A Minimal Solution for Non-Perspective Pose Estimation from Line Correspondences

ECCV'16 EUROPEAN CONFERENCE ON COMPUTER VISION

Gim Hee Lee (gimhee.lee@nus.edu.sg)
National University of Singapore

Objective

Given:

1. Three 2D-3D lines correspondences $l_{c_i} \leftrightarrow L_{W_i}, \forall i = 1,2,3$.
2. l_{c_i} are seen respectively by three cameras $F_{C_i}, \forall i = 1,2,3$ and L_{W_i} is defined in a fixed world frame F_W.
3. F_{C_i} are rigidly fixed together with known camera intrinsics K_i and extrinsics $T^G_{C_i}, \forall i = 1,2,3$.

Find: The pose of the multi-camera system with respect to the fixed world frame, i.e. relative transformation (R^W_G, t^W_G) that brings a point defined in the multi-camera reference frame F_G to the fixed world frame F_W.

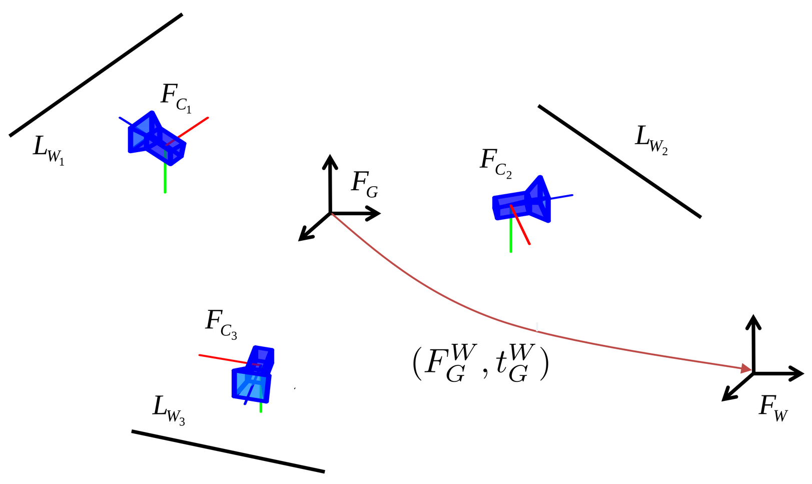


Fig.1. An illustration of the non-perspective pose estimation problem from line correspondences.

Note: In general, the minimal case for non-perspective pose estimation can be three 2D-3D line correspondences from a multi-camera system made up of either two or three cameras.

NP3L: 3-Line Minimal Solution

Solving for R^G_W

- Since $U^T V$ is zero for any Plücker line $[U^T \ V^T]^T$, u_C is parallel to U_C, and $R^C_W = R^C_G R^G_W$, from $L_C \leftrightarrow l_C$ we get
$$u_C^T R^C_G R^G_W V_W = 0. \tag{2}$$
- Rewrite Eq. (2) into $ar = 0$. a is a 1×9 matrix from known variables u_C^T, R^C_G and V_W, and r is a 9-vector from the *unknown* rotation matrix R^G_W.
- Given three 2D-3D line correspondences in the minimal case, we get $Ar = 0$ where A is a 3×9 matrix from $a_j, \forall j = 1,2,3$. Right null-space of A gives
$$r = \beta_1 b_1 + \beta_2 b_2 + \beta_3 b_3 + \beta_4 b_4 + \beta_5 b_5 + \beta_6 b_6. \tag{3}$$
- Setting $\beta_6 = 1$ and enforcing the orthogonality constraint on the rotation matrix formed by r, we get a system of 10 polynomial equations in terms of the unknowns $\beta_1 ... \beta_5$:
$$f_j(\beta_1, \beta_2, \beta_3, \beta_4, \beta_5) = 0, \quad j = 1,2...10, \tag{4}$$
which can be solved with Gröbner basis to get eight solutions.

