# Lecture 13: Brief Group Conv Recap + Intro to Lie Groups

**Last time:** "Steerable" CNNs and Basis Functions

$$[f \star \psi]^{(\rho_i \otimes \rho_j)}(x) = \sum_{y \in \mathbb{Z}^2} f^{(\rho_i)}(y) \otimes \psi^{(\rho_j)}(y-x)$$

Define $L_g f^{(\rho_i)}(x) = D^{(\rho_i)}(g) f^{(\rho_i)}(gx)$

Equivariant if

$$\psi^{(\rho_j)}(g^{-1}x) = D^{(\rho_j)}(g)\, \psi^{(\rho_j)}(x)$$

definition of filters that are "steerable"

Note, if we change our definition of $L_g$ this will change the constraint

Weights must be scalars! Can be introduced in two ways:

→ matrix that commutes w/ representation of group

$$W^{(\rho_i)} \psi^{(\rho_j)}(g^{-1}x) = W^{(\rho_j)} D^{(\rho_j)}(g) \psi^{(\rho_j)}(x) \overset{?}{=} D^{(\rho_j)}(g) W^{(\rho_j)} \psi^{(\rho_j)}(x)$$

If $D^{(\rho_j)}$ direct sum of irreps such that it is block diagonal, only $W^{(\rho_j)}$ that would commute are constant matrices (scalars) between same irreps, Linear layer defined by Schur's Lemma.

→ Second argument to $\psi^{(\rho_j)}$

For $\psi^{(\rho_j)}(g^{-1}x, w) = D^{(\rho_j)}(g)\, \psi^{(\rho_j)}(x, w)$, $w$ must transform a scalar (trivial representation)

$D^{w}(g) = 1, \forall g \in G$

Then why did we have arbitrary filters for group convolution?

→ In group convolution, filters (and inputs and outputs) transform as the tensor product representation of pixels ⊗ reg rep.

→ In bottom of colab for Lecture 12, we derive "scalars" of this representation, and it exactly matches arbitrary 3x3 filters but then permuted according to the pixel ⊗ reg. rep tensor product representation.

→ The "change of basis" for this representation is simple! Its $(\mathbb{1}_{\text{pixel basis}} \otimes \text{reg rep})$.

→ Each of the $|G|$ "paths" in the reg rep $(h_j \to g)$ gets its own scalar, hence $|G|$ filter parameters, but crucially these parameters must be structured as the trivial rep of (pixel basis ⊗ reg. rep) tp representation.

More generally, we can add scalar weights to any rep in network

$$[f \star \psi]^{\rho_{out}}(x) = \sum_{y \in \mathbb{Z}^2} \underbrace{C^{\rho_{out}}_{\rho_{in},\rho_{filter}}}_{\substack{\text{Linear} \\ \text{Layer applied} \\ \text{to output}}} \underbrace{f^{\rho_{in}}(y)}_{\substack{\text{Linear layers} \\ \text{applied to input}}} \otimes \underbrace{\psi^{\rho_{filter}}(y-x)}_{\substack{\text{Linear layers} \\ \text{applied to filter}}}$$

e.g. Weights applied to basis functions

→ radii in case of irreps

→  ...

basis for reg rep ⊗ pixel

# Today: SO(3) and its Representations

## How do we deal with infinite groups? (Georgi Ch 2)

Intro to Lie Groups

Generally it's hard. But we'll deal with some particularly nice infinite groups called "Lie Groups", which are parameterized smoothly on a set of parameters.

$$g(\alpha) \quad \text{for } g \in G, \text{ parameters } \alpha$$

Similar $\alpha \rightarrow$ similar elements

examples: $O(n)$, $E(n)$

Just as we could "generate" a finite group from a well chosen subset of elements, Lie Groups have "infinitesimal generators" with which we can generate the entire group.

