Equivariant Neural Networks

Gijs Bellaard

Presentation date: November 04, 2023
Compiled date: April 10, 2023

1 Convolution Neural Networks

Lets first motivate why we are interested in standard planar convolution/correlation by showing how they naturally arise in the context of machine learning.

Suppose I want to make a feed-forward neural network that takes a photograph of an apple in an environment and returns a segmentation of that apple. It makes sense that if I translate the photograph and give this translated version to the neural network, that I want the network to return a translate version of the original segmentation. In other words we want the network to be *equivariant* with respect to translations of the input image.

Lets put this idea into more mathematical terms. We model the photograph and segmentation as real-valued functions on $\mathbb{R}^2$. Let $F(\mathbb{R}^2, \mathbb{R})$ denote this space of functions. The neural network is then a mapping $\mathcal{N} : F(\mathbb{R}^2, \mathbb{R}) \to F(\mathbb{R}^2, \mathbb{R})$. We consider a translation $\mathbf{v}$ an element of the mathematical group $(\mathbb{R}^2, +)$, and this group acts naturally on points $\mathbf{x} \in \mathbb{R}^2$:

$$\mathbf{v} \triangleright \mathbf{x} := \mathbf{x} + \mathbf{v} \tag{1}$$

and this action extends in the normal way to an action on real-valued functions f on $\mathbb{R}^2$:

$$(\mathbf{v} \triangleright f)(\mathbf{x}) = f(\mathbf{x} - \mathbf{v}) \tag{2}$$

The desired translation equivariance property can now be written as:

$$\mathcal{N}(\mathbf{v} \triangleright f) = \mathbf{v} \triangleright \mathcal{N}(f) \tag{3}$$

for every $f : \mathbb{R}^2 \to \mathbb{R}$ and $\mathbf{v} \in \mathbb{R}^2$, or more abstractly:

$$\mathcal{N} \circ (\mathbf{v}\triangleright) = (\mathbf{v}\triangleright) \circ \mathcal{N} \tag{4}$$

i.e. the network $\mathcal{N}$ and translation $\mathbf{v}\triangleright$ commute, where we now consider $\mathbf{v}\triangleright : F(\mathbb{R}^2, \mathbb{R}) \to F(\mathbb{R}^2, \mathbb{R})$ also as a mapping from functions to functions.

But how does one make a - possibly very complicated - neural network equivariant? Well, it is easier to consider just the constituent parts of the network.

Suppose our feed-forward network can be separated into *layers* $L_i : F(\mathbb{R}^2, \mathbb{R}) \to F(\mathbb{R}^2, \mathbb{R})$, that is $\mathcal{N} = L_n \circ L_{n-1} \circ \cdots \circ L_1$. If we make every layer L_i equivariant then the whole network is also equivariant, this can be quickly understood using (4):

$$\begin{aligned} \mathcal{N} \circ (\mathbf{v}\triangleright) &= L_n \circ L_{n-1} \circ \cdots \circ L_2 \circ L_1 \circ (\mathbf{v}\triangleright) \\ &= L_n \circ L_{n-1} \circ \cdots \circ L_2 \circ (\mathbf{v}\triangleright) \circ L_1 \\ &\vdots \\ &= (\mathbf{v}\triangleright) \circ L_n \circ L_{n-1} \circ \cdots \circ L_2 \circ L_1 \\ &= (\mathbf{v}\triangleright) \circ \mathcal{N} \end{aligned} \tag{5}$$

Linear layers are ubiquitous in machine learning, so let us consider a linear translation equivariant layer. One easy translation equivariant linear operation is *cross-correlation*, or what the machine learning community wrongly calls *convolution*. Cross-correlation $\star$ is defined as:

$$(k \star f)(\mathbf{x}) = \int_{\mathbb{R}^2} k(\mathbf{y} - \mathbf{x})\ f(\mathbf{y})\ d\mathbf{y} \tag{6}$$

notice that this subtly differs from convolution:

$$(k * f)(\mathbf{x}) = \int_{\mathbb{R}^2} k(\mathbf{x} - \mathbf{y})\ f(\mathbf{y})\ d\mathbf{y} \tag{7}$$

The interpretation of cross-correlation is straightforward: we have a filter/kernel k that we move over the image f and we calculate the response. Notice that convolution is different in that the filter/kernel is first inverted and then moved over the image. What most machine learning libraries call convolution is actually implemented as cross-correlation!

