Lie Groups for Beginners

Frank Dellaert

March 4, 2010

1 Motivation: Rigid Motions in the Plane

IMAGINE A FIGURE HERE

Figure 1: (a) A robot translating. (b) A robot rotating.

We will start with a small example of a robot moving in a plane, parameterized by a *2D pose* (x,y,θ). When we give it a small forward velocity v_x, we know that the location changes as

$$\dot{x} = v_x$$

The solution to this trivial differential equation is, with x_0 the initial x-position f the robot,

$$x = x_0 + v_x t$$

A similar story holds for translation in the y direction, and in fact for translations in general:

$$(x,y,\theta) = (x_0 + v_x t, y_0 + v_y t, \theta_0)$$

Similarly for rotation we have

$$(x,y,\theta) = (x_0, y_0, \theta_0 + \omega t)$$

where ω is angular velocity, measured in rad/s in counterclockwise direction.

IMAGINE A FIGURE HERE

Figure 2: Robot moving along a circular trajectory.

However, if we combine translation and rotation, the story breaks down! We cannot write

$$(x,y,\theta) = (x_0 + v_x t, y_0 + v_y t, \theta_0 + \omega t)$$

The reason is that, if we move the robot a tiny bit according to the velocity vector (v_x, v_y, ω), we do have to first order

$$(x_{t+\delta}, y_{t+\delta}, \theta_{t+\delta}) = (x_0 + v_x\delta, y_0 + v_y\delta, \theta_0 + \omega\delta)$$

but now the robot has rotated, and for the next incremental change, the velocity vector would have to be rotated before it can be applied. In fact, the robot will move on a *circular* trajectory.

The reason is that *translation and rotation do not commute*: if we rotate and then move we will end up in a different place than if we moved first, then rotated. In fact, someone once said (I forget who, kudos for who can track down the exact quote):

> If rotation and translation commuted, we could do all rotations before leaving home.

To make progress, we have to be more precise about how the robot behaves. Specifically, let us define composition of two poses T_1 and T_2 as

$$T_1 T_2 = (x_1, y_1, \theta_1)(x_2, y_2, \theta_2) = (x_1 + \cos\theta_1 x_2 - \sin\theta y_2, y_1 + \sin\theta_1 x_2 + \cos\theta_1 y_2, \theta_1 + \theta_2)$$

This is a bit clumsy, so we resort to a trick: embed the 2D poses in the 3×3 matrices, so we can define composition as matrix multiplication:

$$T_1 T_2 = \begin{bmatrix} R_1 & t_1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_2 & t_2 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R_1 R_2 & R_1 t_2 + t_1 \\ 0 & 1 \end{bmatrix}$$

where the matrices R are 2D rotation matrices defined as

$$R = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

IMAGINE A FIGURE HERE

Figure 3: Approximating a circular trajectory with n steps.

Now, a “tiny” motion of the robot can be written as

$$T(\delta) = \begin{bmatrix} \cos\omega\delta & -\sin\omega\delta & v_x\delta \\ \sin\omega\delta & \cos\omega\delta & v_y\delta \\ 0 & 0 & 1 \end{bmatrix} \approx \begin{bmatrix} 1 & -\omega\delta & v_x\delta \\ \omega\delta & 1 & v_y\delta \\ 0 & 0 & 1 \end{bmatrix} = I + \delta \begin{bmatrix} 0 & -\omega & v_x \\ \omega & 0 & v_y \\ 0 & 0 & 1 \end{bmatrix}$$

Let us define the *2D twist* vector $\xi = (v, \omega)$, and the matrix above as

$$\hat{\xi} \stackrel{\Delta}{=} \begin{bmatrix} 0 & -\omega & v_x \\ \omega & 0 & v_y \\ 0 & 0 & 1 \end{bmatrix}$$

If we wanted t to be large, we could split up t into smaller timesteps, say n of them, and compose them as follows:

$$T(t) \approx \left(I + \frac{t}{n}\hat{\xi}\right) \ldots \text{n times} \ldots \left(I + \frac{t}{n}\hat{\xi}\right) = \left(I + \frac{t}{n}\hat{\xi}\right)^n$$

The result is shown in Figure 3.

