Orthographic-Perspective Epipolar Geometry
Supplementary Material

Viktor Larsson[1],   Marc Pollefeys[1,2],   Magnus Oskarsson[3]

[1] Department of Computer Science, ETH Zurich
[2] Microsoft Mixed Reality and AI Lab Zurich
[3] Centre for Mathematical Sciences, Lund University

1. Overview Supplementary Material

In the supplementary material we present the following;

- Comparison of using different internal constraints for building minimal solvers (Section 2)

- Additional discussion about related works (Section 3)

- Derivation of the generalized epipolar constraints using Plücker lines (Section 4)

- More details on the radar-camera calibration experiment (Section 5)

- Finally we consider two special cases; planar scene (Section 6) and known vertical direction (Section 7)

In Table 2 we also present more details results for the experiment on approximating large focal lengths as orthographic.

2. Impact of Internal Constraints

In this section we evaluate how the different internal constraints on the ortho-perspective essential matrix impact the minimal solvers. We compare the following equation sets

- **Orthogonality** constraints from Zhang et al.[29],
$$\mathbf{e}_1^T \mathbf{e}_1 - \mathbf{e}_2^T \mathbf{e}_2 = 0, \quad \mathbf{e}_1^T \mathbf{e}_2 = 0, \quad det(E) = 0 \quad (1)$$

- **Trace**-constraints proposed in the main paper,
$$2EE^T DE - \mathrm{tr}(EE^T D)E = 0, \quad det(E) = 0 \quad (2)$$

- Combining **trace**-constraints with the **orthogonality** constraints from Zhang et al. [29].

- Directly parameterizing the essential matrix using the **Cayley**-parameterization for the rotation
$$E = [(t_1, t_2, 1)]_\times DR \quad (3)$$

For each set of constraints we create a minimal solver using the generator from Larsson et al. [13]. The solvers are implemented in C++ and benchmarked on 1000 random synthetic instances. Table 1 show the solver statistics and the distribution of the residuals can be found in Figure 1.

Constraint	Template sz.	Solutions	Runtime	Residual
Ortho.	42×54	12	0.09 ms	-11.45
Trace	10×20	10	0.07 ms	-12.27
Trace + Ortho.	12×20	8	0.05 ms	-12.09
Cayley	389×405	16	4.97 ms	-8.85

Table 1. Statistics for minimal solvers created using different constraints. The table shows the average runtime and $\log_{10}$ equation residual for 1000 synthetic instances. Only by using the proposed trace constraints together with the orthogonality constraints from Zhang et al. [29] do we get the minimal number of solutions. Additionally the solver is slightly faster compared to the other approaches.
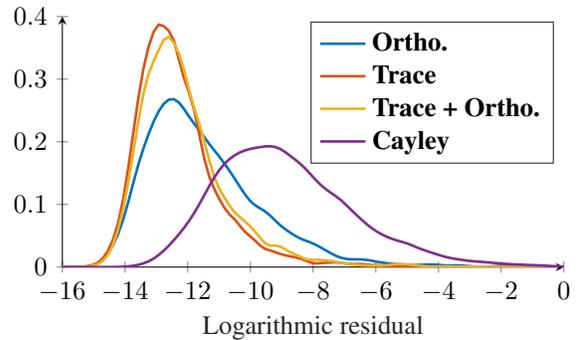
Figure 1. *Solver stability*. The figure shows the distribution of the $\log_{10}$ equation residuals for 1000 random synthetic instances.

3. Additional Discussion on Related Works

In this section we discuss additional related works; mainly related to the applications we show in the paper.

Aerial/Satellite. There have been several works which try to align 3D models with aerial or satellite images. Kaminsky et al. [11] align structure-from-motion models to overhead images. The method first extracts edges in the overhead image. Then it optimizes the alignment by promoting that the 3D points project onto edges in the ortho-image while enforcing free-space constraints obtained from the

rays between the 3D points and the cameras. In [16] Ni et al. proposed a method for aligning 3D-point clouds to an aerial image. They first estimate the up-direction in the point cloud, and then estimate the remaining parameters using an approach based on the Hough-transform. Another approach was proposed by [28] which instead aligns 3D maps to a satellite image by minimizing a cost derived from the Chamfer distance of the projections. Shan et al. [22] consider the problem of performing local feature matching between aerial and ground-level images. Their approach is based on using dense depth to synthesize images where viewpoints align with the aerial photo. For settings where only rough position estimates are available from GPS, Park et al. [18] propose to use matching between the query image and an aerial view to estimate camera viewing direction. See Gao et al. [8] for a recent survey on ground-to-aerial registration.

