Trajectory Alignment and Evaluation in SLAM: Horn's Method vs Alignment on the Manifold

Marta Salas, Universidad de Zaragoza, msalasg@unizar.es

Yasir Latif, University of Adelaide/ACRV, yasir.latif@adelaide.edu.au

Ian D. Reid, University of Adelaide/ACRV, ian.reid@adelaide.edu.au

J.M.M. Montiel, Universidad de Zaragoza, josemari@unizar.es

***Abstract*—Under the assumption of identical covariance at each pose, Horn's method finds an aligning transform that is very similar to the one found using an optimization on the manifold.**

I. Introduction

Given the output of a Simultaneous Localization And Mapping (SLAM) system, that is, environment representation and robot poses, how can we measure its performance? Comparing environment representations is difficult and in some cases not possible due to lack of a usable ground truth. Alternatively, ground truth trajectory can be easily acquired using either a motion capture set up or high quality GPS. Therefore it is common to assess the quality of a SLAM system by comparing the estimated robot poses against those in the ground truth. In general, two metrics have been employed for this purpose:
a) *Absolute Trajectory Error (ATE)*, which measures the difference between the translation part of two trajectories by first aligning them into a common reference frame, and
b) *Relative Pose Error (RPE)*, which measures the difference between relative transformations at time instances i and $i+k$, for different values of k. This method is independent of the reference frame but when the scale of the map is not known (for example monocular mapping), a scale alignment needs to be done before comparing trajectories using RPE.

Formally, assuming a set of poses from an estimated trajectory: $\mathbf{P}_1, \mathbf{P}_2, ..., \mathbf{P}_N$ and the corresponding time-aligned ground truth poses: $\mathbf{Q}_1, \mathbf{Q}_2, ..., \mathbf{Q}_N$ where each pose belongs to the appropriate Lie Group $\mathbf{G}(n)$, we are interested in finding a space and/or scale aligning transform $\mathbf{T}$. The error $\mathbf{E}_i$ between $\mathbf{P}_i$ and $\mathbf{Q}_i$ is then given by: $\mathbf{E}_i = \mathbf{Q}_i^{-1}\mathbf{T}\mathbf{P}_i$. The overall error is then calculated as the Root Mean Squared (RMS) error over the translation components of all the poses (7).

Predominantly two approaches are used for trajectory alignment: **a)** Sturm et al. [8] uses a least squares approximation and **b)** Horn's method [4] provides a closed-form solution using unit quaternion. These methods rely on the translation part of the poses and do not take into account the rotational part. The reasoning behind relies in the fact that SLAM is a sequential algorithm, and consequently, errors in rotation would show up as errors in translation later on, therefore, it is sufficient to align trajectories based on their translational component. This work observes the validity of this assumption by formulating trajectory alignment as an optimization problem on the manifold, which explicitly takes into account both the rotational as well as the translational components.

II. Trajectory Alignment on the Manifold

We formulate the trajectory alignment problem as a least squares optimization by minimizing:

$$\arg\min_{\mathbf{T}} \sum_{i=1}^{N} \mathbf{d}_i^T(\mathbf{T})\mathbf{\Lambda}_i\mathbf{d}_i(\mathbf{T}), \tag{1}$$

where d_i is the residual error and $\mathbf{\Lambda}$ is a covariance matrix. The residual error in the tangent space $\mathfrak{g}(n)$ is given by:

$$\mathbf{d}_i := log\left((\mathbf{Q}_i)^{-1}\mathbf{T}\mathbf{P}_i\right)^{\vee}, \tag{2}$$

where $log : \mathbf{G} \rightarrow \mathfrak{g}$ is the logarithmic map which takes an element $\mathbf{G}$ into its tangent space $\mathfrak{g}$. The vee-operator $log(\cdot)^{\vee} := (log(\cdot))^{\vee}$ maps the element in the tangent space to its minimal representation. We use Levenberg-Marquardt algorithm to solve (1) by iteratively solving the *normal equation*:

$$\left(\mathbf{J}^T\mathbf{\Lambda}\mathbf{J} + \mu\mathbf{I}\right)\boldsymbol{\delta} = -\mathbf{J}^T\mathbf{\Lambda}\mathbf{d}(\mathbf{T}), \tag{3}$$

where $\mathbf{J}$ is the Jacobian of (2). Then, the update rule at each step of the optimization is given by:

$$\mathbf{T}^{k+1} = exp(\hat{\boldsymbol{\delta}})\mathbf{T}^k, \tag{4}$$

where $exp : \mathfrak{g} \rightarrow \mathbf{G}$ is the exponential map which takes an element of the tangent space $\mathfrak{g}(n)$ back into $\mathbf{G}(n)$. Detailed derivation for $\mathbf{SE}(3)$ and $\mathbf{Sim}(3)$ are given in Appendix A.