Of course, the perfect solution would be obtained if we take n to infinity:

$$T(t) = \lim_{n\to\infty} \left(I + \frac{t}{n}\hat{\xi}\right)^n$$

For real numbers, this series is familiar and is actually a way to compute the exponential function:

$$e^x = \lim_{n\to\infty}\left(I + \frac{x}{n}\right)^n = \sum_{k=0}^{\infty}\frac{x^k}{k!}$$

The series can be similarly defined for square matrices,

$$e^A = \lim_{n\to\infty}\left(I + \frac{A}{n}\right)^n = \sum_{k=0}^{\infty}\frac{A^k}{k!}$$

Our final result is that we can write the motion of a robot along a circular trajectory, resulting from the 2D twist $\xi = (v, \omega)$ as the matrix exponential of $\hat{\xi}$:

$$T(t) = e^{t\hat{\xi}} \triangleq \lim_{n\to\infty}\left(I + \frac{t}{n}\hat{\xi}\right)^n = \sum_{k=0}^{\infty}\frac{\left(t\hat{\xi}\right)^k}{k!}$$

We call this mapping from 2D twists matrices $\hat{\xi}$ to 2D rigid transformations the *exponential map.*

The above has all elemtns of Lie group theory. We call the space of 2D rigid transformations, along with the composition operation, the *special Euclidean group* $SE(2)$. It is called a Lie group because it is both a manifold, and its group operation is smooth when operating on this manifold. The space of 2D twists, together with a special binary operation to be defined below, is called the Lie algebra $\mathfrak{se}(2)$ associated with $SE(2)$. Below we generalize these concepts and then introduce the most commonly used Lie groups and their Lie algebras.

2 Basic Lie Group Concepts

2.1 A Manifold and a Group

A Lie group G is a manifold that possesses a smooth group operation. Associated with it is a Lie Algebra $\mathfrak{g}$ which, loosely speaking, can be identified with the tangent space at the identity and completely defines how the groups behaves around the identity. There is a mapping from $\mathfrak{g}$ back to G, called the exponential map

$$\exp : \mathfrak{g} \rightarrow G$$

which is typically a many-to-one mapping. The corresponding inverse can be define locally around the origin and hence is a "logarithm":

$$\log : G \rightarrow \mathfrak{g}$$

that maps elements in a neighborhood of *id* in G to an element in $\mathfrak{g}$.

2.2 Lie Algebra

The Lie Algebra $\mathfrak{g}$ is called an algebra because it is endowed with a binary operation, the Lie bracket $[X,Y]$, the properties of which are closely related to the group operation of G. For example, in matrix Lie groups, the Lie bracket is given by $[A,B] \stackrel{\Delta}{=} AB - BA$. The relationship with the group operation is as follows: for commutative Lie groups vector addition $X+Y$ in $\mathfrak{g}$ mimicks the group operation. For example, if we have $Z = X+Y$ in $\mathfrak{g}$, when mapped backed to G via the exponential map we obtain

$$e^Z = e^{X+Y} = e^X e^Y$$

However, this does *not* hold for non-commutative Lie groups:

$$Z = \log(e^X e^Y) \neq X+Y$$

Instead, Z can be calculated using the Baker-Campbell-Hausdorff (BCH) formula:[1]

$$Z = X+Y+[X,Y]/2+[X-Y,[X,Y]]/12-[Y,[X,[X,Y]]]/24+\dots$$

For commutative groups the bracket is zero and we recover $Z = X+Y$. For non-commutative groups we can use the BCH formula to approximate it.

2.3 Exponential Coordinates

For n-dimensional matrix Lie groups, the Lie algebra $\mathfrak{g}$ is isomorphic to $\mathbb{R}^n$, and we can define the wedge operator [3, page 41],

$$\hat{} : x \in \mathbb{R}^n \rightarrow \hat{x} \in \mathfrak{g}$$

which maps n-vectors $x \in \mathbb{R}^n$ to elements of $\mathfrak{g}$. In the case of matrix Lie groups, the elements $\hat{x}$ of $\mathfrak{g}$ are $n \times n$ matrices, and the map is given by

$$\hat{x} = \sum_{i=1}^{n} x_i G^i \tag{1}$$

[1] http://en.wikipedia.org/wiki/Baker–Campbell–Hausdorff_formula

where the G^i are $n \times n$ matrices known as the Lie group generators. The meaning of the map $x \longrightarrow \hat{x}$ will depend on the group G and will generally have an intuitive interpretation.