Solving for t^G_W

- Since U_C and u_C are parallel, from $L_C \leftrightarrow l_C$ we get
$$\lambda u_C = \begin{pmatrix} R^C_G R^G_W & \lfloor R^C_G t^G_W + t^C_G \rfloor_\times R^C_G R^G_W \end{pmatrix} \begin{pmatrix} U_W \\ V_W \end{pmatrix}. \tag{5}$$
- Taking cross-product of u_C on both sides to get rid of the unknown scalar value λ and with three 2D-3D line correspondences, we get $Bt = 0$.
- Solution of the unknown $t = [t^G_{Wx} \ t^G_{Wy} \ t^G_{Wz} \ 1]^T$ is given by the right null-space of B which is made of known variables variables $R^C_G, t^C_G, R^G_W, u_C, U_W$ and V_W.

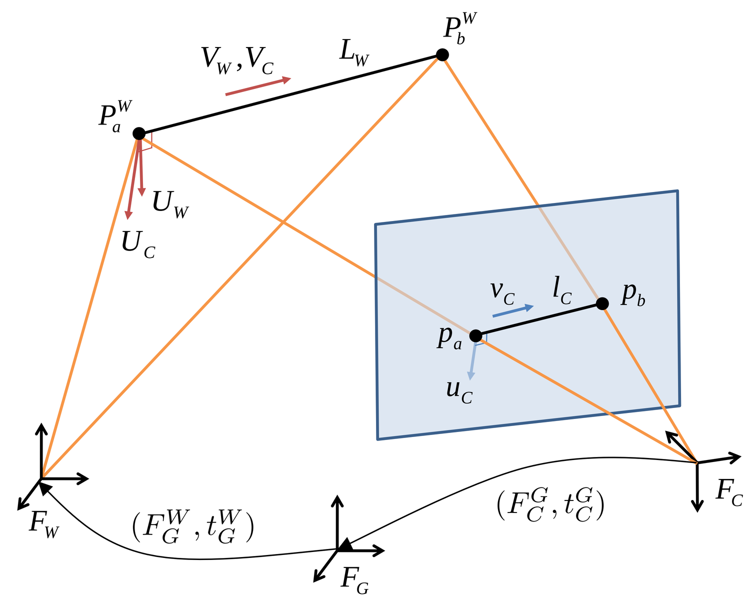
NPnL: ≥ 3 Line Correspondences

- For $n \geq 3$ 2D-3D line correspondences, A and B are $n \times 9$ and $2n \times 4$ matrices. The solution steps remain the same as the minimal case.

Special Cases

- **One Camera:** becomes the perspective pose estimation problem when all line correspondences are seen by only one camera. Camera extrinsics (R^C_G, t^C_G) vanishes and we directly solve for the camera pose (R^C_W, t^C_W) without any change to the algorithm.
- **Parallel 3D Lines:** minimal case where two or all the three 3D lines are parallel. Rank of matrix A drops below 3 and R^G_W cannot be solved. Fortunately, we can prevent this degenerate case by omitting parallel lines.

Plücker Line Representation

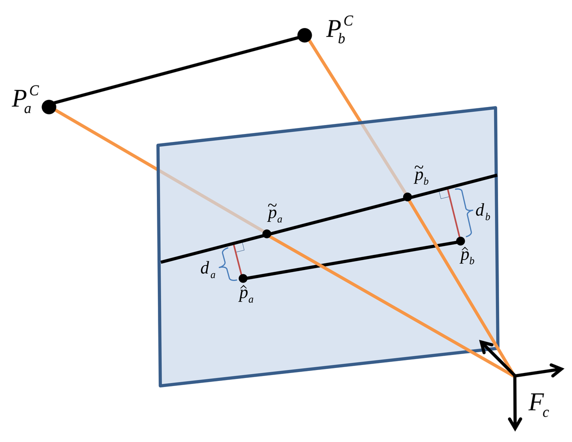


- 2D-3D line correspondence $l_C \leftrightarrow L_W$.
- P^W_a, P^W_b: end-points of 3D line L_W.
- p_a, p_b: end-points of 2D line l_C.

