{R} = 1$$

$$\arg\min_{\mathbf{R}} \sum_i \|\mathbf{Z}_i - \mathbf{R}\mathbf{W}_i\|^2 = \arg\min_{\mathbf{R}} \sum_i \left(\|\mathbf{Z}_i\|^2 - 2\mathbf{Z}_i^\top \mathbf{R}\mathbf{W}_i + \|\mathbf{W}_i\|^2\right) = \cdots = \arg\max_{\mathbf{R}} \sum_i \mathbf{Z}_i^\top \mathbf{R}\mathbf{W}_i$$

Obs 1: Let $\mathbf{A} : \mathbf{B} = \sum_{i,j} a_{ij} b_{ij}$ be the dot-product (Frobenius inner product) over real matrices. Then

$$\mathbf{A} : \mathbf{B} = \mathbf{B} : \mathbf{A} = \operatorname{tr}(\mathbf{A}^\top \mathbf{B}) = \operatorname{vec}(\mathbf{A})^\top \operatorname{vec}(\mathbf{B}) = \mathbf{a} \cdot \mathbf{b}$$

Obs 2: (cyclic property for matrix trace)

$$\operatorname{tr}(\mathbf{ABC}) = \operatorname{tr}(\mathbf{CAB})$$

Obs 3: ($\mathbf{Z}_i$, $\mathbf{W}_i$ are vectors)

$$\mathbf{Z}_i^\top \mathbf{R}\mathbf{W}_i = \operatorname{tr}(\mathbf{Z}_i^\top \mathbf{R}\mathbf{W}_i) \overset{O2}{=} \operatorname{tr}(\mathbf{W}_i \mathbf{Z}_i^\top \mathbf{R}) \overset{O1}{=} (\mathbf{Z}_i \mathbf{W}_i^\top) : \mathbf{R} = \mathbf{R} : (\mathbf{Z}_i \mathbf{W}_i^\top)$$

- Then we can factor the $\mathbf{R}$ out of the sum

$$\sum_i \mathbf{Z}_i^\top \mathbf{R}\mathbf{W}_i = \mathbf{R} : \left(\sum_i \mathbf{Z}_i \mathbf{W}_i^\top\right) \overset{\text{def}}{=} \mathbf{R} : \mathbf{M}$$

- Consider the SVD of $\mathbf{M}$: $\mathbf{M} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$. Then

$$\mathbf{R} : \mathbf{M} = \mathbf{R} : (\mathbf{U}\mathbf{D}\mathbf{V}^\top) \overset{O1}{=} \operatorname{tr}(\mathbf{R}^\top \mathbf{U}\mathbf{D}\mathbf{V}^\top) \overset{O2}{=} \operatorname{tr}(\mathbf{V}^\top \mathbf{R}^\top \mathbf{U}\mathbf{D}) \overset{O1}{=} (\mathbf{U}^\top \mathbf{R}\mathbf{V}) : \mathbf{D} \rightarrow \max$$

$U^\top U V^\top V = I$

We are solving

$$\mathbf{R}^* = \arg\max_{\mathbf{R}} \sum_i \mathbf{Z}_i^\top \mathbf{R} \mathbf{W}_i = \arg\max_{\mathbf{R}} \left(\mathbf{U}^\top \mathbf{R} \mathbf{V}\right) : \mathbf{D}$$

A particular solution is found as follows:

- $\mathbf{U}^\top \mathbf{R} \mathbf{V}$ must be (1) orthogonal, and closest to: (2) diagonal and (3) positive definite $\mathbf{D}$
- Since $\mathbf{U}$, $\mathbf{V}$ are orthogonal matrices then the solution to the problem is among $\mathbf{R}^* = \mathbf{U}\mathbf{S}\mathbf{V}^\top$, where $\mathbf{S}$ is diagonal and orthogonal, i.e. one of

$$\pm\operatorname{diag}(1,1,1), \quad \pm\operatorname{diag}(1,-1,-1), \quad \pm\operatorname{diag}(-1,1,-1), \quad \pm\operatorname{diag}(-1,-1,1)$$