III. Benchmarking Metrics

Once the needed transformation is found, the next step is to define a metric that would tell us how well our SLAM algorithm is performing. In order to do this, we present a benchmark for measuring ATE which takes into account rotation errors as well as translational errors. For translation, we use the well-known mean, standard deviation and root mean square error (RMSE):

$$trans_{mean} = \frac{1}{N}\sum_{i=1}^{N} ||trans(\mathbf{E}_i)||_2 \tag{5}$$

$$trans_{std} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (trans(\mathbf{E})_i - trans_{mean})^2} \tag{6}$$

$$trans_{rmse} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} ||trans(\mathbf{E}_i)||^2} \tag{7}$$

		translation error(cm)		rotation error (deg)
		mean ± std	rmse	mean ± std
translation + noise	Horn [4]	0.88 ± 0.37	0.95	0.00 ± 0.00
	manifold	0.88 ± 0.37	0.95	0.00 ± 0.00
rotation + noise	Horn [4]	0.00± 0.00	0.00	1.00 ± 0.42
	manifold	0.00± 0.00	0.00	1.00 ± 0.42
trans + rot + noise	Horn [4]	0.88 ± 0.37	0.95	1.00 ± 0.42
	manifold	0.88 ± 0.37	0.95	1.00 ± 0.42

TABLE I: Simulation results $\mathbf{SE}(3)$. Average error on the trajectory.

		translation error(cm)		rotation error (deg)
		mean ± std	rmse	mean ± std
translation + noise	Horn [4]	0.88 ± 0.37	0.95	0.00 ± 0.00
	manifold	0.88 ± 0.37	0.95	0.00 ± 0.00
rotation + noise	Horn [4]	0.00± 0.00	0.00	1.00 ± 0.42
	manifold	0.00± 0.00	0.00	1.00 ± 0.42
trans + rot + scale + noise	Horn [4]	0.50 ± 0.21	0.54	1.00 ± 0.42
	manifold	0.50 ± 0.21	0.54	1.00 ± 0.42

TABLE II: Simulation results $\mathbf{Sim}(3)$. Average error on the trajectory.

For rotation, we propose using the circular mean and the circular standard deviation because arithmetic mean is not appropriate for circular quantities [2]. First, we recover the angle θ_i from the rotation of $\mathbf{E}_i$ by applying (11) to $rot(\mathbf{E}_i)$, then we calculate:

$$rot_{mean} = \begin{cases} tan^{-1}\left(\frac{\overline{S}}{\overline{C}}\right) & \overline{S} > 0 \text{ and } \overline{C} > 0 \\ tan^{-1}\left(\frac{\overline{S}}{\overline{C}}\right) + \pi & \overline{C} < 0 \\ tan^{-1}\left(\frac{\overline{S}}{\overline{C}}\right) + 2\pi & \overline{S} < 0 \text{ and } \overline{C} > 0 \end{cases} \quad (8)$$

$$rot_{std} = \sqrt{-2ln\left(\sqrt{\left(\overline{S}\right)^2 + \left(\overline{C}\right)^2}\right)} \quad (9)$$

with $\overline{S} = \frac{\sum_{i=1}^{N} sin(\theta_i)}{N}, \overline{C} = \frac{\sum_{i=1}^{N} cos(\theta_i)}{N}$.

IV. EXPERIMENTS

The basic aim of the experiments is to compare the performance of Horn's method to that of optimization on the manifold in terms of the resulting error, measured according to the presented metrics. Firstly, synthetic noise is added in various configurations to a ground truth trajectory and in the second part, trajectories obtained with different SLAM systems are compared using both methods.

