Supplementary Material

1. SO(3), SE(3) and Sim(3) Formulas

Given a Lie group $\mathcal{G}$ with Lie algebra $\mathfrak{g}$, we provide the expressions for $SO(3)$, $SE(3)$, and $Sim(3)$

$\wedge$ **Operator:** The $\wedge$ operator takes elements from $\mathbb{R}^k$ to the lie algebra $\mathfrak{g}$. For $\boldsymbol{\phi} \in \mathbb{R}^3$

$$\boldsymbol{\phi}^\wedge = \begin{pmatrix} 0 & \text{-}\phi_z & \phi_y \\ \phi_z & 0 & \text{-}\phi_x \\ \text{-}\phi_y & \phi_x & 0 \end{pmatrix} \in \mathfrak{so}(3) \tag{1}$$

for $\boldsymbol{\xi} = (\boldsymbol{\tau}, \boldsymbol{\phi}) \in \mathbb{R}^6$

$$\boldsymbol{\xi}^\wedge = \begin{pmatrix} \boldsymbol{\phi}^\wedge & \boldsymbol{\tau} \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} 0 & \text{-}\phi_z & \phi_y & \tau_x \\ \phi_z & 0 & \text{-}\phi_x & \tau_y \\ \text{-}\phi_y & \phi_x & 0 & \tau_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \in \mathfrak{se}(3) \tag{2}$$

and for $\boldsymbol{\eta} = (\boldsymbol{\tau}, \boldsymbol{\phi}, \sigma) \in \mathbb{R}^7$

$$\boldsymbol{\eta}^\wedge = \begin{pmatrix} \boldsymbol{\phi}^\wedge + \sigma \mathbf{I}_{3\times3} & \boldsymbol{\tau} \\ \sigma & 1 \end{pmatrix} = \begin{pmatrix} \sigma & \text{-}\phi_z & \phi_y & \tau_x \\ \phi_z & \sigma & \text{-}\phi_x & \tau_y \\ \text{-}\phi_y & \phi_x & \sigma & \tau_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \in \mathfrak{sim}(3) \tag{3}$$

$\curlywedge$ **Operator:** The $\curlywedge$ operator takes elements from $\mathbb{R}^k$ to the lie algebra $\mathrm{adj}(\mathfrak{g})$, where $\mathrm{adj}(\mathfrak{g})$ is the Lie algebra associated with the group $\mathrm{Adj}(\mathcal{G}) = \{\mathrm{Adj}(X) \mid X \in \mathcal{G}\}$, It can be shown that $\mathrm{Adj}(\mathcal{G})$ also forms a Lie group [1].

For $\boldsymbol{\phi} \in \mathbb{R}^3$

$$\boldsymbol{\phi}^\curlywedge = \boldsymbol{\phi}^\wedge \in \mathrm{adj}(\mathfrak{so}(3)) \tag{4}$$

for $\boldsymbol{\xi} = (\boldsymbol{\tau}, \boldsymbol{\phi}) \in \mathbb{R}^6$

$$\boldsymbol{\xi}^\curlywedge = \begin{pmatrix} \boldsymbol{\phi}^\wedge & \boldsymbol{\tau}^\wedge \\ \mathbf{0} & \boldsymbol{\phi}^\wedge \end{pmatrix} \in \mathrm{adj}(\mathfrak{se}(3)) \tag{5}$$

and for $\boldsymbol{\eta} = (\boldsymbol{\tau}, \boldsymbol{\phi}, \sigma) \in \mathbb{R}^7$

$$\boldsymbol{\eta}^\curlywedge = \begin{pmatrix} \boldsymbol{\phi}^\wedge + \sigma \mathbf{I}_{3\times3} & \boldsymbol{\tau}^\wedge & -\boldsymbol{\tau} \\ \mathbf{0} & \boldsymbol{\phi}^\wedge & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & 0 \end{pmatrix} \in \mathrm{adj}(\mathfrak{sim}(3)) \tag{6}$$