Fig.2. A 2D-3D line correspondence represented as Plücker lines.

- Plücker representation of a 3D line segment in the world frame is a 6-vector $L_W = [U_W^T \ \ V_W^T]^T$ computed from the 3D line end-points P^W_a, P^W_b.
- L_W can be expressed in the camera reference frame F_C as follows:
$$L_C = [U_C^T \ \ V_C^T]^T = \mathcal{T}^C_W L_W = \begin{pmatrix} R^C_W & \lfloor t^C_W \rfloor_\times R^C_W \\ 0_{3\times3} & R^C_W \end{pmatrix} L_W, \tag{1}$$
- $R^C_W = R^C_G R^G_W$ and $t^C_W = R^C_G t^G_W + t^C_G$, where $(R^C_G,\ t^C_G)$ is the known camera extrinsics and $(R^G_W,\ t^G_W)$ is the *unknown* pose of the multi-camera system.
- Plücker representation of the 2D line correspondence is a 6-vector $l_C = [u_C^T \ \ v_C^T]^T$ computed from the 2D line end-points p_a, p_b.

Reprojection Error for RANSAC



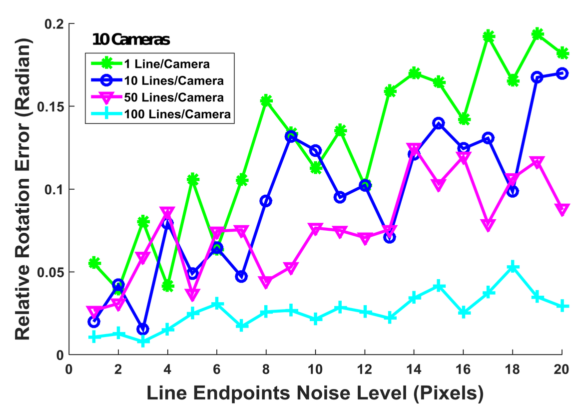
- P^C_a, P^C_b: end-points of 3D line transformed into camera frame.
- $\tilde{p}_a$, $\tilde{p}_b$: reprojection of P^C_a, P^C_b onto the image.
- $\hat{p}_a$, $\hat{p}_b$: camera matrix normalized 2D line end-points.

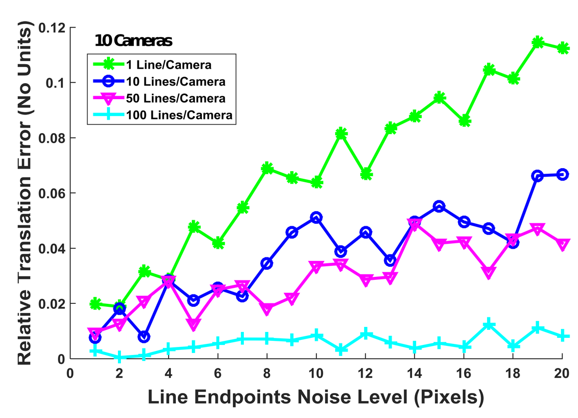
Reprojection Error: $$e = \frac{d_a + d_b}{2(\|\hat{p}_b - \hat{p}_a\|)}$$

Fig.3. An illustration of reprojection error from a 2D-3D line correspondence.