Intuitively it makes sense that cross-correlation is translation equivariant: *the same filter is applied everywhere so how could it not be?* Lets check if cross-correlation is really translation equivariant mathematically, i.e if

$$(\mathbf{v}\triangleright) \circ (k\star) = (k\star) \circ (\mathbf{v}\triangleright) \tag{8}$$

Let's apply both sides to some dummy function f and evaluate it at the dummy position $\mathbf{x}$:

$$\begin{aligned} (\mathbf{v} \triangleright (k \star f))(\mathbf{x}) &= (k \star f)(\mathbf{x} - \mathbf{v}) \\ &= \int_{\mathbb{R}^2} k(\mathbf{y} - (\mathbf{x} - \mathbf{v}))\ f(\mathbf{y})\ d\mathbf{y} \\ &= \int_{\mathbb{R}^2} k(\mathbf{y}' - \mathbf{x})\ f(\mathbf{y}' - \mathbf{v})\ d\mathbf{y}' \\ &= \int_{\mathbb{R}^2} k(\mathbf{y}' - \mathbf{x})\ (\mathbf{v} \triangleright f)(\mathbf{y}')\ d\mathbf{y}' \\ &= (k \star (\mathbf{v} \triangleright f))(\mathbf{x}) \end{aligned} \tag{9}$$

where we applied the substitution $\mathbf{y} = \mathbf{y}' - \mathbf{v}$. Indeed, everything works out as expected: cross-correlation is translation equivariant.

So, if we want to make a neural network that is translation equivariant, one easy way to do this is to create it from layers that apply a cross-correlation $\star$ with some kernel k, where the kernel can be chosen freely. The parameters that are learned during training are the parameters that determine the kernel.

2 Roto-translation Equivariance

Let us return to our segmentation of a photograph of an apple. We said that we want the neural network to be equivariant to translations. Makes sense, but there is an additional set of symmetries we might also desire: rotations. So actually we want the network to be equivariant to what we call the *roto-translation group*.

Let $SE(2) = \mathbb{R}^2 \times SO(2) \subset \mathbb{R}^2 \times \mathbb{R}^{2\times 2}$ denote the roto-translation group. An element $g = (\mathbf{v}, \mathbf{R}) \in SE(2)$ of this group consists of a translation $\mathbf{v}$ and a rotation matrix $\mathbf{R}$. It acts on $\mathbb{R}^2$ in the standard way:

$$g \triangleright \mathbf{x} = (\mathbf{v}, \mathbf{R}) \triangleright \mathbf{x} = \mathbf{R}\mathbf{x} + \mathbf{v} \tag{10}$$

The action of this group on $\mathbb{R}^2$ extends straightforwardly to an action on real-valued functions on $\mathbb{R}^2$:

$$(g \triangleright f)(\mathbf{x}) = f(g^{-1} \triangleright \mathbf{x}) = f(\mathbf{R}^{-1}(\mathbf{x} - \mathbf{v})) \tag{11}$$

Now we already saw that the cross-correlation we defined in (6) is equivariant with translations. Maybe it turns out that it is already equivariant to roto-translations? Let us check. So let's first manipulate $g \triangleright (k \star f)$ in a similar way as before and do the substitution $\mathbf{y} = g^{-1} \triangleright \mathbf{y}'$

$$\begin{aligned}
(g \triangleright (k \star f))(\mathbf{x}) &= (k \star f)(g^{-1} \triangleright \mathbf{x}) \\
&= \int_{\mathbb{R}^2} k(\mathbf{y} - g^{-1} \triangleright \mathbf{x})\ f(\mathbf{y})\ d\mathbf{y} \\
&= \int_{\mathbb{R}^2} k(g^{-1} \triangleright \mathbf{y}' - g^{-1} \triangleright \mathbf{x})\ f(g^{-1} \triangleright \mathbf{y}')\ d\mathbf{y}' \\
&= \int_{\mathbb{R}^2} k(\mathbf{R}^{-1}(\mathbf{y}' - \mathbf{x}))\ f(g^{-1} \triangleright \mathbf{y}')\ d\mathbf{y}'
\end{aligned} \tag{12}$$

