Supplementary Material of PDO-eS^{2}CNNs: Partial Differential Operator Based Equivariant Spherical CNNs

1 Proof

Lemma 1 *If $s \in C^\infty(S^2)$, $\forall R, \widetilde{R} \in SO(3)$ and $i = 1, 2$, we have*

$$\frac{\partial}{\partial x_i^{(A_R)}} \left[\pi^S_{\widetilde{R}}[s]\right](P_R) = \frac{\partial}{\partial x_i^{(A_{\widetilde{R}^{-1}R})}}[s](P_{\widetilde{R}^{-1}R}), \qquad (1)$$

where (P_R, A_R) is the representation of R.

Proof 1 *Firstly, we show that $\forall R \in SO(3)$ and $i = 1, 2$,*

$$\begin{aligned}
&\frac{\partial}{\partial x_i^{(A_R)}}[s](P_R) \qquad (2)\\
=&e_i^T \nabla_x^{(A_R)}[s](P_R)\\
=&e_i^T A_R^{-1} \nabla_x \left[\bar{s} \cdot \varphi^{-1}_{P_R}\right](0)\\
=&e_i^T A_R^{-1} \nabla_x \left[\bar{s}\left(\bar{P}_R \left(\begin{array}{c} x_1 \\ x_2 \\ \sqrt{1-|x|^2}\end{array}\right)\right)\right]\Bigg|_{x_1=x_2=0}, \qquad (3)
\end{aligned}$$

where $e_1 = (1, 0)^T$ and $e_2 = (0, 1)^T$. We denote $y = F(x_1, x_2) = (x_1, x_2, \sqrt{1-|x|^2})^T$, then

$$\begin{aligned}
&\frac{\partial}{\partial x_i^{(A_R)}}[s](P_R)\\
=&e_i^T A_R^{-1} J_F(x_1, x_2)^T \bar{P}_R^T \nabla[\bar{s}]\left(\bar{P}_R y\right)\Big|_{x_1=x_2=0},
\end{aligned}$$

where the Jacobian

$$\begin{aligned}
J_F(x_1, x_2)^T &= \left(\begin{array}{ccc} \partial y_1/\partial x_1 & \partial y_2/\partial x_1 & \partial y_3/\partial x_1 \\ \partial y_1/\partial x_2 & \partial y_2/\partial x_2 & \partial y_3/\partial x_2 \end{array}\right)\\
&= \left(\begin{array}{ccc} 1 & 0 & \frac{-x_1}{\sqrt{1-|x|^2}} \\ 0 & 1 & \frac{-x_2}{\sqrt{1-|x|^2}} \end{array}\right).
\end{aligned}$$

So

$$\begin{aligned}
&\frac{\partial}{\partial x_i^{(A_R)}}[s](P_R)\\
=&e_i^T A_R^{-1} \left(I, \frac{-x}{\sqrt{1-|x|^2}}\right) \bar{P}_R^{-1} \nabla[\bar{s}]\left(\bar{P}_R y\right)\Big|_{x_1=x_2=0}\\
=&e_i^T \left(I, \frac{-A_R^{-1}x}{\sqrt{1-|x|^2}}\right) Z(\gamma_R)^{-1} \bar{P}_R^{-1} \nabla[\bar{s}]\left(\bar{P}_R y\right)\Big|_{x_1=x_2=0}\\
=&e_i^T \left(I, \frac{-A_R^{-1}x}{\sqrt{1-|x|^2}}\right) R^{-1} \nabla[\bar{s}]\left(\bar{P}_R y\right)\Big|_{x_1=x_2=0}\\
=&\left(e_i^T, 0\right) R^{-1} \nabla[\bar{s}](P_R). \qquad (4)
\end{aligned}$$

Thus for the right hand side of (1),

$$\begin{aligned}
\frac{\partial}{\partial x_i^{(A_{\widetilde{R}^{-1}R})}}[s](P_{\widetilde{R}^{-1}R}) &= \left(e_i^T, 0\right) R^{-1} \widetilde{R} \nabla[\bar{s}]\left(P_{\widetilde{R}^{-1}R}\right)\\
&= \left(e_i^T, 0\right) R^{-1} \widetilde{R} \nabla[\bar{s}]\left(\widetilde{R}^{-1} P_R\right).
\end{aligned}$$