Exp **Map:** The exponential map takes elements from the Lie algebra to the Lie group. For $SO(3)$, $SE(3)$, and $Sim(3)$ the exponential map has a closed form expressions. For $\boldsymbol{\phi} \in \mathbb{R}^3$

$$\mathrm{Exp}(\boldsymbol{\phi}) = \exp(\boldsymbol{\phi}^\wedge) = \mathbf{I}_{3\times3} + \frac{\sin(\theta)}{\theta}\boldsymbol{\phi}^\wedge + \frac{1-\cos(\theta)}{\theta^2}(\boldsymbol{\phi}^\wedge)^2, \qquad \theta = ||\boldsymbol{\phi}|| \tag{7}$$

for $\boldsymbol{\xi} = (\boldsymbol{\tau}, \boldsymbol{\phi}) \in \mathbb{R}^6$

$$\mathrm{Exp}(\boldsymbol{\xi}) = \begin{pmatrix} \mathbf{R} & \mathbf{V}\boldsymbol{\tau} \\ \mathbf{0} & 1 \end{pmatrix}, \qquad \mathbf{R} = \mathrm{Exp}(\boldsymbol{\phi}) \tag{8}$$

$$\mathbf{V} = \mathbf{I}_{3\times3} + \frac{1-\cos(\theta)}{\theta^2}\boldsymbol{\phi}^\wedge + \frac{\theta - \sin(\theta)}{\theta^3}(\boldsymbol{\phi}^\wedge)^2, \qquad \theta = ||\boldsymbol{\phi}|| \tag{9}$$

and for $\boldsymbol{\eta} = (\boldsymbol{\tau}, \boldsymbol{\phi}, \sigma) \in \mathbb{R}^7$

$$\text{Exp}(\boldsymbol{\xi}) = \begin{pmatrix} e^{\sigma}\mathbf{R} & \mathbf{W}\boldsymbol{\tau} \\ \mathbf{0} & 1 \end{pmatrix}, \qquad \mathbf{R} = \text{Exp}(\boldsymbol{\phi}) \tag{10}$$

$$\mathbf{W} = \left(\frac{e^{\sigma}-1}{\sigma}\right)\mathbf{I}_{3\times 3} + \frac{1}{\theta}\left(\frac{e^{\sigma}\sin(\theta)\sigma + (1-e^{\sigma}\cos(\theta))\theta}{\sigma^2+\theta^2}\right)\boldsymbol{\phi}^{\wedge} + \tag{11}$$

$$\frac{1}{\theta^2}\left(\frac{e^{\sigma}-1}{\sigma} + \frac{(e^{\sigma}\cos(\theta)-1)\sigma + e^{\sigma}\sin(\theta)\theta}{\sigma^2+\theta^2}\right)(\boldsymbol{\phi}^{\wedge})^2, \qquad \theta = ||\boldsymbol{\phi}|| \tag{12}$$

When θ or σ is small, we use second order Taylor approximations of the exponential maps to avoid numerical issues.

Log **Map:** The logarithm map takes elements from the Lie group to the Lie algebra. For $SO(3)$, $SE(3)$, and $Sim(3)$ the logarithm map can be computed in closed form. For a rotation $\mathbf{R} \in SO(3)$

$$\text{Log}(\mathbf{R}) = \frac{\psi(\mathbf{R}-\mathbf{R}^T)^{\vee}}{2\sin(\psi)}, \ \psi = \cos^{-1}\left(\frac{tr(\mathbf{R})-1}{2}\right) \tag{13}$$

for $\mathbf{G} = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix} \in SE(3)$ we have

$$\boldsymbol{\xi} = \begin{pmatrix} \boldsymbol{\tau} \\ \boldsymbol{\phi} \end{pmatrix} = \begin{pmatrix} \mathbf{V}^{-1}\mathbf{t} \\ \text{Log}(\mathbf{R}) \end{pmatrix} = \text{Log}(\mathbf{G}) \tag{14}$$

$$\mathbf{V}^{-1} = \mathbf{I}_{3\times 3} - \frac{1}{2}\phi^{\wedge} + \left(\frac{1}{\phi^2} - \frac{1+\cos\theta}{2\theta\sin\theta}\right)(\phi^{\wedge})^2, \qquad \theta = ||\boldsymbol{\phi}|| \tag{15}$$

and for $\mathbf{T} = \begin{pmatrix} s\mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix} \in Sim(3)$

$$\boldsymbol{\eta} = \begin{pmatrix} \boldsymbol{\tau} \\ \boldsymbol{\phi} \\ \sigma \end{pmatrix} = \begin{pmatrix} \mathbf{W}^{-1}\mathbf{t} \\ \text{Log}(\mathbf{R}) \\ \ln(s) \end{pmatrix} = \text{Log}(\mathbf{T}) \tag{16}$$

where $\mathbf{W}^{-1}$ can be computed by taking the inverse Eqn. 12.