2D-Maps. There have also been several works which utilize 2D-maps (such as floor plans) for various 3D vision tasks. For example, Wijmans et al. [27] leverage 2D-maps for co-registering several panoramic RGB-D images. In [3] the authors propose to use floor plans for LiDAR-based localization. There are also several works which perform image-based localization against 2D-maps. For example, in [4] the authors localize by finding vertical edges of buildings in an omnidirectional image. These are then used to estimate the camera pose w.r.t. a 2D map. Chu et al. [5] present a method for refining a coarse initial position/orientation estimate given a 2D-map. Arth et al. [1, 2] propose a localization framework which instead uses a 2.5D map representation (i.e. 2D-map augmented with height information).

Radar calibration. Extrinsic relative calibration of combined camera and radar systems has been investigated in a number of previous works. The exact geometry relating the camera and radar frame is highly non-linear and therefore simplifications both to the feature extraction and to the radar model are often utilized. One can assume that the scene is planar, which leads to homography based methods, [24, 26]. The calibration is here done by solving for the homography in a least squares manner. Many methods use specialized calibration targets [19, 6] in order to simplify the feature matching. These methods are based on iteratively minimizing some cost function, and if the full radar model is used, as in [15], then additional properties of the 3D scene are assumed to be known apriori. Recently also learning approaches have been proposed, however often these require known calibration ground truth data for training [20]. To our knowledge the mixed orthographic-perspective model has not been used previously for calibrating radar-camera systems.

4. Derivation of Non-central Constraints using Plücker Coordinates

In the main paper we derived the constraints for non-central perspective projection (*i.e.* a generalized camera) in combination with orthographic projection. In [23] the constraints and solver for the pure perspective case were developed using the Plücker representations of the 3D viewing lines. For additional geometric understanding we will now show how a similar analysis can be done also in our case. For further details on Plücker represesentations and properties see *e.g.* [21]. A line in 3D can be parametrized using two vectors, q and q'. Here q represents the direction of the line, and q' is given by the cross product of any two points on the line (or by q and any point on the line). The representation is only determined up to scale and also fulfills a quadratic constraint, and hence has four degrees of freedom. Two 3D lines (q_1, q'_1) and (q_2, q'_2) will intersect if and only if

$$q_1^T q'_2 + q_2^T q'_1 = 0. \tag{4}$$

Now consider a correspondence between an orthographic camera and a non-central camera. Again let $\mathbf{x}_o$ denote the image point in the ortho-image and parameterize the viewing ray in the non-central cameras as $\lambda \mathbf{x}_p + \mathbf{c}_p$. The viewing ray in the perspective camera has Plucker coordinates $(\mathbf{x}_p, s(\mathbf{c}_p \times \mathbf{x}_p))$, where s is the unknown scale between the two coordinate systems. In the orthographic camera we have the viewing ray given by $((0,0,1)^T, \mathbf{x}_o \times (0,0,1)^T)$. In order to check if these two lines intersect we need to have them in the same coordinate system. To this end we transform the Plucker coordinates of the orthographic camera to the perspective coordinate system, giving $q_2 = R^T(0,0,1)^T$ and

$$q'_2 = (R^T(\mathbf{x}_o - \begin{bmatrix} t_1 \\ t_2 \\ 0 \end{bmatrix})) \times (R^T \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}) = \tag{5}$$

$$R^T(\begin{bmatrix} m_x - t_1 \\ my - t_2 \\ 1 \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}) = R^T \begin{bmatrix} m_y - t_2 \\ -(m_x - t_1) \\ 0 \end{bmatrix}. \tag{6}$$