Well... we are almost there *if* we assume that $k(\mathbf{R}^{-1}\mathbf{x}) = k(\mathbf{x})$ for any $\mathbf{R}^{-1}$, i.e. the kernel is rotationally symmetric:

$$\begin{aligned}
&= \int_{\mathbb{R}^2} k(\mathbf{y}' - \mathbf{x})\ f(g^{-1} \triangleright \mathbf{y}')\ d\mathbf{y}' \\
&= \int_{\mathbb{R}^2} k(\mathbf{y}' - \mathbf{x})\ (g \triangleright f)(\mathbf{y}')\ d\mathbf{y}' \\
&= (k \star (g \triangleright f))(\mathbf{x})
\end{aligned} \tag{13}$$

So we can create an roto-translation equivariant operation by performing a cross-correlation with an rotationally symmetric, i.e. *isotropic*, kernel. But in practice this constraint is way too restrictive!

3 Spherical CNNs

Suppose we now have a real-valued feature map on the sphere $S^2 \subset \mathbb{R}^3$, and we want to generalize cross-correlation to S^2 such that it is equivariant to three dimensional rotations $SO(3) \subset \mathbb{R}^{3\times 3}$.

We can´t just directly re-implement what we wrote in (6): there is no making sense of subtracting two points on S^2. Also, we notice that we made an implicit assumption previously. We assumed/imagined that when we moved the kernel k around that it has some "center": the origin of $\mathbb{R}^2$. We can't do the same on S^2: there is no canonical origin. Now we also need to define integration on S^2, but that is no problem: we use the canonical measure μ on S^2 that we borrow from the ambient space $\mathbb{R}^3$.

Before we continue with S^2 we reconsider what we are *actually* doing when performing a cross-correlation on $\mathbb{R}^2$. So, the kernel k has an implied center, lets call it $\mathbf{x}_0$, and when moving the filter to an position $\mathbf{x}$ we actually mean translating it with some $\mathbf{v}$ such that its center $\mathbf{x}_0$ is at $\mathbf{x}$. In other words, our cross-correlation can be equivalently written as:

$$(k \star f)(\mathbf{x}) = \int_{\mathbb{R}^2} (\mathbf{v}_{\mathbf{x}_0 \to \mathbf{x}} \triangleright k)(\mathbf{y})\ f(\mathbf{y})\ d\mathbf{y} \tag{14}$$

where $\mathbf{v}_{\mathbf{x}_0 \to \mathbf{x}}$ is such that $\mathbf{v}_{\mathbf{x}_0 \to \mathbf{x}} \triangleright \mathbf{x}_0 = \mathbf{x}$.

Now this equivalent way of looking at cross-correlation does generalize: we can *exploit our desired rotational symmetries to transport* our kernel about the feature map. We presuppose that our kernel has some implied center $\mathbf{p}_0 \in S^2$ and we define:

$$(k \star f)(\mathbf{p}) = \int_{S^2} (g_{\mathbf{p}_0 \to \mathbf{p}} \triangleright k)(\mathbf{q})\ f(\mathbf{q})\ d\mu(\mathbf{q}) \tag{15}$$

where $g_{\mathbf{p}_0 \to \mathbf{p}} \triangleright \mathbf{p}_0 = \mathbf{p}$.