Adj **Operator:** The ajoint operator is a linear map which allows us to move an element $\nu \in \mathfrak{g}$ in the right tangent space of $X \in \mathcal{G}$ to the left tangent space

$$\text{Exp}(\text{Adj}_X \, \nu) \circ X = X \circ \text{Exp}(\nu) \tag{17}$$

For $\mathbf{R} \in SO(3)$

$$\text{Adj}_{\mathbf{R}} = \mathbf{R} \tag{18}$$

for $\mathbf{G} = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix} \in SE(3)$

$$\text{Adj}_{\mathbf{G}} = \begin{pmatrix} \mathbf{R} & \boldsymbol{\tau}^{\wedge}\mathbf{R} \\ \mathbf{0} & \mathbf{R} \end{pmatrix} \tag{19}$$

and for $\mathbf{T} = \begin{pmatrix} s\mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix} \in Sim(3)$ we have

$$\text{Adj}_{\mathbf{T}} = \begin{pmatrix} s\mathbf{R} & \boldsymbol{\tau}^{\wedge}\mathbf{R} & -\mathbf{t} \\ \mathbf{0} & \mathbf{R} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & 1 \end{pmatrix} \tag{20}$$

Inv **Operator:** Since $SO(3)$, $SE(3)$, and $Sim(3)$ all form a group, each element has a unique inverse. For $\mathbf{R} \in SO(3)$

$$\mathbf{R}^{-1} = \mathbf{R}^T \tag{21}$$

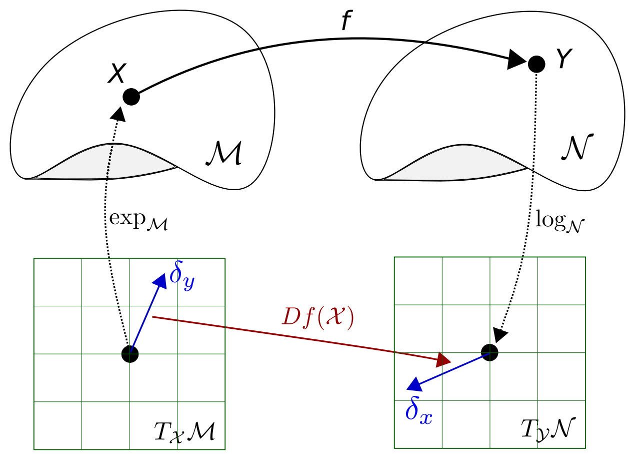

Figure 1. Illustration of the differential between two Lie groups.

for $\mathbf{G} = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix} \in SE(3)$

$$\mathbf{G}^{-1} = \begin{pmatrix} \mathbf{R}^T & -\mathbf{R}^T\mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix} \tag{22}$$

and for $\mathbf{T} = \begin{pmatrix} s\mathbf{R} & \mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix} \in Sim(3)$ we have

$$\mathbf{T}^{-1} = \begin{pmatrix} s^{-1}\mathbf{R}^T & -s^{-1}\mathbf{R}^T\mathbf{t} \\ \mathbf{0} & 1 \end{pmatrix} \tag{23}$$

2. Differentials and Jacobians

In the main paper, we derived the gradients for group multiplication. Here we provide derivations of the gradients for the remaining operators