Now inserting the expression for the two lines in (4) gives the following constraint

$$(R^T \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix})^T (s(\mathbf{c}_p \times \mathbf{x}_p)) + (R^T \begin{bmatrix} m_y - t_2 \\ -(m_x - t_1) \\ 0 \end{bmatrix})^T \mathbf{x}_p = \tag{7}$$

$$s \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}^T R(\mathbf{c}_p \times \mathbf{x}_p) + \begin{bmatrix} m_x \\ m_y \\ 1 \end{bmatrix}^T TR\mathbf{x}_p = \tag{8}$$

$$s\mathbf{c}_p^T [\mathbf{r}_3]_\times \mathbf{x}_p + \mathbf{x}_o^T E\mathbf{x}_p = 0. \tag{9}$$

f_o		OPE 5pt.	f_o+E 6pt. [12]	OPE+f_p 6pt.	F 7pt. [10]	E 5pt. [17]
150mm (N=10237)	AUC5	4.00	**26.29**	7.13	**21.60**	57.68
	AUC10	16.05	**44.52**	18.66	**44.13**	69.53
	AUC20	37.90	**65.68**	39.53	**66.81**	80.23
	F1	0.87	**0.92**	0.90	**0.93**	0.90
300mm (N=9391)	AUC5	10.35	**19.49**	10.70	**12.87**	54.88
	AUC10	27.13	**34.71**	25.36	**31.21**	67.05
	AUC20	49.07	**56.09**	47.51	**56.07**	76.88
	F1	0.91	**0.93**	0.93	**0.94**	0.92
600mm (N=3313)	AUC5	**20.39**	11.22	**15.16**	6.98	43.22
	AUC10	**37.78**	21.76	**29.20**	20.18	59.09
	AUC20	**56.13**	42.57	**48.66**	44.92	70.93
	F1	0.92	**0.93**	0.93	**0.94**	0.92

Table 2. **Ortho-approximation for large focal lengths.** The table shows different AUC for the rotation error in degrees and the F1 score of the inliers w.r.t. the ground truth inliers. Orthographic approximation works better for large focal lengths whereas the performance degrades for the solver which tries to estimate the focal length (OPE vs. f_o+E). The same trend occurs when the focal length in the *perspective* camera is unknown (OPE+f_p vs. F).
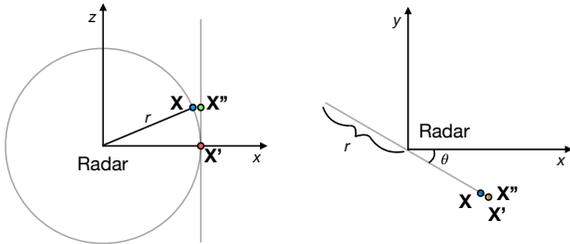
Figure 2. Schematic overview of the radar geometry, shown in the $xz-$plane (left) and the $xy-$plane (right). The 2D radar measures the distance r and the angle θ. The true unknown 3D position is given by **X**. The homography approximation is given by **X'** and the orthographic approximation by **X"**

5. Orthographic Radar Model

In the main paper we showed results on camera-radar calibration using the ortho-perspective essential matrix. We will here provide some more details and motivation on the geometric setup and the experiments. A schematic of the geometry for the 2D radar we consider is given in Fig. 2. The 2D radar measures the distance to a point in 3D, and also the angle to the point in the plane that the radar is swept in. This means that in the coordinate system of the radar, the 3D point is constrained to lie on a circle. The problem that we would like to address is to estimate the relative pose between a 2D radar and an ordinary camera, given a number of measured point correspondences, possibly containing a large amount of outliers due to mismatches. To solve this problem directly is very difficult, but we will show that our ortho-perspective epipolar geometry gives a very good approximation of the geometry for many practical cases. An approximation that simplifies the geometry

is to assume that all points lie in the plane that the radar is swept in. The benefit of this approximation is that one can directly compute it from the radar measurement, simply as $\mathbf{X}' = (r\cos\theta, r\sin\theta, 0)$ if the radar is swept in the plane $z = 0$. This approximation is valid if the height (i.e. offset from the radar plane) is small compared to the distance from the radar. Since this approximation directly gives us the 3D coordinate from the radar measurement, and all points lie in a plane, we can describe the mapping from the radar data to any perspective camera viewing the same points with a homography. If we now instead let the 3D point lie on the line $\mathbf{X}'' = (r\cos\theta, r\sin\theta, z)$ we get a much better approximation of the true position. The mapping from the 3D point to the radar data can now be described using an orthographic camera