Like all groups, Lie Groups have an identity element

$$g(\alpha)\Big|_{\alpha=0} = e \text{, with representation}$$

$$D(\alpha)\Big|_{\alpha=0} = \mathbb{1}$$

Infinitesimally away from $\alpha = 0$ we can Taylor expand

$$D(d\alpha) = \mathbb{1} + (d\alpha_a) X_a + \ldots$$

(assuming einstein summation notation)

(in Georgi there's an extra "$i$" because physicist like Hermitian matrices)

we will deal w/ the real representation for SO(3)

$$\rightarrow X_a \equiv \frac{\partial}{\partial \alpha_a} D(\alpha)\Big|_{\alpha=0} \quad \text{for } a = 1, \ldots, N$$

$X_a$ are the (matrix) generators of the group. For not small $\alpha$:

$p=0$ … $p=\alpha$ — divide into $k$ pieces

$$D(d\alpha) = 1 + (d\alpha_a)X_a + \cancel{\frac{(d\alpha_a)^2 X_a^2}{2!}} \rightarrow D(\alpha) = \lim_{k \to \infty} \left(1 + \alpha_a X_a / k\right)^k = e^{\alpha_a X_a}$$

small

$$\left(e^X = \sum_{k=0}^{\infty} \frac{1}{k!} X^k\right) \to e^X \text{ is always invertible}$$
$$\hookrightarrow e^{-X}$$

This is called the exponential parameterization

## Deriving Generators for SO(n)

Elements of Lie groups take the form $e^A$ with $A$ matrix.

For $SO(n)$, $R^T R = 1$ (definition of orthogonal matrices)

What is $R^T$? $R^T e^A e^{-A} = e^{-A} \Rightarrow R^T \left(e^{\cancel{A-A}} + \cancel{\tfrac{1}{2}[A,-A]} + \dots\right) = R^T \cancel{e^0} = \boxed{R^T = e^{-A}}$

Important formula "Baker-Campbell-Hausdorff"

$$e^X e^Y = e^Z \quad \text{for matrices } X, Y, Z$$

$$Z = X + Y + \frac{1}{2}[X,Y] + \frac{1}{12}[X,[X,Y]] - \frac{1}{12}[Y,[X,Y]] + \dots$$

Where $[X,Y] = XY - YX$

→ the "commutator" of matrices $X$ & $Y$
→ how different $XY$ is from $YX$

What constraints on $A$?

$$(e^A)^T = \sum_{n=0}^{\infty} \frac{1}{n!} (A^n)^T = \sum_{n=0}^{\infty} \frac{1}{n!} (A^T)^n = e^{(A^T)}$$

$$\underbrace{(e^A)^T = e^{-A}}_{\text{def of orthogonal}} = e^{(A^T)} \to \boxed{A^T = -A}$$

So for $SO(n)$ skew symmetric matrices

$$A = \begin{bmatrix} 0 & a_{01} & a_{02} & \cdots & a_{0n} \\ -a_{01} & 0 & a_{12} & \cdots & a_{1n} \\ -a_{02} & -a_{12} & 0 & & \vdots \\ \vdots & \vdots & & \ddots & \vdots \\ \vdots & \vdots & & 0 & a_{nn} \\ -a_{0n} & -a_{1n} & \cdots & -a_{nn} & 0 \end{bmatrix}$$

$$\frac{n \times n - n \text{ diagonals}}{2} = \frac{n(n-1)}{2} \text{ d.o.f.}$$

$n=2 = 2(1)/2 = 1$

$n=3, \ \frac{3(2)}{2} = 3$

$n=4, \ \frac{4(3)}{2} = 6$

degrees of freedom → # of generators → basis for all A

SO(2)

$$A = \begin{bmatrix} 0 & a \\ -a & 0 \end{bmatrix}$$

only one generator!

SO(3)

$$A = \begin{bmatrix} 0 & a & b \\ -a & 0 & c \\ -b & -c & 0 \end{bmatrix} = a\begin{bmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + b\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} + c\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{bmatrix}$$

So our group element are the exponentiation of generators $X_\alpha$ multiplied by parameters $\alpha$.