For the left hand side of (1), we denote a spherical function $t(P) = \pi^S_{\widetilde{R}}[s](P) = s(\widetilde{R}^{-1}P)$, then we have

$$\begin{aligned}
\frac{\partial}{\partial x_i^{(A_R)}} \left[\pi^S_{\widetilde{R}}[s]\right](P_R) =&\frac{\partial}{\partial x_i^{(A_R)}}[t](P_R)\\
=&\left(e_i^T, 0\right) R^{-1} \nabla[\bar{t}](P_R).
\end{aligned}$$

Obviously, we can take the extended function on the Euclidean space $\bar{t}(x) = \bar{s}(\widetilde{R}^{-1}x), \forall x \in \mathbb{R}^3$, then

$$\nabla[\bar{t}](P_R) = \nabla\left[\bar{s}\left(\widetilde{R}^{-1}x\right)\right]\Big|_{x=P_R} = \widetilde{R}\nabla[\bar{s}]\left(\widetilde{R}^{-1}P_R\right).$$

As a result, we have

$$\begin{aligned}
\frac{\partial}{\partial x_i^{(A_R)}} \left[\pi^S_{\widetilde{R}}[s]\right](P_R) &= \left(e_i^T, 0\right) R^{-1} \widetilde{R} \nabla[\bar{s}]\left(\widetilde{R}^{-1}P_R\right)\\
&=\frac{\partial}{\partial x_i^{(A_{\widetilde{R}^{-1}R})}}[s](P_{\widetilde{R}^{-1}R}).
\end{aligned}$$

$\blacksquare$

Lemma 2 *If $s \in C^\infty(S^2)$, $\forall R, \widetilde{R} \in SO(3), i,j=1,2$, we have*

$$\begin{aligned}&\frac{\partial}{\partial x_i^{(A_R)}}\frac{\partial}{\partial x_j^{(A_R)}}\left[\pi_{\widetilde{R}}^S[s]\right](P_R)\\=&\frac{\partial}{\partial x_i^{(A_{\widetilde{R}^{-1}R})}}\frac{\partial}{\partial x_i^{(A_{\widetilde{R}^{-1}R})}}[s](P_{\widetilde{R}^{-1}R}),\end{aligned}\tag{5}$$

where (P_R, A_R) is the representation of R.

Proof 2 *Firstly, by definition, $\forall R, \widetilde{R} \in SO(3), i,j=1,2$,*

$$\begin{aligned}&\frac{\partial}{\partial x_i^{(A_R)}}\frac{\partial}{\partial x_j^{(A_R)}}[s](P_R)\\=&\frac{\partial}{\partial x_i^{(A_R)}}\frac{\partial}{\partial x_j^{(A_R)}}[\bar{s}\cdot\varphi_{P_R}^{-1}](0)\\=&\frac{\partial}{\partial x_i^{(A_R)}}\left[e_j^T A_R^{-1}\nabla_x[\bar{s}\cdot\varphi_{P_R}^{-1}]\right](0)\\=&e_i^T A_R^{-1}\nabla_x\left[e_j^T\left(I,\frac{-A_R^{-1}x}{\sqrt{1-|x|^2}}\right)R^{-1}\nabla[\bar{s}]\left(\bar{P}_R y\right)\right]\Bigg|_{x_1=x_2=0},\end{aligned}$$

where $e_1=(1,0)^T, e_2=(0,1)^T$ and $y=F(x_1,x_2)=(x_1,x_2,\sqrt{1-|x|^2})^T$. The derivation from the second line to the third line is due to (3) and (4). For ease of presentation, we denote that

$$h(x)=e_j^T\left(I,\frac{-A_R^{-1}x}{\sqrt{1-|x|^2}}\right)R^{-1}\nabla[\bar{s}]\left(\bar{P}_R y\right),$$