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad (10)$$

so that

$$\begin{bmatrix} r\cos\theta \\ r\sin\theta \\ 1 \end{bmatrix} = P \begin{bmatrix} r\cos\theta \\ r\sin\theta \\ z \\ 1 \end{bmatrix}. \qquad (11)$$

For our calibration setup, persons walking were tracked automatically both in the radar and a calibrated camera, and the goal is to automatically find the relative pose between the camera and the radar. We model the radar using an orthographic camera and compare our results with using a homography based approximation of the scene. To test the repeatability and stability of the results, we collected two different datasets (*i.e.* two different point tracks), but where the camera-radar configuration was the same. The reprojection results can be seen in Figure 3, for the two datasets (top and bottom, respectively). Left and middle shows the reprojections in the projective camera and the radar using our orthographic model. Right shows the reprojection using a homography. The homography based method gives worse results especially in the areas where the planar scene model isn't valid. In Figure 4 the reconstruction results for the two sets (top and bottom, respectively) are given. One can see that we get consistent relative pose estimated for the two datasets.

6. Ortho-Perspective Planar Homography

In the case of a planar scene there exists a homography between the images. Similar to the perspective case (see e.g. [7, 30, 14]) this homography can be factorized into the relative pose and the plane-normal.

Assume that the plane is given by $\mathbf{n}^T\mathbf{X} = 1$ in the coordinate system of the perspective camera. The perspective depth can then be found as $\mathbf{X} = \lambda\mathbf{x}_p \implies \lambda = 1/\mathbf{n}^T\mathbf{x}_p$.

Projecting into the orthographic camera we get

$$\begin{pmatrix} m_x \\ m_y \end{pmatrix} = \frac{1}{\mathbf{n}^T \mathbf{x}_p} \begin{bmatrix} \mathbf{r}_1^T \\ \mathbf{r}_2^T \end{bmatrix} \mathbf{x}_p + \begin{pmatrix} t_1 \\ t_2 \end{pmatrix}. \tag{12}$$

Rewriting this reveals the structure of the homography

$$\mathbf{x}_o = \begin{pmatrix} m_x \\ m_y \\ 1 \end{pmatrix} \simeq \left(\begin{bmatrix} \mathbf{r}_1^T \\ \mathbf{r}_2^T \\ \mathbf{0}^T \end{bmatrix} + \begin{pmatrix} t_1 \\ t_2 \\ 1 \end{pmatrix} \mathbf{n}^T \right) \mathbf{x}_p. \tag{13}$$

Note however that it still has eight degrees of freedom. Therefore there does not exist any additional internal constraints and the matrix can be linearly estimated using DLT [9] from four point-correspondences.

Now given a homography H we can recover the factorization in (13). We start by noting that the third row directly gives the normal vector $\mathbf{n} = \mathbf{h}_3$. We can further parameterize the rotation in terms of the translation by

$$\mathbf{r}_1 = \mathbf{h}_1 - t_1 \mathbf{h}_3, \quad \mathbf{r}_2 = \mathbf{h}_2 - t_2 \mathbf{h}_3, \tag{14}$$

where $\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3$ are the rows of H. Since the homography is only estimated up to scale, this yields two polynomial constraints in $\mathbf{r}_1^T \mathbf{r}_2 = 0$, $\mathbf{r}_1^T \mathbf{r}_1 = \mathbf{r}_2^T \mathbf{r}_2$, which are quadratic in t_1 and t_2. Hiding the variable t_1 we can rewrite this as $A(t_1) \left(t_2^2, \ t_2, \ 1 \right)^T = 0$ where the matrix $A(t_1) =$

$$\begin{bmatrix} -\mathbf{n}^T \mathbf{n} & 2\mathbf{h}_2^T \mathbf{n} & \mathbf{h}_1^T \mathbf{h}_1 - 2t_1 \mathbf{h}_1^T \mathbf{n} + t_1^2 \mathbf{n}^T \mathbf{n} - \mathbf{h}_2^T \mathbf{h}_2 \\ 0 & t_1 \mathbf{n}^T \mathbf{n} - \mathbf{h}_1^T \mathbf{n} & \mathbf{h}_1^T \mathbf{h}_2 - t_1 \mathbf{h}_2^T \mathbf{n} \\ t_1 \mathbf{n}^T \mathbf{n} - \mathbf{h}_1^T \mathbf{n} & \mathbf{h}_1^T \mathbf{h}_2 - t_1 \mathbf{h}_2^T \mathbf{n} & 0 \end{bmatrix}.$$