A. Simulation experiments

For the simulated experiments, we use the ground truth trajectory from the RGB-D Benchmark [8] for the sequence *fr3/nostructure_texture_near_withloop*. We generate "noisy" trajectories by applying a series of transformations with different levels of Gaussian noise $\mathcal{N}(0, \sigma)$ with $\sigma \in [0.001, 0.01]$. The different scenarios are listed in Tables I and II. In each case, we run 10 different experiments using 10 different levels of noise. As it can be seen from Tables I and II, both methods perform equally well.

B. Real data

We report the comparison of Horn's method and our own for different SLAM systems. We present results for PTAM [5] and RGB-D SLAM [6] on the aforementioned sequence from the RGB-D Dataset. We run PTAM 10 times with two different initializations and report average results in Table III. PTAM is a monocular SLAM system and therefore the aligning transform is in $\mathbf{Sim}(3)$. RGB-D SLAM uses RGB and depth information to estimate the pose of the robot. The transformation between the estimated trajectory and ground truth in this case is in $\mathbf{SE}(3)$. Table III shows results for RGB-D SLAM performance. We also consider the case when there is a scale drift, and try to find an alignment in $\mathbf{Sim}(3)$. In all these experiments, manifold optimization is able to find better angular alignment but at the cost of increased translational error.

		translation error (cm)		rotation error (deg)
		mean ± std	rmse	mean ± std
PTAM [5] Initialization 1	Horn [4]	2.18 ± 1.73	**2.87**	0.99 ± 0.82
	manifold	**2.10 ± 1.91**	2.93	**0.97 ± 0.81**
PTAM [5] Initialization 2	Horn [4]	2.37 ± 2.40	**3.47**	1.08 ± 1.07
	manifold	**2.06 ± 2.67**	3.57	**1.02 ± 1.09**
RGBD-SLAM [6] $\mathbf{SE}(3)$	Horn [4]	8.45 ± 2.16	8.72	3.90 ± 0.57
	manifold	8.45 ± 2.16	8.72	3.90 ± 0.57
RGBD-SLAM [6] $\mathbf{Sim}(3)$	Horn [4]	2.45 ± 1.21	**2.73**	3.87 ± 0.59
	manifold	**2.44 ± 1.26**	2.74	**3.86 ± 0.59**

TABLE III: ATE error comparison on the sequence *fr3/nostructure_texture_near_withloop* between Horn's method and manifold alignment.

V. CONCLUSION

Under the assumption that all the covariance matrices are the same, Horn's method which provides a closed form solution to the trajectory alignment problem, performs as well as finding an alignment on the manifold. For experiments with real data, manifold optimization finds a transformation that aligns the angular components better than Horn's method, but this comes at the cost of increased translation error, which is understandable and expected. For SLAM systems that generate an estimate of uncertainty at each pose, manifold optimization allows a natural way of incorporating the uncertainty which can be used to correctly weigh the importance of translation and angular components during optimization.

ACKNOWLEDGMENTS

This research has been funded by the Dirección General de Investigación of Spain under projects DPI2012-32168 and DPI2012-32100. YL and IDR are extremely grateful to the Australian Research Council for funding this research through the ARC Centre for Robotic Vision C E140100016 [and through a Laureate Fellowship FL130100102 to IDR]. Authors would like to thank O. Grasa, P. Urcola and W. Hussain for fruitful discussions.

REFERENCES

[1] J. E. Campbell. On a law of combination of operators bearing on the theory of continuous transformation groups. *Proceedings of the London Mathematical Society*, 28:381–390, 1897.

[2] N. I. Fisher. *Statistical analysis of spherical data*. Cambridge University Press, 1993.

[3] J. Gallier. *Geometric methods and applications: for computer science and engineering*, volume 38. Springer Science & Business Media, 2011.

[4] B. K. Horn. Closed-form solution of absolute orientation using unit quaternions. *JOSA A*, 4(4):629–642, 1987.

[5] G. Klein and D. Murray. Parallel Tracking and Mapping for Small AR Workspaces. In *IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 225–234, 2007.

[6] F. Steinbruecker, J. Sturm, and D. Cremers. Real-time visual odometry from dense rgb-d images. In *Workshop on Live Dense Reconstruction with Moving Cameras at the Intl. Conf. on Computer Vision (ICCV)*, 2011.

[7] H. Strasdat, J. Montiel, and A. J. Davison. Scale drift-aware large scale monocular slam. In *Robotics: Science and Systems (RSS)*, 2010.