Group Inverse: Using the definition of the differential

$$Df(X)[\mathbf{v}] = \lim_{t \to 0} \frac{\mathrm{Log}((e^{t\mathbf{v}}X)^{-1}(X^{-1})^{-1})}{t} \tag{24}$$

$$= \lim_{t \to 0} \frac{\mathrm{Log}(X^{-1}e^{-t\mathbf{v}}X)}{t} \tag{25}$$

using the adjoint

$$= \lim_{t \to 0} \frac{\mathrm{Log}(\mathrm{Exp}(-\mathrm{Adj}_{X^{-1}}(t\mathbf{v}))X^{-1}X)}{t} \tag{26}$$

$$= \lim_{t \to 0} \frac{-\mathrm{Adj}_{X^{-1}}\, t\mathbf{v}}{t} = -\mathrm{Adj}_{X^{-1}}\,\mathbf{v} \tag{27}$$

This can be used to recover the Jacobian $-\mathrm{Adj}_{X^{-1}}$.

Action on a Point: We can use elements from the 3D transformation groups to transform a 3D point or set of points. Given a homogeneous point $p = (X, Y, Z, 1)^T$ we can transform p using a transformation X

$$\mathbf{p}' = X\mathbf{p} \tag{28}$$

To make the notation consistent for all groups, a rotation can be represented as the 4×4 matrix $X = \begin{pmatrix} \mathbf{R} & 0 \\ 0 & 1 \end{pmatrix}$. X is a linear operator on p, so the Jacobian is simply the matrix representation of X

$$\frac{\partial \mathbf{p}'}{\partial \mathbf{p}} = X \tag{29}$$

We can also get the differential with respect to the transformation

$$Df(X)[\mathbf{v}] = \frac{d}{dt}(e^{t\mathbf{v}}X\mathbf{p})\Big|_{t=0} = \frac{d}{dt}(e^{t\mathbf{v}}\mathbf{p}')\Big|_{t=0} = \mathbf{v}^\wedge\mathbf{p}' \tag{30}$$

Adjoint: We consider the adjoint as the function $Adj : \mathcal{G} \times \mathfrak{g} \mapsto \mathfrak{g}$, $Adj_X(\boldsymbol{\omega}) = \boldsymbol{v}$. We need the Jacobians with respect to both X and ω. Since the adjoint is a linear map in terms of v then

$$\frac{\partial \boldsymbol{v}}{\partial \boldsymbol{\omega}} = \mathrm{Adj}_X, \qquad \frac{\partial L}{\partial \omega} = \frac{\partial L}{\partial v}\mathrm{Adj}_X \tag{31}$$

where $\mathrm{Adj}_X \in \mathbb{R}^{6\times 6}$ is the matrix representation of the adjoint. The gradient with respect to X can be found

$$D\,\mathrm{Adj}_X(\boldsymbol{\omega})[\mathbf{v}] = \frac{\partial}{\partial t}\left(e^{t\mathbf{v}}X\boldsymbol{\omega}^\wedge(e^{t\mathbf{v}}X)^{-1}\right)^\vee\Big|_{t=0} \tag{32}$$

$$= \frac{\partial}{\partial t}\left(e^{t\mathbf{v}}X\boldsymbol{\omega}^\wedge X^{-1}e^{-t\mathbf{v}}\right)^\vee\Big|_{t=0} \tag{33}$$

$$= \frac{\partial}{\partial t}\left(e^{t\mathbf{v}}\boldsymbol{v}^\wedge e^{-t\mathbf{v}}\right)^\vee\Big|_{t=0} \tag{34}$$

$$= \left(\frac{\partial}{\partial t}e^{t\mathbf{v}}\boldsymbol{v}^\wedge e^{-t\mathbf{v}}\right)^\vee\Big|_{t=0} \tag{35}$$

$$= (\mathbf{v}^\wedge\boldsymbol{v}^\wedge - \boldsymbol{v}^\wedge\mathbf{v}^\wedge)^\vee = \boldsymbol{v}^\curlywedge\mathbf{v} \tag{36}$$

Where $\curlywedge$ is defined in Sec 1. We note that the differential is linear in $\mathbf{v}$ allowing us to write the Jacobian and gradients as

$$\frac{\partial \boldsymbol{v}}{\partial X} = \boldsymbol{v}^\curlywedge, \qquad \frac{\partial L}{\partial X} = \frac{\partial L}{\partial v}\boldsymbol{v}^\curlywedge \tag{37}$$