The third row of A is simply $t_1 \mathbf{r}_1^T \mathbf{r}_2$. The determinant of A now yields a degree four univariate polynomial in t_1 which can be solved in closed form. For each solution of t_1 we can recover $t_2 = (t_1 \mathbf{h}_2^T \mathbf{n} - \mathbf{h}_1^T \mathbf{h}_2)/(t_1 \mathbf{n}^T \mathbf{n} - \mathbf{h}_1^T \mathbf{n})$.

7. Known Vertical Direction

In this section we consider the special case of known vertical direction. This can e.g. be from IMU devices or estimated from vanishing points in the perspective image. For aerial/satellite images the inclination angle is also often provided. Let $\mathbf{v}_p$ and $\mathbf{v}_o$ be the vertical direction in the coordinate systems of the perspective and orthographic camera, respectively. Further, let R_0 be any rotation which takes $\mathbf{v}_p$ onto $\mathbf{v}_o$, i.e. $\mathbf{v}_o = R_0 \mathbf{v}_p$. This then gives us the relative rotation up to an unknown rotation around $\mathbf{v}_o$, i.e.

$$R = R(\theta, \mathbf{v}_o) R_0, \tag{15}$$

where $R(\theta, \mathbf{v}_o)$ is a rotation around $\mathbf{v}_o$ with angle θ. For the perspective camera we can pre-rotate the coordinate system with R_0, i.e. replacing $\mathbf{x}_p$ with $R_0 \mathbf{x}_p$, which reduces the relative rotation to $R(\theta, \mathbf{v}_o)$.

7.1. Minimal Solver

The special structure of the rotation reduces the overall degrees of freedom to three instead of five. Thus we can minimally estimate the essential matrix from only three correspondences. We will now derive a minimal solver for this case. Note however that due to the similar structure, the derivations closely follow the perspective essential matrix solver with known vertical direction from Sweeney et al. [25].

Using the Weierstrass substitution $q = \tan(\theta/2)$, the rotation matrix (up to scale) can be written as

$$R(q) = (1 + q^2)I + 2q[\mathbf{v}_o]_\times + 2q^2[\mathbf{v}_o]_\times^2. \tag{16}$$

This allows us to parameterize the essential matrix with three parameters via

$$E(q, t_1, t_2) = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ -t_2 & t_1 & 0 \end{bmatrix} R(q). \tag{17}$$

The three epipolar constraints can then be rewritten as

$$\left(C_2 q^2 + C_1 q + C_0 \right) \begin{pmatrix} t_1 \\ t_2 \\ 1 \end{pmatrix} = 0, \quad C_i \in \mathbb{R}^{3 \times 3}. \tag{18}$$

This is a quadratic eigenvalue problem which can be solved using standard methods.

7.2. Overhead Orthographic Images

Interestingly the problem has a special structure when the vertical direction is aligned with the viewing direction of the orthographic camera, i.e. $\mathbf{v}_o = (0, 0, 1)^T$. In this case the relative rotation is around the z-axis, i.e.

$$R = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{19}$$

The essential matrix then has the form

$$E = \begin{bmatrix} -\sin\theta & -\cos\theta & 0 \\ \cos\theta & -\sin\theta & 0 \\ x & y & 0 \end{bmatrix}, \tag{20}$$

where $x = t_1 \sin\theta - t_2 \cos\theta$ and $y = t_1 \cos\theta + t_2 \sin\theta$. From the epipolar constraints we can write the problem as

$$A(\cos\theta, \sin\theta, x, y)^T = 0, \quad A \in \mathbb{R}^{3 \times 4}. \tag{21}$$

Finding the nullvector of A and rescaling such that the first two elements are a unit-vector yields the solution.

