Using and Abusing Equivariance
Supplementary material

Tom Edixhoven Attila Lengyel Jan C. van Gemert
Delft University of Technology

1. Proof Mirroring Equivariance

The proof for the mirroring transformation is practically identical to the proof for rotation, given in Section 3.2 of the paper.

We use the function *index*, introduced in the paper. The index function returns the indices of the input values used by a convolutional or pooling layer to calculate the value located at index (x, y) in the output:

$$\text{index}\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \left[\begin{bmatrix} sx \\ sy \end{bmatrix}, \begin{bmatrix} sx+k-1 \\ sy+k-1 \end{bmatrix}\right]. \tag{1}$$

Here s is the stride used for subsampling and k represents the kernel size. The output of the function is a square patch, denoted as $[\vec{u}, \vec{v}]$, where $\vec{u}$ and $\vec{v}$ represent the indices of the top left and bottom right corner, respectively. The sampled indices include all integer tuples within this patch.

Similarly to the R function in the paper, we now introduce a function M, which takes an index (x, y) as input and returns the indices mirrored horizontally:

$$M_n\left(\begin{bmatrix} x \\ y \end{bmatrix}\right) = \begin{bmatrix} n-1-x \\ y \end{bmatrix}, \tag{2}$$

where n indicates the width and height of the feature map in which the index (x, y) is located. We further generalise Equation 2 to an input patch $[\vec{u}, \vec{v}]$ rather than a single coordinate, resulting in Equation 3:

$$\begin{aligned} M_n([\vec{u}, \vec{v}]) &= M_n\left(\left[\begin{bmatrix} x_1 \\ y_1 \end{bmatrix}, \begin{bmatrix} x_2 \\ y_2 \end{bmatrix}\right]\right) \\ &= \left[\begin{bmatrix} n-1-x_2 \\ y_1 \end{bmatrix}, \begin{bmatrix} n-1-x_1 \\ y_2 \end{bmatrix}\right] \end{aligned} \tag{3}$$

In the resulting output coordinates x_1 and x_2 get interchanged due to the mirroring of the patch: the top left corner becomes the top right corner, while the bottom right corner becomes the bottom left corner.

Given that our layer takes a feature map with a width and height of i as input, we can write the width and height of the output feature map as

$$o = \lfloor \frac{i-k}{s} \rfloor + 1. \tag{4}$$

For a layer to be exactly equivariant, determining the sampled indices and then mirroring should return the same result as mirroring first and then determining the sampled indices, which we can formally denote as

$$\text{index}\left(M_o\left(\begin{bmatrix} x \\ y \end{bmatrix}\right)\right) = M_i\left(\text{index}\left(\begin{bmatrix} x \\ y \end{bmatrix}\right)\right). \tag{5}$$

To solve the left-hand side, we substitute Equation 2 into Equation 1, yielding

$$\begin{aligned} &\text{index}\left(M_o\left(\begin{bmatrix} x \\ y \end{bmatrix}\right)\right) = \\ &\text{index}\left(\begin{bmatrix} \lfloor \frac{i-k}{s} \rfloor - x \\ y \end{bmatrix}\right) = \\ &\left[\begin{bmatrix} s\lfloor \frac{i-k}{s} \rfloor - sx \\ sy \end{bmatrix}, \begin{bmatrix} s\lfloor \frac{i-k}{s} \rfloor - sx + k - 1 \\ sy+k-1 \end{bmatrix}\right]. \end{aligned} \tag{6}$$

The same can be done for the right-hand side, by substituting Equation 1 into Equation 3, resulting in

$$\begin{aligned} &M_i\left(\text{index}\left(\begin{bmatrix} x \\ y \end{bmatrix}\right)\right) = \\ &R_i\left(\left[\begin{bmatrix} sx \\ sy \end{bmatrix}, \begin{bmatrix} sx+k-1 \\ sy+k-1 \end{bmatrix}\right]\right) = \\ &\left[\begin{bmatrix} i-k-sx \\ sy \end{bmatrix}, \begin{bmatrix} i-1-sx \\ sy+k-1 \end{bmatrix}\right]. \end{aligned} \tag{7}$$