and $h(x)=f(x)^T g(x)$, where

$$f(x)=\left(I,\frac{-A_R^{-1}x}{\sqrt{1-|x|^2}}\right)^T e_j \tag{6}$$

and

$$g(x)=R^{-1}\nabla[\bar{s}]\left(\bar{P}_R y\right). \tag{7}$$

As a result,

$$\begin{aligned}&\frac{\partial}{\partial x_i^{(A_R)}}\frac{\partial}{\partial x_j^{(A_R)}}[s](P_R)\\=&e_i^T A_R^{-1}\nabla_x\left[h(x)\right]\Big|_{x_1=x_2=0}\\=&e_i^T A_R^{-1}\nabla_x\left[f(x)^T g(x)\right]\Big|_{x_1=x_2=0}\\=&e_i^T A_R^{-1}\left(Df(x)^T g(x)\Big|_{x_1=x_2=0}+Dg(x)^T f(x)\Big|_{x_1=x_2=0}\right)\\=&e_i^T A_R^{-1}Df(x)^T g(x)\Big|_{x_1=x_2=0}+e_i^T A_R^{-1}Dg(x)^T f(x)\Big|_{x_1=x_2=0}.\end{aligned}\tag{8}$$

Firstly, we calculate the first term of the right hand side of (8). When $e_j=e_1$ in (6) and (7), we have

$$f(x)^T=\left(1,0,-\frac{\cos\gamma_R x_1+\sin\gamma_R x_2}{\sqrt{1-|x|^2}}\right),$$

then

$$\begin{aligned}Df(x)^T=&\left(0,0,-\frac{1}{\sqrt{1-|x|^2}}\begin{pmatrix}\cos\gamma_R\\ \sin\gamma_R\end{pmatrix}\right.\\&\left.-(\cos\gamma_R x_1+\sin\gamma_R x_2)\nabla_x\left[\frac{1}{\sqrt{1-|x|^2}}\right]\right).\end{aligned}$$

So

$$\begin{aligned}&e_i^T A_R^{-1}Df(x)^T g(x)\Big|_{x_1=x_2=0}\\=&e_i^T\begin{pmatrix}0&0&-1\\0&0&0\end{pmatrix}R^{-1}\nabla[\bar{s}]\left(P_R\right).\end{aligned}$$

Similarly, we can get that when $e_j=e_2$,

$$\begin{aligned}&e_i^T A_R^{-1}Df(x)^T g(x)\Big|_{x_1=x_2=0}\\=&e_i^T\begin{pmatrix}0&0&0\\0&0&-1\end{pmatrix}R^{-1}\nabla[\bar{s}]\left(P_R\right).\end{aligned}$$

In all,

$$\begin{aligned}&e_i^T A_R^{-1}Df(x)^T g(x)\Big|_{x_1=x_2=0}\\=&\left(0,0,-e_i^T e_j\right)R^{-1}\nabla[\bar{s}]\left(P_R\right).\end{aligned}$$

Now we calculate the second term of the right hand side of (8), we have

$$\begin{aligned}g(x)^T=&\left(\nabla[\bar{s}]\left(\bar{P}_R y\right)\right)^T R\\=&\left(\partial_1[\bar{s}]\left(\bar{P}_R y\right),\partial_2[\bar{s}]\left(\bar{P}_R y\right),\partial_3[\bar{s}]\left(\bar{P}_R y\right)\right)R,\end{aligned}$$

where ∂_k deontes the first-order PDO w.r.t. the k-th coordinate, so

$$\begin{aligned}e_i^T A_R^{-1}Dg(x)^T=&e_i^T A_R^{-1}\left(\nabla_x\left[\partial_1[\bar{s}]\left(\bar{P}_R y\right)\right],\right.\\&\left.\nabla_x\left[\partial_2[\bar{s}]\left(\bar{P}_R y\right)\right],\nabla_x\left[\partial_3[\bar{s}]\left(\bar{P}_R y\right)\right]\right)R.\end{aligned}$$

According to (3) and (4), we can get that

$$\begin{aligned}&e_i^T A_R^{-1}\nabla_x\left[\partial_k[\bar{s}]\left(\bar{P}_R y\right)\right]\\=&e_i^T\left(I,\frac{-A_R^{-1}x}{\sqrt{1-|x|^2}}\right)R^{-1}\nabla\left[\partial_k[\bar{s}]\right]\left(\bar{P}_R y\right),\end{aligned}$$

i.e.,

$$e_i^T A_R^{-1}Dg(x)^T=e_i^T\left(I,\frac{-A_R^{-1}x}{\sqrt{1-|x|^2}}\right)R^{-1}\nabla^2[\bar{s}]\left(\bar{P}_R y\right)R.$$