It turns out that the problem is geometrically equivalent to 1D absolute camera pose estimation. To see this algebraically, note that we can rewrite the epipolar constraint as

$$\mathbf{x}_o^T E \mathbf{x}_p = (-y, \ x) R_{2 \times 2}^T \left[\begin{pmatrix} m_x \\ m_y \end{pmatrix} - \begin{pmatrix} t_1 \\ t_2 \end{pmatrix} \right] = 0, \tag{22}$$

where $\mathbf{x}_p = (x, y, z)^T$, $\mathbf{x}_o = (m_x, m_y, 1)$ and $R_{2\times2}$ is the top 2×2-block of R. We can also interpret these as the 1D reprojection equations

$$\lambda \begin{pmatrix} x \\ y \end{pmatrix} = R_{2\times2}^T \left(\begin{pmatrix} m_x \\ m_y \end{pmatrix} - \begin{pmatrix} t_1 \\ t_2 \end{pmatrix} \right), \qquad (23)$$

where λ is the unknown depth.

References

[1] C. Arth, C. Pirchheim, J. Ventura, and V. Lepetit. Global 6dof pose estimation from untextured 2d city models. *arXiv preprint arXiv:1503.02675*, 2015. 2

[2] C. Arth, C. Pirchheim, J. Ventura, D. Schmalstieg, and V. Lepetit. Instant outdoor localization and slam initialization from 2.5 d maps. *IEEE Transactions on Visualization and Computer Graphics*, 11(21):1309–1318, 2015. 2

[3] F. Boniardi, T. Caselitz, R. Kümmerle, and W. Burgard. Robust lidar-based localization in architectural floor plans. In *International Conference on Intelligent Robots and Systems (IROS)*, 2017. 2

[4] T.-J. Cham, A. Ciptadi, W.-C. Tan, M.-T. Pham, and L.-T. Chia. Estimating camera pose from a single urban ground-view omnidirectional image and a 2d building outline map. In *Computer Vision and Pattern Recognition (CVPR)*, 2010. 2

[5] H. Chu, A. Gallagher, and T. Chen. Gps refinement and camera orientation estimation from a single image and a 2d map. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2014. 2

[6] J. Domhof, J. F. Kooij, and D. M. Gavrila. An extrinsic calibration tool for radar, camera and lidar. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 8107–8113. IEEE, 2019. 2

[7] O. D. Faugeras and F. Lustman. Motion and structure from motion in a piecewise planar environment. *International Journal of Pattern Recognition and Artificial Intelligence*, 2(03):485–508, 1988. 3

[8] X. Gao, S. Shen, Z. Hu, and Z. Wang. Ground and aerial meta-data integration for localization and reconstruction: A review. *Pattern Recognition Letters*, 2019. 2

[9] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, USA, 2003. 4

[10] R. I. Hartley. Projective reconstruction and invariants from multiple images. *Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 1994. 3

[11] R. S. Kaminsky, N. Snavely, S. M. Seitz, and R. Szeliski. Alignment of 3d point clouds to overhead images. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2009. 1

[12] Z. Kukelova, J. Kileel, B. Sturmfels, and T. Pajdla. A clever elimination strategy for efficient minimal solvers. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4912–4921, 2017. 3

[13] V. Larsson, K. Astrom, and M. Oskarsson. Efficient solvers for minimal problems by syzygy-based reduction. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[14] E. Malis and M. Vargas. Deeper understanding of the homography decomposition for vision-based control. In *INRIA Research Report*, 2007. 3

[15] G. E. Natour, O. Ait-Aider, R. Rouveure, F. Berry, and P. Faure. Toward 3d reconstruction of outdoor scenes using an mmw radar and a monocular vision sensor. *Sensors*, 15(10):25937–25967, 2015. 2

[16] K. Ni, N. Armstrong-Crews, and S. Sawyer. Geo-registering 3d point clouds to 2d maps with scan matching and the hough transform. In *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013. 2

[17] D. Nistér. An efficient solution to the five-point relative pose problem. *Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 26(6):756–770, 2004. 3

[18] M. Park, J. Luo, R. T. Collins, and Y. Liu. Beyond gps: Determining the camera viewing direction of a geotagged image. In *ACM International Conference on Multimedia*, 2010. 2