Substituting Equations 6 and 7 into Equation 5, we find two equations

$$s\lfloor \frac{i-k}{s} \rfloor - sx = i-k-sx, \tag{8}$$

$$s\lfloor \frac{i-k}{s} \rfloor - sx + k - 1 = i-1-sx. \tag{9}$$

Removing duplicate terms yields a single equation

$$s\lfloor \frac{i-k}{s} \rfloor = i-k. \tag{10}$$

Equation 10 is actually identical to the one found for rotation equivariance. Therefore, when making a network that is exactly equivariant to mirroring, the same restriction on stride, kernel size and input size should hold as for rotation equivariance.

2. Statistical Analysis of Significance

To determine the statistical significance of our results, we compare each pair of models using an independent t-Test testing the null hypothesis $H_0 : \mu_a = \mu_b$. We use a significance level $\alpha = 1.0 \times 10^{-2}$. However, since we perform 12 comparisons in total, we use Bonferroni correction and find a new significance level $\alpha = 8.33 \times 10^{-4}$. The p-values resulting the the t-Tests for MNIST can be found in Table 1 and in Table 2 for RotMNIST. The values were calculated using a 100 repeats for each condition to ensure we had a representative normal distribution for the performance. We then visually confirmed the performance distribution to be a normal distribution. Due to unequal variances between the performance of P4 and Z2 networks on RotMNIST, a Welch's t-Test was used to calculate the p-value for comparisons including the Z2 network.

	P4 (27)	P4 (28)	P4 (29)
Z2 (28)	1.43×10^{-1}	7.92×10^{-82}	1.09×10^{-1}
P4 (27)	-	1.19×10^{-62}	9.97×10^{-2}
P4 (28)	-	-	3.03×10^{-57}

Table 1: p-values for two sided t-Test for different networks trained on the MNIST dataset. The input dimension of the network is indicated using parentheses.

	P4 (27)	P4 (28)	P4 (29)
Z2 (28)	2.95×10^{-99}	1.53×10^{-99}	2.95×10^{-99}
P4 (27)	-	4.54×10^{-1}	1.70×10^{-1}
P4 (28)	-	-	4.44×10^{-1}

Table 2: p-values for two sided t-Test for different networks trained on the RotMNIST dataset. For p-values of comparisons containing the Z2 network, a Welch's t-Test is used due to unequal variances. The input dimension of the network is indicated using parentheses.

For MNIST, we find a significant difference between our exactly equivariant network and the other networks. For RotMNIST we find no significant differences between the P4 equivariant networks, but we do find that the Z2 equivariant network performs significantly worse than the others.

To assert the effect size, we look at the *95%-confidence intervals*, given in Table 3. We find that on MNIST, the exactly equivariant network has a performance drop between 0.65% and 0.91% compared to the other networks. On RotMNIST, P4 equivariant networks offer a performance increase between 4.97% and 5.62% compared to a standard CNN.

Model	Equivariance	MNIST	RotMNIST
Z2CNN	- (28)	[98.44; 98.51]	[91.35; 91.85]
P4CNN	Approx (27)	[98.47; 98.57]	[96.86; 96.97]
P4CNN	Exact (28)	[97.66; 97.72]	[96.85; 96.93]
P4CNN	Approx (29)	[98.37; 98.47]	[96.82; 96.92]

Table 3: Network accuracy confidence interval on MNIST and RotMNIST test sets. The standard deviation is calculated using a 100 runs with different seeds. The equivariance column contains whether the network is exactly or approximately equivariant and the input dimensions of the network in parentheses.

To make the analysis more robust, one could also choose to model the performance according to two independent variables: (1) type of equivariance and (2) input size. This would require the additional training of two Z2CNN networks, one with an input size of 27 and another with an input size of 29. Due to the scope of our work, we deemed our current statistical analysis to be sufficient.