<u>What do they mean?</u>

e.g. When you rotate around $\hat{z}$ your $z$ coordinate stays the same, your $y$ coordinate contributes negatively to your $x$ coordinate and your $x$ coordinate contributes positively to your $y$ coordinate (<u>infinitesimally</u>)

$$e^{\omega \cdot L} = e^{(xL_x + yL_y + zL_z)} = e^{\begin{bmatrix} 0 & -z & y \\ z & 0 & -x \\ -y & x & 0 \end{bmatrix}}$$

Now, we need to check that this representation satisfies all the properties of a group.

# Group Multiplication

for SO(2)

1 generator $\quad e^{\alpha A_1} e^{\alpha' A_1} = e^{\alpha A_1 + \alpha' A_1 + \frac{1}{2}\cancelto{0}{[A_1, A_1]} + \dots} = e^{(\alpha+\alpha')A_1}$

$[A_1, A_1] = 0$

for SO(3)

3 generators $\quad e^{\alpha_i A_i} e^{\alpha_j' A_j} = e^{\alpha_i A_i + \alpha_j' A_j + \frac{1}{2}[\alpha_i A_i, \alpha_j' A_j] + \dots}$

$[A_i, A_i] = 0$, $[A_i, A_j]$ where $i \neq j$

$$[A_1, A_2] = \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}\begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix} - \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix}\begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{pmatrix} - \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -1 & 0 \end{pmatrix}$$

$$= \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix} = -A_3$$

$$[A_2, A_3] = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix}\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{pmatrix} - \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{pmatrix}\begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & -1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} - \begin{pmatrix} 0 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} = -A_1$$

$$[A_3, A_1] = -A_2$$

i.e. remove minus signs.

To simplify commutation relationships, we can choose different basis convention.

One way to write them is

$$L_x = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix} \quad L_y = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix} \quad L_z = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

such that $[L_i, L_j] = \varepsilon_{ijk} L_k$ where $\varepsilon_{ijk} = 1$ for $\varepsilon_{xyz} = \varepsilon_{yzx} = \varepsilon_{zxy}$

$\varepsilon_{ijk} = -1$ for $\varepsilon_{yxz} = \varepsilon_{xzy} = \varepsilon_{zyx}$

$\varepsilon_{ijk} = 0$ for $i=j$, $j=k$, or $i=k$

So $\quad e^{\alpha_i A_i} e^{\alpha_j' A_j} = e^{\underbrace{\alpha_i A_i + \alpha_j' A_j + \frac{1}{2}[\alpha_i A_i, \alpha_j' A_j] + \dots}_{\text{need this to only contain linear combos of generators}}}$

is group multiplication if generators are closed under commutation!

## Aside

What are properties of commutator? $\overset{\#}{}$ $\overset{\text{matrix}}{}$

**Bilinear:** $[aX+bY, cZ] = (aX+bY)cZ - cZ(aX+bY)$
$= acXZ + bcYZ - acZX - bcZY$
$= [aX, cZ] + [bY, cZ]$

**Alternativity:** $[X,X] = XX - XX = 0$ **Anticommutativity:** $[X,Y] = XY - YX = -[Y,X]$

**Jacobi Identity:** $[X,[Y,Z]] + [Y,[Z,X]] + [Z,[X,Y]]$
$= [X, YZ-ZY] + [Y, ZX-XZ] + [Z, XY-YX]$
$= [X, YZ] - [X, ZY] + [Y, ZX] - [Y, XZ] + [Z, XY] - [Z, YX]$
$= XYZ - YZX - XZY + ZYX + YZX - ZXY$
$- YXZ + XZY + ZXY - XYZ - ZYX + YXZ$
$= 0$

## From Algorithms on Reps. to Algorithms on Generators of Lie Groups

We will be able to adapt our algorithms for finite groups to work with generators of infinite Lie groups!

While there are an infinite # of elements, there are finite # of generators!

To do this, we need to understand how common operations on the representation can be rewritten as operations on the generators!