[19] J. Peršić, I. Marković, and I. Petrović. Extrinsic 6dof calibration of a radar–lidar–camera system enhanced by radar cross section estimates evaluation. *Robotics and Autonomous Systems*, 114:217–230, 2019. 2

[20] C. Schöller, M. Schnettler, A. Krämmer, G. Hinz, M. Bakovic, M. Güzet, and A. Knoll. Targetless rotational auto-calibration of radar and camera for intelligent transportation systems. In *Intelligent Transportation Systems Conference (ITSC)*, 2019. 2

[21] J. G. Semple and G. T. Kneebone. *Algebraic projective geometry*. Clarendon Press, 1952. 2

[22] Q. Shan, C. Wu, B. Curless, Y. Furukawa, C. Hernandez, and S. M. Seitz. Accurate geo-registration by ground-to-aerial image matching. In *International Conference on 3D Vision (3DV)*, 2014. 2

[23] H. Stewénius, M. Oskarsson, K. Aström, and D. Nistér. Solutions to minimal generalized relative pose problems. In *Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras (OMNIVIS)*, 2005. 2

[24] S. Sugimoto, H. Tateda, H. Takahashi, and M. Okutomi. Obstacle detection using millimeter-wave radar and its visualization on image sequence. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 3, pages 342–345. IEEE, 2004. 2

[25] C. Sweeney, J. Flynn, and M. Turk. Solving for relative pose with a partially known rotation is a quadratic eigenvalue problem. In *International Conference on 3D Vision (3DV)*, 2014. 4

[26] T. Wang, N. Zheng, J. Xin, and Z. Ma. Integrating millimeter wave radar with a monocular vision sensor for on-road obstacle detection applications. *Sensors*, 11(9):8992–9008, 2011. 2