[8] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers. A benchmark for the evaluation of RGB-D SLAM systems. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 573–580. IEEE, 2012.

APPENDIX

A: OPTIMIZATION ON THE MANIFOLD

1) $\mathbf{T} \in \mathbf{SE}(3)$ *:* We look at the Lie group $\mathbf{SE}(3)$ which defines the group of rigid body transformations in the three dimensional space. Its elements are of the form:

$$\mathbf{H} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix}, \tag{10}$$

where $\mathbf{t} \in \mathbb{R}^3$ is a translation and $\mathbf{R} \in \mathbf{SO}(3)$ is a rotation in 3D.

The residual error $\mathbf{d} = (\boldsymbol{\upsilon}, \boldsymbol{\omega})$ in the tangent space $\mathfrak{se}(3)$ has a minimal representation which is a vector $\in \mathbb{R}^6$. Typically the first three elements represent a translation $\boldsymbol{\upsilon} \in \mathbb{R}^3$ and the latter three elements represent a rotation $\boldsymbol{\omega} \in \mathbb{R}^3$ in the axis-angle form.

The operator $log : \mathbf{SE}(3) \to \mathfrak{se}(3)$ maps an element $\mathbf{H}$ onto a vector in the tangent space $(\boldsymbol{\upsilon}, \boldsymbol{\omega})$.

$$\theta = arcos\left(\frac{tr(\mathbf{R}) - 1}{2}\right) \tag{11}$$

$$ln(\mathbf{R}) = \frac{\theta}{2sin(\theta)}(\mathbf{R} - \mathbf{R}^T) \tag{12}$$

$$\boldsymbol{\omega} = [ln(\mathbf{R})]_\triangledown \tag{13}$$

$$\boldsymbol{\upsilon} = \mathbf{V}^{-1}\mathbf{t} \tag{14}$$

where $[\cdot]_\triangledown$ represents the off-diagonal elements and $\mathbf{V}$ is defined as [3]:

$$\mathbf{V} = \begin{cases} \mathbf{I} & if\ \theta \to 0 \\ \mathbf{I} + \frac{1-cos(\theta)}{\theta^2}[\boldsymbol{\omega}]_\times + \frac{\theta - sin(\theta)}{\theta^3}[\boldsymbol{\omega}]^2_\times & otherwise \end{cases} \tag{15}$$

with $\theta = ||\boldsymbol{\omega}||_2$ and

$$[\boldsymbol{\omega}]_\times = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \in \mathbf{SO}(3), \tag{16}$$

In order to map back to the original space, the operator $exp : \mathfrak{se}(3) \to \mathbf{SE}(3)$ is used which maps a vector in the tangent space $(\boldsymbol{\upsilon}, \boldsymbol{\omega})$ onto an element $\mathbf{H}$:

$$exp\left(\boldsymbol{\upsilon}, \boldsymbol{\omega}\right)_{\mathfrak{se}(3)} := \begin{pmatrix} exp\left([\boldsymbol{\omega}]_\times\right) & \mathbf{V}\boldsymbol{\upsilon} \\ \mathbf{0}_{1\times3} & 1 \end{pmatrix} \in \mathbf{SE}(3), \tag{17}$$

Finally, the Jacobian $\mathbf{J}$ required in (3) is given by:

$$\mathbf{J} = \frac{\partial}{\partial \boldsymbol{\delta}} log\left(\mathbf{Q}_i^{-1} exp(\boldsymbol{\delta})\mathbf{T}\mathbf{P}_i\right)^\vee_{\mathfrak{se}(3)} |_{\delta=0} \approx Adj_{\mathbf{Q}_i^{-1}} \tag{18}$$

We approximate the Jacobian by the Adjoint map (Adj) using the first order Campbell-Baker-Hausdorff expansion [1]. The Adjoint map of $\mathbf{H} \in \mathbf{SE}(3)$ is:

$$Adj_{\mathbf{H}} := \begin{bmatrix} \mathbf{R} & [\mathbf{t}]_\times\mathbf{R} \\ \mathbf{0}_{1\times3} & \mathbf{R} \end{bmatrix} \in \mathbb{R}^{6\times6} \tag{19}$$

2) $\mathbf{T} \in \mathbf{Sim}(3)$ *:* Similarly, the group $\mathbf{Sim}(3)$ defines the group of similarity transformations in 3D, whose elements are of the form:

$$\mathbf{S} = \begin{bmatrix} s\mathbf{R} & \mathbf{t} \\ \mathbf{0}_{1\times3} & 1 \end{bmatrix}, \tag{20}$$

where $\mathbf{t} \in \mathbb{R}^3$ is a translation and $\mathbf{R} \in \mathbf{SO}(3)$ a rotation, and $s \in \mathbb{R}$ is a scale.