2.4 The Adjoint Map

Below we frequently make use of the equality[2]

$$ge^{\hat{x}}g^{-1} = e^{Ad_g \hat{x}}$$

where $Ad_g : \mathfrak{g} \longrightarrow \mathfrak{g}$ is a map parameterized by a group element g. The intuitive explanation is that a change $\exp(\hat{x})$ defined around the orgin, but applied at the group element g, can be written in one step by taking the adjoint $Ad_g \hat{x}$ of $\hat{x}$. In the case of a matrix group the ajoint can be written as [3]

$$Ad_T \hat{x} \stackrel{\Delta}{=} T\hat{x}T^{-1}$$

and hence we have

$$Te^{\hat{x}}T^{-1} = e^{T\hat{x}T^{-1}}$$

where both $T \in G$ and $\hat{x} \in \mathfrak{g}$ are $n \times n$ matrices for an n-dimensional Lie group.

2.5 Actions

The (usual) action of an n-dimensional matrix group G is matrix-vector multiplication on $\mathbb{R}^n$,

$$q = Tp$$

with $p, q \in \mathbb{R}^n$ and $T \in GL(n)$.

[2] http://en.wikipedia.org/wiki/Exponential_map

[3] http://en.wikipedia.org/wiki/Adjoint_representation_of_a_Lie_group

3 2D Rotations

We first look at a very simple group, the 2D rotations.

3.1 Basics

The Lie group $SO(2)$ is a subgroup of the general linear group $GL(2)$ of 2×2 invertible matrices. Its Lie algebra $\mathfrak{so}(2)$ is the vector space of 2×2 skew-symmetric matrices. Since $SO(2)$ is a one-dimensional manifold, $\mathfrak{so}(2)$ is isomorphic to $\mathbb{R}$ and we define

$$\hat{}: \mathbb{R} \to \mathfrak{so}(2)$$
$$\hat{}: \theta \to \hat{\theta} = [\theta]_+$$

which maps the angle θ to the 2×2 skew-symmetric matrix $[\theta]_+$:

$$[\theta]_+ = \left[\begin{array}{cc} 0 & -\theta \\ \theta & 0 \end{array}\right]$$

The exponential map can be computed in closed form as

$$R = e^{[\theta]_+} = \left[\begin{array}{cc} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{array}\right]$$

3.2 Adjoint

The adjoint map for $\mathfrak{so}(2)$ is trivially equal to the identity, as is the case for *all* commutative groups:

$$\begin{aligned} Ad_R\hat{\omega} &= \left[\begin{array}{cc} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{array}\right]\left[\begin{array}{cc} 0 & -\omega \\ \omega & 0 \end{array}\right]\left[\begin{array}{cc} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{array}\right]^T \\ &= \omega\left[\begin{array}{cc} -\sin\theta & -\cos\theta \\ \cos\theta & -\sin\theta \end{array}\right]\left[\begin{array}{cc} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{array}\right] = \left[\begin{array}{cc} 0 & -\omega \\ \omega & 0 \end{array}\right] \end{aligned}$$

which we can write in terms of ω as

$$Ad_R\omega = \omega$$

3.3 Actions

In the case of $SO(2)$ the vector space is $\mathbb{R}^2$, and the group action corresponds to rotating a point

$$q = Rp$$

We would now like to know what an incremental rotation parameterized by θ would do:

$$q(\theta) = Re^{[\theta]_+}p$$

hence the derivative is:

$$\frac{\partial q(\omega)}{\partial\omega} = R\frac{\partial}{\partial\omega}\left(e^{[\theta]_+}p\right) = R\frac{\partial}{\partial\omega}([\theta]_+p) = RH_p$$

Note that

$$[\theta]_+\left[\begin{array}{c} x \\ y \end{array}\right] = \theta R_{\pi/2}\left[\begin{array}{c} x \\ y \end{array}\right] = \theta\left[\begin{array}{c} -y \\ x \end{array}\right] \tag{2}$$

which acts like a restricted “cross product” in the plane.