Exponential and Logarithm Maps: The Jacobian of the exponential map $\mathbf{J}_l = \frac{\partial}{\partial \mathbf{x}}\mathrm{Exp}(\mathbf{x})$ is referred to as the left-Jacobian and can be written using the series [1] (page 235)

$$\mathbf{J}_l(\phi) = \sum_{n=0}^{\infty}\frac{1}{(n+1)!}(\phi^\curlywedge)^n \tag{38}$$

For $SO(3)$ and $SE(3)$ closed form expressions exist for Eqn. 38, otherwise we use the first 3 terms.

The Jacobian of the logarithm map $\mathbf{J}_l^{-1} = \frac{\partial}{\partial X}\mathrm{Log}(X)$ and can be computed using the series

$$\mathbf{J}_l^{-1}(\phi) = \sum_{n=0}^{\infty}\frac{B_n}{n!}(\phi^\curlywedge)^n \tag{39}$$

where B_n are the Bernoulli numbers [1](page 234). Again, we used analytic expressions of $\mathbf{J}_l^{-1}$ for $SO(3)$ and $SE(3)$, and the first 3 terms for $Sim(3)$.

3. Sim(3) Network Architecture

A overview of the Sim(3) network architecture is shown in Fig. 2. The context and feature encoders are identical to RAFT. We replace the 5×1, 1×5 GRU used in RAFT with a single 3×3 convolutional GRU, using a hidden state size of 128 channels. We apply 12 iterations during both training and testing.

References

[1] Timothy D Barfoot. *State estimation for robotics*. Cambridge University Press, 2017. 1, 4

[2] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. *arXiv preprint arXiv:2003.12039*, 2020. 5

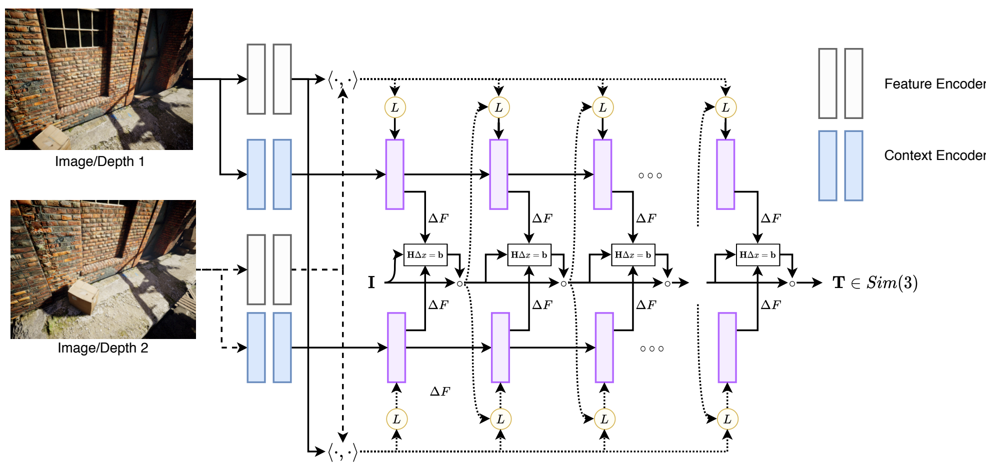

Figure 2. Network architecture used for our Sim(3) registration experiments. The network architecture is based on RAFT[2]. The top branch estimates motion from $I_1 \rightarrow I_2$ and the bottom branch estimates motion in the opposite direction $I_2 \rightarrow I_1$. Features are first extracted from each of the two input images and used to construct two 4D correlation volumes, which are pooled at multiple resolutions according to RAFT. During each iteration, the current estimate of the transformation $\mathbf{T}$ is used to index from each of the correlation volumes. The correlation features are processed by the GRU which outputs a residual flow estimate (optical flow not explained by the current transformation $\mathbf{T}$. Both bidirectional residual flow estimates are used as input to a optimization layer, which unrolls 3 Gauss-Newton iterations to produced a transformation update $\Delta\mathbf{T}$, which is applied to the current transformation estimate.