The residual error $\mathbf{d} = (\boldsymbol{\upsilon}, \boldsymbol{\omega}, \sigma)$ in the tangent space $\mathfrak{sim}(3)$ has a minimal representation $\in \mathbb{R}^7$. As previously defined, $\mathbf{d}_i$ defines a translation $\boldsymbol{\upsilon} \in \mathbb{R}^3$, a rotation $\boldsymbol{\omega} \in \mathbb{R}^3$ and a scale $\sigma \in \mathbb{R}$.

The operator $log : \mathbf{Sim}(3) \to \mathfrak{sim}(3)$ maps an element S into the tangent space $(\boldsymbol{\upsilon}, \boldsymbol{\omega}, \sigma)$: $\boldsymbol{\omega}$ is recovered by applying 11,12,13, and $(\boldsymbol{\upsilon}, \sigma)$ using:

$$\sigma = ln(s) \tag{21}$$

$$\boldsymbol{\upsilon} = \mathbf{W}^{-1}\mathbf{t} \tag{22}$$

where $\mathbf{W}$ is [7]:

$$\mathbf{W} = \begin{cases} \mathbf{I} & if\ \theta \to 0, \sigma \to 0 \\ C\mathbf{I} & if\ \theta \to 0 \\ \mathbf{I} + \frac{1-cos(\theta)}{\theta^2}[\boldsymbol{\omega}]_\times + \frac{\theta - sin(\theta)}{\theta^3}[\boldsymbol{\omega}]^2_\times & if\ \sigma \to 0 \\ C\mathbf{I} + \frac{A\sigma + (1-B)\theta}{\sigma^2 + \theta^2}\left(\frac{[\boldsymbol{\omega}]_\times}{\theta}\right) + & \\ \quad \left(C - \frac{(B-1)\sigma + A\theta}{\sigma^2+\theta^2}\right)\left(\frac{[\boldsymbol{\omega}]_\times}{\theta}\right)^2 & otherwise \end{cases} \tag{23}$$

with $A = e^\sigma sin(\theta), B = e^\sigma cos(\theta), C = \dfrac{e^\sigma - 1}{\sigma}, \theta = ||\boldsymbol{\omega}||_2$.

The operator $exp : \mathfrak{sim}(3) \to \mathbf{Sim}(3)$ is given by:

$$exp\left(\boldsymbol{\upsilon}, \boldsymbol{\omega}, \sigma\right)_{\mathfrak{sim}(3)} := \begin{pmatrix} e^\sigma exp\left([\boldsymbol{\omega}]_\times\right) & \mathbf{W}\boldsymbol{\upsilon} \\ \mathbf{0}_{1\times3} & 1 \end{pmatrix} \tag{24}$$

Finally, the Jacobian $\mathbf{J}$ is given by:

$$\mathbf{J} = \frac{\partial}{\partial \boldsymbol{\delta}} log\left(\mathbf{Q}_i^{-1} exp(\boldsymbol{\delta})\mathbf{T}\mathbf{P}_i\right)^\vee_{\mathfrak{sim}(3)} |_{\delta=0} \approx Adj_{\mathbf{Q}_i^{-1}} \tag{25}$$

We approximate again the Jacobian by the Adjoint using the first order Campbell-Baker-Hausdorff expansion. The Adjoint map of $\mathbf{S} \in \mathbf{Sim}(3)$ is:

$$Adj_{\mathbf{S}} = \begin{bmatrix} s\mathbf{R} & [\mathbf{t}]_\times\mathbf{R} & -\mathbf{t} \\ \mathbf{0}_{3\times3} & \mathbf{R} & \mathbf{0}_{3\times1} \\ \mathbf{0}_{1\times3} & \mathbf{0}_{1\times3} & 1 \end{bmatrix} \in \mathbb{R}^{7\times7} \tag{26}$$