4 2D Rigid Transformations

4.1 Basics

The Lie group $SE(2)$ is a subgroup of the general linear group $GL(3)$ of 3×3 invertible matrices of the form

$$T \triangleq \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix}$$

where $R \in SO(2)$ is a rotation matrix and $t \in \mathbb{R}^2$ is a translation vector. $SE(2)$ is the *semi-direct product* of $\mathbb{R}^2$ by $SO(2)$, written as $SE(2) = \mathbb{R}^2 \rtimes SO(2)$. In particular, any element T of $SE(2)$ can be written as

$$T = \begin{bmatrix} 0 & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R & 0 \\ 0 & k \end{bmatrix}$$

and they compose as

$$T_1 T_2 = \begin{bmatrix} R_1 & t_1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R_2 & t_2 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R_1 R_2 & R_1 t_2 + t_1 \\ 0 & 1 \end{bmatrix}$$

Hence, an alternative way of writing down elements of $SE(2)$ is as the ordered pair (R,t), with composition defined a

$$(R_1, t_1)(R_2, t_2) = (R_1 R_2, R_1 t_2 + t_1)$$

The corresponding Lie algebra $\mathfrak{se}(2)$ is the vector space of 3×3 twists $\hat{\xi}$ parameterized by the *twist coordinates* $\xi \in \mathbb{R}^3$, with the mapping

$$\xi \triangleq \begin{bmatrix} v \\ \omega \end{bmatrix} \rightarrow \hat{\xi} \triangleq \begin{bmatrix} [\omega]_+ & v \\ 0 & 0 \end{bmatrix}$$

Note we think of robots as having a pose (x, y, θ) and hence I reserved the first two components for translation and the last for rotation. The corresponding Lie group generators are

$$G^x = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad G^y = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad G^\theta = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Applying the exponential map to a twist ξ yields a screw motion yielding an element in $SE(2)$:

$$T = e^{\hat{\xi}} = \left(e^{[\omega]_+}, (I - e^{[\omega]_+}) \frac{v^\perp}{\omega} \right)$$

4.2 The Adjoint Map

The adjoint is

$$\begin{aligned} Ad_T \hat{\xi} &= T \hat{\xi} T^{-1} \\ &= \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} [\omega]_+ & v \\ 0 & 0 \end{bmatrix} \begin{bmatrix} R^T & -R^T t \\ 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} [\omega]_+ & -[\omega]_+ t + R v \\ 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} [\omega]_+ & R v - \omega R_{\pi/2} t \\ 0 & 0 \end{bmatrix} \end{aligned} \tag{3}$$

From this we can express the Adjoint map in terms of plane twist coordinates:

$$\begin{bmatrix} v' \\ \omega' \end{bmatrix} = \begin{bmatrix} R & -R_{\pi/2}t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}$$

4.3 Actions

The action of $SE(2)$ on 2D points is done by embedding the points in $\mathbb{R}^3$ by using homogeneous coordinates

$$\hat{q} = \begin{bmatrix} q \\ 1 \end{bmatrix} = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p \\ 1 \end{bmatrix} = T\hat{p}$$

Analoguous to $SE(3)$, we can compute a velocity $\hat{\xi}\hat{p}$ in the local T frame:

$$\hat{\xi}\hat{p} = \begin{bmatrix} [\omega]_+ & v \\ 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ 1 \end{bmatrix} = \begin{bmatrix} [\omega]_+ p + v \\ 0 \end{bmatrix}$$

By only taking the top two rows, we can write this as a velocity in $\mathbb{R}^2$, as the product of a 2×3 matrix H_p that acts upon the exponential coordinates ξ directly:

$$[\omega]_+ p + v = v + R_{\pi/2} p \omega = \begin{bmatrix} I_2 & R_{\pi/2} p \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} = H_p \xi$$