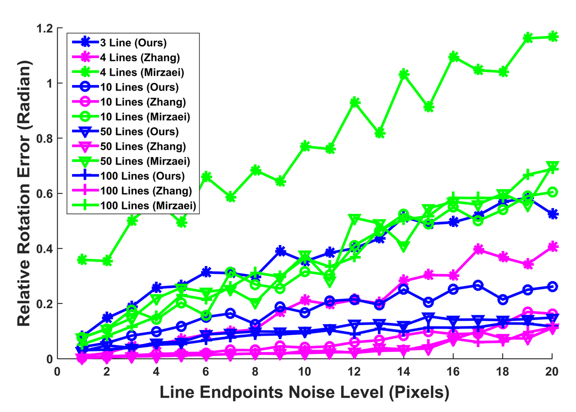
Results

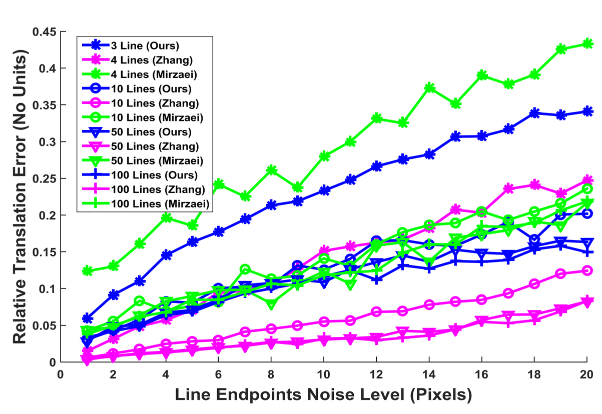
Simulations (Non-Perspective):





Simulations (Perspective):



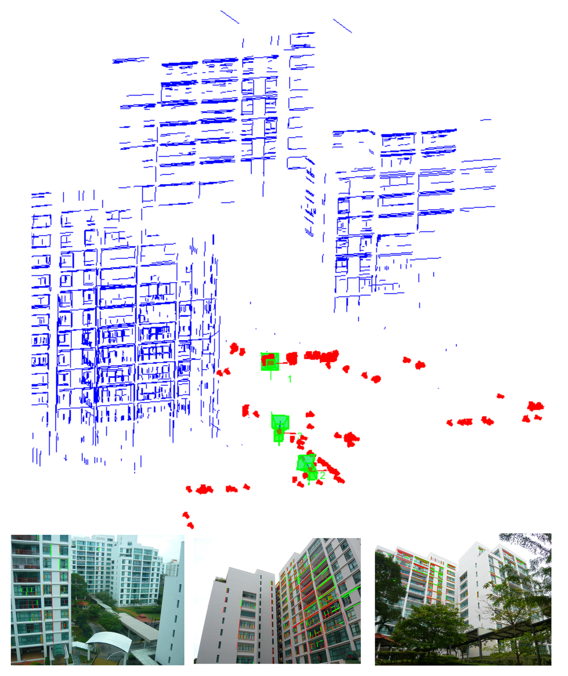


References

Zhang *et. al.*: Robust and efficient pose estimation from line correspondences, ACCV 2012.

Mirzaei *et. al.*: Globally optimal pose estimation from line correspondences, ICRA 2011.

Dataset 02



Real Data (Non-Perspective):

Table 1: Average errors from the multi-camera systems emulated with 3, 5, 10, 20 and 25 cameras from datasets 01 and 02 respectively.

Dataset	# of Cameras				
	3	5	10	20	25
01 (R Error)	0.0119	0.0051	0.0041	0.0032	0.0034
02 (R Error)	0.0562	0.1600	0.1381	0.2166	0.2778
01 (t Error)	0.0886	0.0465	0.0256	0.0289	0.0262
02 (t Error)	0.0380	0.0459	0.0700	0.0747	0.0900

Real Data (Perspective):

Table 2: Comparisons of errors from our algorithm, Zhang *et. al.* and Mirzaei *et. at.* for one camera.

Dataset	Zhang *et. al.*		Mirzaei *et. at.*		Ours	
	R Error	t Error	R Error	t Error	R Error	t Error
01	0.3016	0.1988	0.3733	0.2015	0.0155	0.0195
02	0.0698	0.0407	0.0186	0.0229	0.0369	0.2549