## What about similarity transforms?

$$\mathcal{U} e^{X} \mathcal{U}^{-1} = \mathcal{U}\left(\sum_k \frac{1}{k!} X^k\right)\mathcal{U}^{-1} = \sum_k \frac{1}{k!} \mathcal{U} X^k \mathcal{U}^{-1} = \sum_n \frac{1}{k!} Y = e^{Y} = \boxed{e^{\mathcal{U} X \mathcal{U}^{-1}} = \mathcal{U} e^{X} \mathcal{U}^{-1}}$$

for $Y = \mathcal{U} X \mathcal{U}^{-1}$

Let's define $\mathcal{U} X \mathcal{U}^{-1} = Y$.

$$Y^2 = (\mathcal{U} X \cancel{\mathcal{U}^{-1}})(\mathcal{U} X \mathcal{U}^{-1}) = \mathcal{U} X^2 \mathcal{U}^{-1}$$

$$Y^n = Y\, Y^{n-1} = (\mathcal{U} X \cancel{\mathcal{U}^{-1}})(\mathcal{U} X^{n-1} \mathcal{U}^{-1}) = \mathcal{U} X^n \mathcal{U}^{-1}$$

We can work with similarity tranforms in the exponential!

## What about direct sums?

$$e^{(A \oplus B)} = \sum_k \frac{1}{k!}(A \oplus B)^k = \sum_k \frac{1}{k!}\begin{pmatrix} A & 0 \\ 0 & B \end{pmatrix}^k = \sum_k \frac{1}{k!}\begin{pmatrix} A^k & \\ & B^k \end{pmatrix} = \begin{pmatrix} e^A & 0 \\ 0 & e^B \end{pmatrix} = \boxed{e^A \oplus e^B = e^{A \oplus B}}$$

## What about tensor products?

$e^A \overset{kr}{\otimes} e^B \rightarrow$ Let's use "infinitesimal generators" i.e. generators multiplied by infinitesimal parameters.
Assume

$$\frac{\partial}{\partial \alpha_b} e^{\alpha_a X_a} = e^{\varepsilon_a X_a} = \mathbb{1}_m + \varepsilon_b \frac{\partial}{\partial \varepsilon_b} e^{\varepsilon_a X_a}\Big|_{\vec{\varepsilon}=0} + \varepsilon_b \varepsilon_c \frac{\partial}{\partial \varepsilon_b \partial \varepsilon_c} e^{\varepsilon_a X_a}\Big|_{\vec{\varepsilon}=0}$$

$$= \mathbb{1}_m + \varepsilon_b X_b + \mathcal{O}(|\vec{\varepsilon}|^2)$$

$$\frac{\partial}{\partial \alpha_b} e^{\alpha_a Y_a} = e^{\varepsilon_a Y_a} = \mathbb{1}_n + \varepsilon_b Y_b + \cancel{\mathcal{O}(|\vec{\varepsilon}|)^2}$$

$$e^{\varepsilon_a X_a} \overset{kr}{\otimes} e^{\varepsilon_b Y_b} = (\mathbb{1}_m + \varepsilon_a X_a) \otimes (\mathbb{1}_n + \varepsilon_b Y_b) \qquad \mathcal{O}(|\vec{\varepsilon}|)^2$$

$$= \mathbb{1}_{m \times n} + (\mathbb{1}_m \otimes \varepsilon_b Y_b) + (\varepsilon_a Y_a \otimes \mathbb{1}_n) + \cancel{(\varepsilon_a X_a \otimes \varepsilon_b Y_b)}$$

$$= \mathbb{1}_{m \times n} + \varepsilon_a (X_a \overset{kr}{\oplus} Y_a)$$

$$e^{\alpha_a X_a} \overset{kr}{\otimes} e^{\alpha_b Y_b} = \lim_{k \to \infty} \left(\mathbb{1}_{m \times n} + \frac{\alpha_a}{k}(X_a \overset{kr}{\oplus} Y_a)\right)^k = e^{\alpha_a (X_a \overset{kr}{\oplus} Y_a)}$$