5 3D Rotations

5.1 Basics

The Lie group $SO(3)$ is a subgroup of the general linear group $GL(3)$ of 3×3 invertible matrices. Its Lie algebra $\mathfrak{so}(3)$ is the vector space of 3×3 skew-symmetric matrices $\hat{\omega}$. The exponential map can be computed in closed form using Rodrigues' formula [3, page 28]:

$$e^{\hat{\omega}} = I + \hat{\omega}\sin\theta + \hat{\omega}^2(1 - \cos\theta)$$

where $\hat{\omega}^2 = \omega\omega^T - I$, with $\omega\omega^T$ the outer product of ω. Hence, a slightly more efficient variant is

$$e^{\hat{\omega}} = \cos\theta I + \hat{\omega} sin\theta + \omega\omega^T(1 - cos\theta)$$

Since $SO(3)$ is a three-dimensional manifold, $\mathfrak{so}(3)$ is isomorphic to $\mathbb{R}^3$ and we define the map

$$\hat{}: \mathbb{R}^3 \to \mathfrak{so}(3)$$

$$\hat{}: \omega \to \hat{\omega} = [\omega]_\times$$

which maps 3-vectors ω to skew-symmetric matrices $[\omega]_\times$:

$$[\omega]_\times = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} = \omega_x G^x + \omega_y G^y + \omega_z G^z$$

where the G^i are the generators for $SO(3)$,

$$G^x = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix} G^y = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix} G^z = \begin{pmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

corresponding to a rotation around X, Y, and Z, respectively. The Lie bracket $[x, y]$ corresponds to the cross product $x \times y$ in $\mathbb{R}^3$.

For every $3-$vector ω there is a corresponding rotation matrix

$$R = e^{[\omega]_\times}$$

and this is defines the canonical parameterization of $SO(3)$, with ω known as the canonical or exponential coordinates. It is equivalent to the axis-angle representation for rotations, where the unit vector $\omega / \|\omega\|$ defines the rotation axis, and its magnitude the amount of rotation θ.

5.2 The Adjoint Map

For rotation matrices R we can prove the following identity (see 5 on page 15):

$$R[\omega]_\times R^T = [R\omega]_\times \tag{4}$$

Hence, given property (5), the adjoint map for $\mathfrak{so}(3)$ simplifies to

$$Ad_R[\omega]_\times = R[\omega]_\times R^T = [R\omega]_\times$$

and this can be expressed in exponential coordinates simply by rotating the axis ω to $R\omega$.

As an example, to apply an axis-angle rotation ω to a point p in the frame R, we could:

1. First transform p back to the world frame, apply ω, and then rotate back:

$$q = Re^{[\omega]_\times} R^T$$

2. Immediately apply the transformed axis-angle transformation $Ad_R[\omega]_\times = [R\omega]_\times$:

$$q = e^{[R\omega]_\times} p$$

5.3 Actions

In the case of $SO(3)$ the vector space is $\mathbb{R}^3$, and the group action corresponds to rotating a point

$$q = Rp$$

We would now like to know what an incremental rotation parameterized by ω would do:

$$q(\omega) = Re^{[\omega]_\times} p$$

hence the derivative is:

$$\frac{\partial q(\omega)}{\partial \omega} = R \frac{\partial}{\partial \omega} \left(e^{[\omega]_\times} p \right) = R \frac{\partial}{\partial \omega} \left([\omega]_\times p \right) = RH_p$$