- $\mathbf{U}^\top \mathbf{V}$ is not necessarily positive definite
- We choose $\mathbf{S}$ so that $(\mathbf{R}^*)^\top \mathbf{R}^* = \mathbf{I}$

Alg:

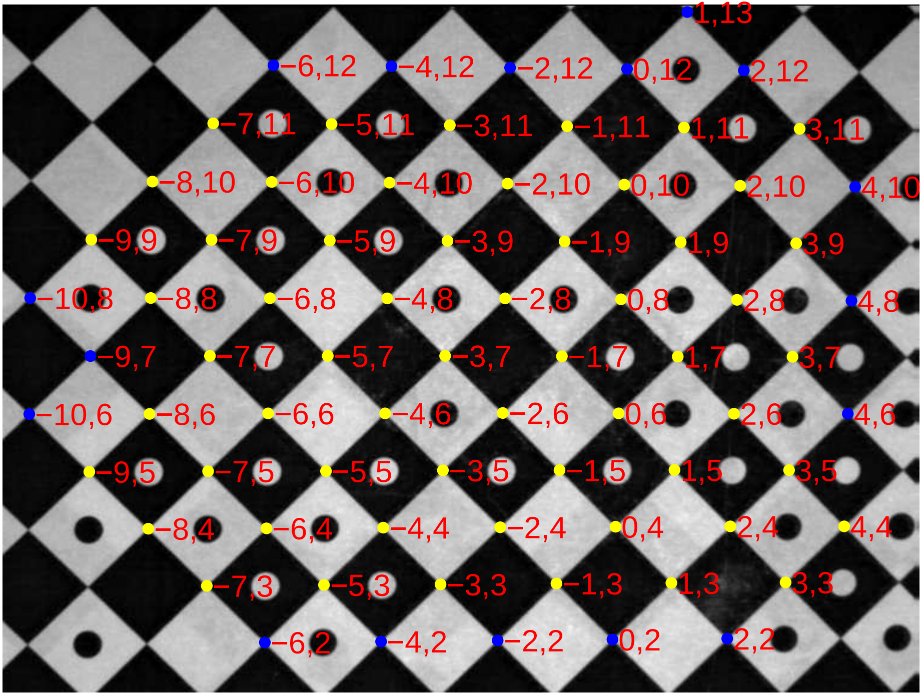
1. Compute matrix $\mathbf{M} = \sum_i \mathbf{Z}_i \mathbf{W}_i^\top$.
2. Compute SVD $\mathbf{M} = \mathbf{U}\mathbf{D}\mathbf{V}^\top$.
3. Compute all $\mathbf{R}_k = \mathbf{U}\mathbf{S}_k\mathbf{V}^\top$ that give $\mathbf{R}_k^\top \mathbf{R}_k = \mathbf{I}$.
4. Compute $\mathbf{t}_k = \bar{\mathbf{Y}} - \mathbf{R}_k \bar{\mathbf{X}}$.

- The algorithm can be used for more than 3 points
- Triple pairs can be pre-filtered based on motion invariants (lengths, angles)
- Can be used for the last step of the exterior orientation (P3P) problem →66

Thank You


CLEMENS·XII·PONT·MAX·ANN·SAC·MDCCXXXVII PONT·VIIII


BOOKS
FILMS
TICKETS
BOOKS


1,13
−6,12
−4,12
−2,12
0,12
2,12
−7,11
−5,11
−3,11
−1,11
1,11
3,11
−8,10
−6,10
−4,10
−2,10
0,10
2,10
4,10
−9,9
−7,9
−5,9
−3,9
−1,9
1,9
3,9
−10,8
−8,8
−6,8
−4,8
−2,8
0,8
2,8
4,8
−9,7
−7,7
−5,7
−3,7
−1,7
1,7
3,7
−10,6
−8,6
−6,6
−4,6
−2,6
0,6
2,6
4,6
−9,5
−7,5
−5,5
−3,5
−1,5
1,5
3,5
−8,4
−6,4
−4,4
−2,4
0,4
2,4
4,4
−7,3
−5,3
−3,3
−1,3
1,3
3,3
−6,2
−4,2
−2,2
0,2
2,2