However, there is a problem: there are multiple ways to choose $g_{\mathbf{p}_0 \to \mathbf{p}}$. We basically sweep this issue under the rug by only considering kernels that do not depend on this choice. I.e. we should have that $g_{\mathbf{p}_0 \to \mathbf{p}} \triangleright k$ is the same for every possibly $g_{\mathbf{p}_0 \to \mathbf{p}}$ one can choose. This is equivalent to saying that $s_{\mathbf{p}_0} \triangleright k = k$ for any $s_{\mathbf{p}_0} \in SO(3)$ s.t. $s_{\mathbf{p}_0} \triangleright \mathbf{p}_0 = \mathbf{p}_0$, i.e. the kernel must be invariant under the stabilizer subgroup of its center $\mathbf{p}_0$.

Okay, so we have this definition of cross-correlation on S^2, but is it actually

equivariant with respect to rotations? Let us check:

$$\begin{aligned}
(g \triangleright (k \star f))(\mathbf{p}) &= (k \star f)(g^{-1} \triangleright \mathbf{p}) \\
&= \int_{S^2} (g_{\mathbf{p}_0 \to (g^{-1} \triangleright \mathbf{p})} \triangleright k)(\mathbf{q})\ f(\mathbf{q})\ d\mu(\mathbf{q}) \\
&= \int_{S^2} (g^{-1} \triangleright g_{\mathbf{p}_0 \to \mathbf{p}} \triangleright k)(\mathbf{q})\ f(\mathbf{q})\ d\mu(\mathbf{q}) \\
&= \int_{S^2} (g_{\mathbf{p}_0 \to \mathbf{p}} \triangleright k)(g \triangleright \mathbf{q})\ f(\mathbf{q})\ d\mu(\mathbf{q}) \\
&= \int_{S^2} (g_{\mathbf{p}_0 \to \mathbf{p}} \triangleright k)(\mathbf{q}')\ f(g^{-1} \triangleright \mathbf{q}')\ d\mu(\mathbf{q}') \\
&= \int_{S^2} (g_{\mathbf{p}_0 \to \mathbf{p}} \triangleright k)(\mathbf{q}')\ (g \triangleright f)(\mathbf{q}')\ d\mu(\mathbf{q}') \\
&= (k \star (g \triangleright f))(\mathbf{p})
\end{aligned} \tag{16}$$

And we now see another implicit assumption we made earlier: we assume that the measure μ should be invariant under the considered group of symmetries, otherwise the substitution step is not that easily made. Mathematically we mean that:

$$\mu(S) = \mu(g \triangleright S) \tag{17}$$

for every measure set $S \subset S^2$, which has the corollary that:

$$\int_{S^2} f(\mathbf{p})\ d\mu(\mathbf{p}) = \int_{S^2} (g \triangleright f)(\mathbf{p})\ d\mu(\mathbf{p}) \tag{18}$$

Luckily we borrowed the measure on S^2 from $\mathbb{R}^3$, which is indeed invariant under rotations.

All this inspires us to introduce more notation:

$$(f, g) = \int_{S^2} f(\mathbf{q})\ g(\mathbf{q})\ d\mathbf{q} \tag{19}$$

with which we can succinctly write (15) as:

$$(k \star f)(\mathbf{p}) = (g_{\mathbf{p}_0 \to \mathbf{p}} \triangleright k, f) \tag{20}$$

and (16) as:

$$\begin{aligned}
(g \triangleright (k \star f))(\mathbf{p}) &= (k \star f)(g^{-1} \triangleright \mathbf{p}) \\
&= (g_{\mathbf{p}_0 \to (g^{-1} \triangleright \mathbf{p})} \triangleright k, f) \\
&= (g^{-1} \triangleright g_{\mathbf{p}_0 \to \mathbf{p}} \triangleright k, f) \\
&= (g_{\mathbf{p}_0 \to \mathbf{p}} \triangleright k, g \triangleright f) \\
&= (k \star (g \triangleright f))(\mathbf{p})
\end{aligned} \tag{21}$$

4 Group CNNs

Our discussion of Spherical CNNs now easily generalizes to any Lie group G of desired symmetries and homogeneous space M because we already wrote down

everything so abstractly. A homogeneous space is basically a manifold on which the group acts *transitively*, i.e. any point q can be reached from any other point p using a group element g : $g \triangleright p = q$. We need this property to succesfully move the kernel over the space. So, an equivariant cross-correlation on M is literally just what we already wrote down in (15):

$$(k \star f)(\mathbf{p}) = \int_M (g_{\mathbf{p}_0 \to \mathbf{p}} \triangleright k)(\mathbf{q})\ f(\mathbf{q})\ d\mu(\mathbf{q}) \tag{22}$$

which works out so long as the measure μ is invariant w.r.t G and the kernel is symmetric about its center $\mathbf{p}_0$.