To calculate H_p we make use of

$$[\omega]_\times p = \omega \times p = -p \times \omega = [-p]_\times \omega$$

6 3D Rigid Transformations

The Lie group $SE(3)$ is a subgroup of the general linear group $GL(4)$ of 4×4 invertible matrices of the form

$$T\stackrel{\Delta}{=}\left[\begin{array}{cc}R & t\\0 & 1\end{array}\right]$$

where $R\in SO(3)$ is a rotation matrix and $t\in\mathbb{R}^3$ is a translation vector. An alternative way of writing down elements of $SE(3)$ is as the ordered pair (R,t), with composition defined as

$$(R_1,t_1)(R_2,t_2)=(R_1R_2,R_1t_2+t_1)$$

Its Lie algebra $\mathfrak{se}(3)$ is the vector space of 4×4 twists $\hat{\xi}$ parameterized by the *twist coordinates* $\xi\in\mathbb{R}^6$, with the mapping [3]

$$\xi\stackrel{\Delta}{=}\left[\begin{array}{c}\omega\\v\end{array}\right]\rightarrow\hat{\xi}\stackrel{\Delta}{=}\left[\begin{array}{cc}[\omega]_\times & v\\0 & 0\end{array}\right]$$

Note we follow Frank Park's convention and reserve the first three components for rotation, and the last three for translation. Hence, with this parameterization, the generators for $SE(3)$ are

$$G^1=\begin{pmatrix}0&0&0&0\\0&0&-1&0\\0&1&0&0\\0&0&0&0\end{pmatrix}G^2=\begin{pmatrix}0&0&1&0\\0&0&0&0\\-1&0&0&0\\0&0&0&0\end{pmatrix}G^3=\begin{pmatrix}0&-1&0&0\\1&0&0&0\\0&0&0&0\\0&0&0&0\end{pmatrix}$$

$$G^4=\begin{pmatrix}0&0&0&1\\0&0&0&0\\0&0&0&0\\0&0&0&0\end{pmatrix}G^5=\begin{pmatrix}0&0&0&0\\0&0&0&1\\0&0&0&0\\0&0&0&0\end{pmatrix}G^6=\begin{pmatrix}0&0&0&0\\0&0&0&0\\0&0&0&1\\0&0&0&0\end{pmatrix}$$

Applying the exponential map to a twist ξ yields a screw motion yielding an element in $SE(3)$:

$$T=\exp\hat{\xi}$$

A closed form solution for the exponential map is given in [3, page 42].

$$\exp\left(\left[\begin{array}{c}\omega\\v\end{array}\right]t\right)=\left[\begin{array}{cc}e^{[\omega]_\times t} & (I-e^{[\omega]_\times t})(\omega\times v)+\omega\omega^Tvt\\0 & 1\end{array}\right]$$

6.1 The Adjoint Map

The adjoint is

$$\begin{aligned}Ad_T\hat{\xi} &= T\hat{\xi}T^{-1}\\&=\left[\begin{array}{cc}R & t\\0 & 1\end{array}\right]\left[\begin{array}{cc}[\omega]_\times & v\\0 & 0\end{array}\right]\left[\begin{array}{cc}R^T & -R^Tt\\0 & 1\end{array}\right]\\&=\left[\begin{array}{cc}[R\omega]_\times & -[R\omega]_\times t+Rv\\0 & 0\end{array}\right]\\&=\left[\begin{array}{cc}[R\omega]_\times & t\times R\omega+Rv\\0 & 0\end{array}\right]\end{aligned}$$

From this we can express the Adjoint map in terms of twist coordinates (see also [3] and FP):

$$\begin{bmatrix} \omega' \\ v' \end{bmatrix} = \begin{bmatrix} R & 0 \\ [t]_\times R & R \end{bmatrix} \begin{bmatrix} \omega \\ v \end{bmatrix}$$

6.2 Actions

The action of $SE(3)$ on 3D points is done by embedding the points in $\mathbb{R}^4$ by using homogeneous coordinates

$$\hat{q} = \begin{bmatrix} q \\ 1 \end{bmatrix} = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} p \\ 1 \end{bmatrix} = T\hat{p}$$

We would now like to know what an incremental rotation parameterized by ξ would do:

$$\hat{q}(\xi) = Te^{\hat{\xi}}\hat{p}$$

hence the derivative (following the exposition in Section **??**):

$$\frac{\partial \hat{q}(\xi)}{\partial \xi} = T\frac{\partial}{\partial \xi}\left(\hat{\xi}\hat{p}\right) = TH_p$$

where $\hat{\xi}\hat{p}$ corresponds to a velocity in $\mathbb{R}^4$ (in the local T frame):

$$\hat{\xi}\hat{p} = \begin{bmatrix} [\omega]_\times & v \\ 0 & 0 \end{bmatrix} \begin{bmatrix} p \\ 1 \end{bmatrix} = \begin{bmatrix} \omega \times p + v \\ 0 \end{bmatrix}$$

Notice how velocities are anologous to points at infinity in projective geometry: they correspond to free vectors indicating a direction and magnitude of change.