[27] E. Wijmans and Y. Furukawa. Exploiting 2d floorplan for building-scale panorama rgbd alignment. In *Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[28] X. Zhang, G. Agam, and X. Chen. Alignment of 3d building models with satellite images using extended chamfer matching. In *Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2014. 2
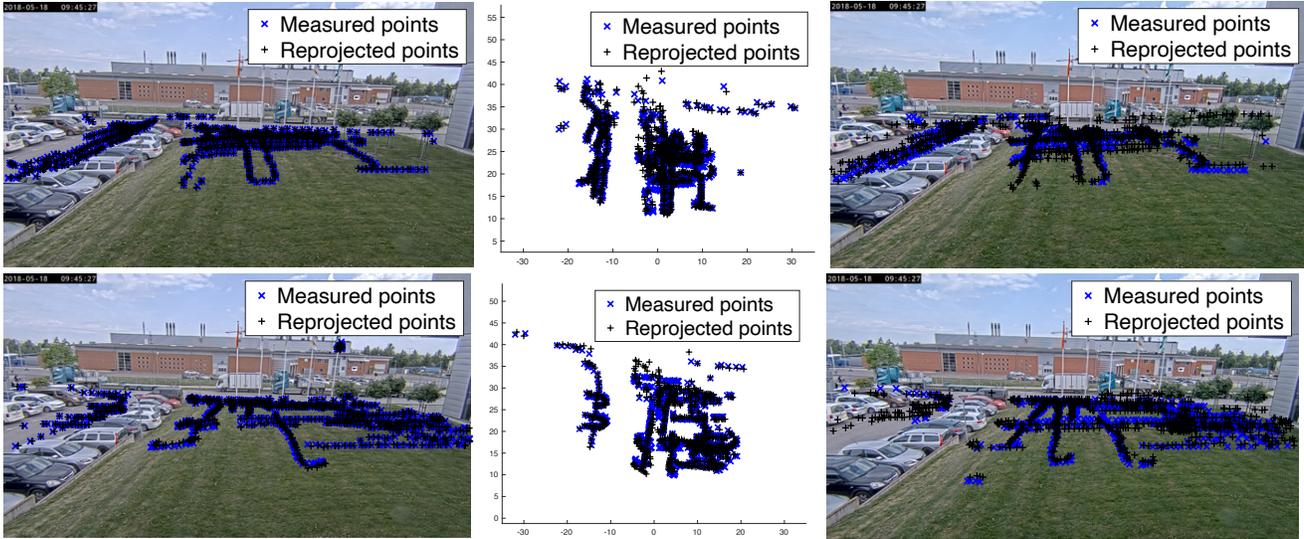
Figure 3. Reprojection errors for the radar experiment using the proposed method. *Left:* Camera image. *Middle:* Radar image. *Right:* Homography.. See Section 5 for more details on the experiment.
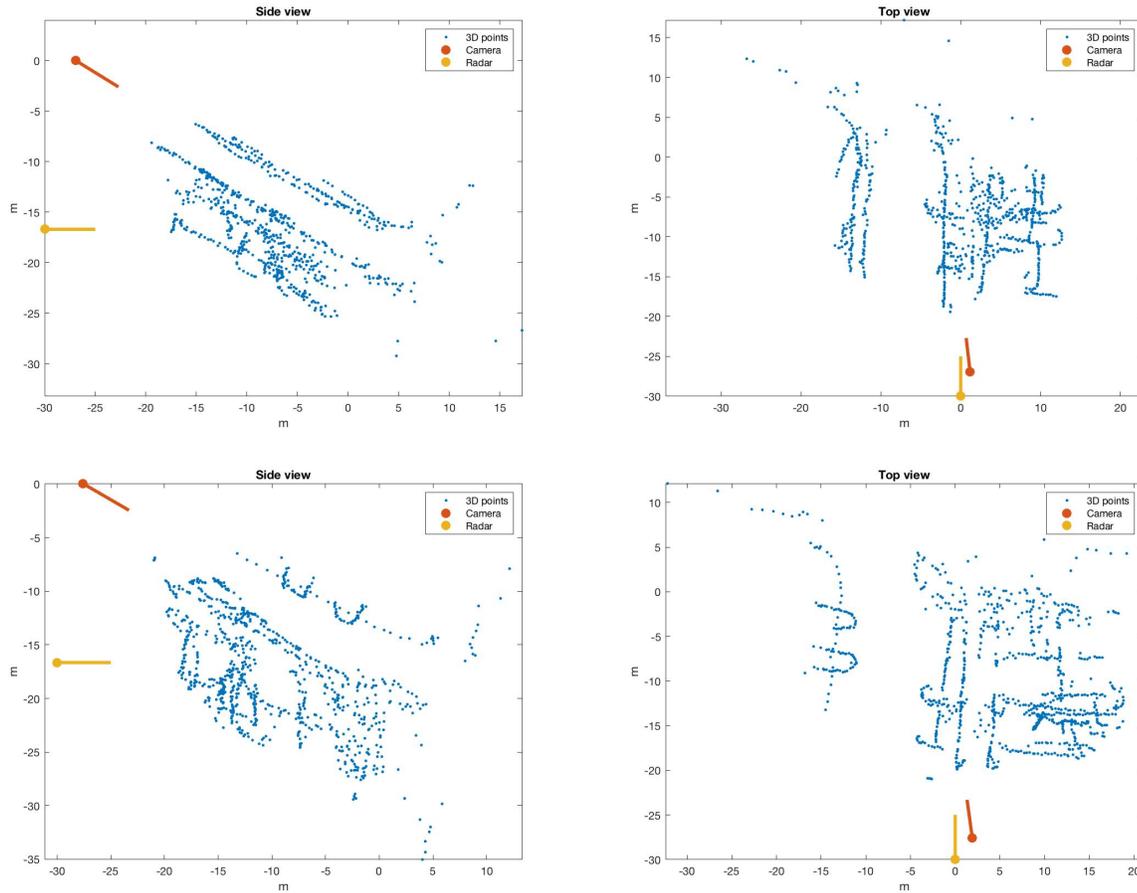
Figure 4. Reconstruction results for the two datasets (top and bottom respectively). Left shows the view from one side, and right shows the view from the top (*i.e.* in the direction of the orthographic camera). The 3D points are shown in blue, and the estimated camera and radar positions and orientations are given in yellow and red respectively. One can see that we get consistent relative pose estimated for the two datasets.

[29] Z. Zhang, P. Anandan, and H.-Y. Shum. What can be determined from a full and a weak perspective image? In *International Conference on Computer Vision (ICCV)*, 1999. 1

[30] Z. Zhang and A. R. Hanson. 3d reconstruction based on homography mapping. *Proc. ARPA96*, pages 1007–1012, 1996. 3