5 Riemannian Manifold CNNs

The main motivation of Manifold CNNs is the attempt to generalize cross-correlation even further to any *Riemannian manifold* M. In general a Riemannian manifold does not have global symmetries, so the first obstacle we encounter is the question of how do we move the filter/kernel k over the manifold M?

To keep things tangible I suggest keeping the sphere S^2 in mind as a prototypical example, but you need to forget that the sphere has global rotational symmetries.

The first thing we do on our way to be able to confidently move the kernel over the manifold is to change its domain. In the examples we saw the kernel was always a function on the space itself, i.e. a function on the manifold. We change this to be the tangent space $T_{\mathbf{p}_0}M$ at some implied center $\mathbf{p}_0 \in M$.

To move the kernel around we can now use a standard tool in differential geometry: parallel transport. The parallel transport $\Gamma_{s \to t} : T_{\gamma(s)}M \to T_{\gamma(t)}M$ with respect to a curve γ gives us a way to transport tangent vectors around the manifold. To move the kernel we extend the definition of parallel transport to functions on tangent spaces:

$$(\Gamma_{s \to t} k)(v) = k(\Gamma_{t \to s} v) \tag{23}$$

where $k : T_{\gamma(s)}M \to \mathbb{R}$ and $\Gamma_{s \to t} k : T_{\gamma(t)}M \to \mathbb{R}$.

But even more problems start popping up. Namely, there is no canonical parallel transport on a general manifold. Luckily we have a Riemannian manifold which *does* have a canonical parallel transport: the Levi-Civita connection. But even then the transport is in general not the same for any of the curves γ one can choose!

So again, we sweep this issue under the rug by assuming that for any parallel transport w.r.t a curve γ the transported kernel is the same. Because we are looking at a specific kind of parallel transport, i.e. the Levi-Civita connection, this requirement can usually be satisfied by making the kernel "symmetric" w.r.t the transport about the center $\mathbf{p}_0$:

$$\Gamma_{s \to t} k = k \tag{24}$$

for any curve γ that goes from $\gamma(s) = \mathbf{p}_0$ back to $\gamma(t) = \mathbf{p}_0$.

In the case of $M = \mathbb{R}^2$ this requirement actually boils down to nothing, parallel transport on $\mathbb{R}^2$ behaves really nicely. In the case of $M = S^2$ it boils down to the kernel being rotationally symmetric. In the case of a Mobius band the requirement becomes being invariant under a reflection symmetry.

Now that the kernel has this property we are allowed to write things like: $\Gamma_{\mathbf{p}_0 \to \mathbf{p}} k$ with which we mean to perform any parallel transport with a curve γ that goes from $\mathbf{p}_0$ to $\mathbf{p}$.

But how do we interpret the integral of the cross-correlation now? The feature map is still a real-valued function on the manifold M... This is where the Riemannian exponential map comes into play. The Riemannian exponential map $\exp : TM \to M$ is a mapping from the tangent bundle to the manifold itself. Intuitively, $\mathbf{p} = \exp(v)$ is the place you end up when one starts walking into the direction of v for one unit of time.

We are now ready to create our first definition of the cross-correlation on a Riemannian manifold:

$$(k \star f)(\mathbf{p}) = \int_{T_{\mathbf{p}}M} (\Gamma_{\mathbf{p}_0 \to \mathbf{p}} k)(v)\ f(\exp(v))\ d\mu(v) \tag{25}$$

where μ is the induced measure on $T_{\mathbf{p}}M$ that we get from the Riemannian metric $\mathbf{p}$.