By only taking the top three rows, we can write this as a velocity in $\mathbb{R}^3$, as the product of a 3×6 matrix H_p that acts upon the exponential coordinates ξ directly:

$$\omega \times p + v = -p \times \omega + v = \begin{bmatrix} -[p]_\times & I_3 \end{bmatrix} \begin{bmatrix} \omega \\ v \end{bmatrix} = H_p\xi$$

7 2D Affine Transformations

The Lie group $Aff(2)$ is a subgroup of the general linear group $GL(3)$ of 3×3 invertible matrices that maps the line infinity to itself, and hence preserves paralellism. The affine transformation matrices A can be written as [2]

$$\begin{bmatrix} m_{11} & m_{12} & t_1 \\ m_{21} & m_{22} & t_2 \\ 0 & 0 & k \end{bmatrix}$$

with $M \in GL(2)$, $t \in \mathbb{R}^2$, and k a scalar chosen such that $det(A) = 1$. Note that just as $SE(2)$ is a semi-direct product, so too is $Aff(2) = \mathbb{R}^2 \rtimes GL(2)$. In particular, any affine transformation A can be written as

$$A = \begin{bmatrix} 0 & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} M & 0 \\ 0 & k \end{bmatrix}$$

and they compose as

$$A_1 A_2 = \begin{bmatrix} M_1 & t_1 \\ 0 & k_1 \end{bmatrix} \begin{bmatrix} M_2 & t_2 \\ 0 & k_2 \end{bmatrix} = \begin{bmatrix} M_1 M_2 & M_2 t_2 + k_2 t_1 \\ 0 & k_1 k_2 \end{bmatrix}$$

From this it can be gleaned that the groups $SO(2)$ and $SE(2)$ are both subgroups, with $SO(2) \subset SE(2) \subset Aff(2)$. By choosing the generators carefully we maintain this subgroup hierarchy. In particular, $\mathfrak{se}(2)$

$$G^1 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \; G^2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \; G^3 = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

can be extended to the Lie algebra $\mathfrak{aff}(2)$ using the three additional generators

$$G^4 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \; G^5 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \; G^6 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Hence, the Lie algebra $\mathfrak{aff}(2)$ is the vector space of 3×3 incremental affine transformations $\hat{a}$ parameterized by 6 parameters $a \in \mathbb{R}^6$, with the mapping

$$a \to \hat{a} \stackrel{\Delta}{=} \begin{bmatrix} a_5 & a_4 - a_3 & a_1 \\ a_4 + a_3 & -a_5 - a_6 & a_2 \\ 0 & 0 & a_6 \end{bmatrix}$$

Note that G_5 and G_6 change the relative scale of x and y but without changing the determinant:

$$e^{xG_5} = \exp\left(\begin{bmatrix} x & 0 & 0 \\ 0 & -x & 0 \\ 0 & 0 & 0 \end{bmatrix} \right) = \begin{bmatrix} e^x & 0 & 0 \\ 0 & 1/e^x & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$e^{xG_6} = \exp\left(\begin{bmatrix} 0 & 0 & 0 \\ 0 & -x & 0 \\ 0 & 0 & x \end{bmatrix} \right) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1/e^x & 0 \\ 0 & 0 & e^x \end{bmatrix}$$

It might be nicer to have the correspondence with scaling x and y more direct, by choosing

$$G^5 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -1 \end{bmatrix} \quad G^6 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

and hence

$$e^{xG_5} = \exp\left(\begin{bmatrix} x & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & -x \end{bmatrix}\right) = \begin{bmatrix} e^x & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1/e^x \end{bmatrix}$$

$$e^{xG_6} = \exp\left(\begin{bmatrix} 0 & 0 & 0 \\ 0 & x & 0 \\ 0 & 0 & -x \end{bmatrix}\right) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & e^x & 0 \\ 0 & 0 & 1/e^x \end{bmatrix}$$

8 2D Homographies

When viewed as operations on images, represented by 2D projective space $\mathscr{P}^3$, 3D rotations are a special case of 2D homographies. These are now treated, loosely based on the exposition in [1, 2].

8.1 Basics

The Lie group $SL(3)$ is a subgroup of the general linear group $GL(3)$ of 3×3 invertible matrices with determinant 1. The homographies generalize transformations of the 2D projective space, and $Aff(2) \subset SL(3)$.

We can extend $\mathfrak{aff}(2)$ to the Lie algebra $\mathfrak{sl}(3)$ by adding two generators

$$G^7 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad G^8 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

obtaining the vector space of 3×3 incremental homographies $\hat{h}$ parameterized by 8 parameters $h \in \mathbb{R}^8$, with the mapping

$$h \to \hat{h} \stackrel{\Delta}{=} \begin{bmatrix} h_5 & h_4 - h_3 & h_1 \\ h_4 + h_3 & -h_5 - h_6 & h_2 \\ h_7 & h_8 & h_6 \end{bmatrix}$$

8.2 Tensor Notation

- A homography between 2D projective spaces A and B can be written in tensor notation H_A^B
- Applying a homography is then a tensor contraction $x^B = H_A^B x^A$, mapping points in A to points in B.

Appendix: Proof of Property 5

We can prove the following identity for rotation matrices R,

$$\begin{aligned}
R[\omega]_\times R^T &= R[\omega]_\times \left[\begin{array}{ccc} a_1 & a_2 & a_3 \end{array}\right] \\
&= R\left[\begin{array}{ccc} \omega\times a_1 & \omega\times a_2 & \omega\times a_3 \end{array}\right] \\
&= \left[\begin{array}{ccc} a_1(\omega\times a_1) & a_1(\omega\times a_2) & a_1(\omega\times a_3) \\ a_2(\omega\times a_1) & a_2(\omega\times a_2) & a_2(\omega\times a_3) \\ a_3(\omega\times a_1) & a_3(\omega\times a_2) & a_3(\omega\times a_3) \end{array}\right] \\
&= \left[\begin{array}{ccc} \omega(a_1\times a_1) & \omega(a_2\times a_1) & \omega(a_3\times a_1) \\ \omega(a_1\times a_2) & \omega(a_2\times a_2) & \omega(a_3\times a_2) \\ \omega(a_1\times a_3) & \omega(a_2\times a_3) & \omega(a_3\times a_3) \end{array}\right] \\
&= \left[\begin{array}{ccc} 0 & -\omega a_3 & \omega a_2 \\ \omega a_3 & 0 & -\omega a_1 \\ -\omega a_2 & \omega a_1 & 0 \end{array}\right] \\
&= [R\omega]_\times
\end{aligned} \tag{5}$$

where a_1, a_2, and a_3 are the *rows* of R. Above we made use of the orthogonality of rotation matrices and the triple product rule:

$$a(b\times c) = b(c\times a) = c(a\times b)$$

Similarly, without proof [3, Lemma 2.3]:

$$R(a\times b) = Ra\times Rb$$

Appendix: Alternative Generators for $\mathfrak{sl}(3)$

[1] uses the following generators for $\mathfrak{sl}(3)$:

$$G^1 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \; G^2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \; G^3 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$G^4 = \begin{bmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \; G^5 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \; G^6 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$G^7 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \; G^8 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

We choose to use a different linear combination as the basis.

References

[1] C. Mei, S. Benhimane, E. Malis, and P. Rives. Homography-based tracking for central catadioptric cameras. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, October 2006.

[2] C. Mei, S. Benhimane, E. Malis, and P. Rives. Efficient homography-based tracking and 3-D reconstruction for single-viewpoint sensors. *IEEE Trans. Robotics*, 24(6):1352–1364, Dec. 2008.

[3] R.M. Murray, Z. Li, and S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.