So

$$\begin{aligned}&e_i^T A_R^{-1}Dg(x)^T f(x)\Big|_{x_1=x_2=0}\\=&\left(e_i^T,0\right)R^{-1}\nabla^2[\bar{s}]\left(P_R\right)R\left(e_j^T,0\right)^T.\end{aligned}$$

As a result, we have

$$\begin{aligned}&\frac{\partial}{\partial x_i^{(A_R)}}\frac{\partial}{\partial x_j^{(A_R)}}[s](P_R)\\=&e_i^T A_R^{-1}Df(x)^T g(x)\Big|_{x_1=x_2=0}\\&+e_i^T A_R^{-1}Dg(x)^T f(x)\Big|_{x_1=x_2=0}\\=&\left(0,0,-e_i^T e_j\right)R^{-1}\nabla[\bar{s}]\left(P_R\right)\\&+\left(e_i^T,0\right)R^{-1}\nabla^2[\bar{s}]\left(P_R\right)R\left(e_j^T,0\right)^T.\end{aligned}$$

Thus for the right hand of (5),

$$\begin{aligned}&\frac{\partial}{\partial x_i^{(A_{\widetilde{R}^{-1}R})}}\frac{\partial}{\partial x_j^{(A_{\widetilde{R}^{-1}R})}}[s](P_{\widetilde{R}^{-1}R})\\&=\left(0,0,-e_i^T e_j\right)R^{-1}\widetilde{R}\nabla[\bar{s}]\left(\widetilde{R}^{-1}P_R\right)\\&\quad+\left(e_i^T,0\right)R^{-1}\widetilde{R}\nabla^2[\bar{s}]\left(\widetilde{R}^{-1}P_R\right)\widetilde{R}^{-1}R\left(e_j^T,0\right)^T.\end{aligned}$$

As for the left hand of (5), similar to Proof 1, we denote that the spherical function $t(P)=\pi_{\widetilde{R}}^{S}[s](P)=s(\widetilde{R}^{-1}P)$ and the extended 3D function $\bar{t}(x)=\bar{s}(\widetilde{R}^{-1}x),\forall x\in\mathbb{R}^3$, then

$$\begin{aligned}&\nabla[\bar{t}](P_R)=\nabla\left[\bar{s}\left(\widetilde{R}^{-1}x\right)\right]\Big|_{x=P_R}=\widetilde{R}\nabla\left[\bar{s}\right]\left(\widetilde{R}^{-1}P_R\right),\\&\nabla^2[\bar{t}](P_R)=\nabla^2\left[\bar{s}\left(\widetilde{R}^{-1}x\right)\right]\Big|_{x=P_R}=\widetilde{R}\nabla^2\left[\bar{s}\right]\left(\widetilde{R}^{-1}P_R\right)\widetilde{R}^{-1}.\end{aligned}$$

So

$$\begin{aligned}&\frac{\partial}{\partial x_i^{(A_R)}}\frac{\partial}{\partial x_j^{(A_R)}}\left[\pi_{\widetilde{R}}^{S}[s]\right](P_R)\\=&\frac{\partial}{\partial x_i^{(A_R)}}\frac{\partial}{\partial x_j^{(A_R)}}[t](P_R)\\=&\left(0,0,-e_i^T e_j\right)R^{-1}\nabla[\bar{t}](P_R)\\&+\left(e_i^T,0\right)R^{-1}\nabla^2[\bar{t}](P_R)R\left(e_j^T,0\right)^T\\=&\left(0,0,-e_i^T e_j\right)R^{-1}\widetilde{R}\nabla[\bar{s}]\left(\widetilde{R}^{-1}P_R\right)\\&+\left(e_i^T,0\right)R^{-1}\widetilde{R}\nabla^2[\bar{s}]\left(\widetilde{R}^{-1}P_R\right)\widetilde{R}^{-1}R\left(e_j^T,0\right)^T\\=&\frac{\partial}{\partial x_i^{(A_{\widetilde{R}^{-1}R})}}\frac{\partial}{\partial x_j^{(A_{\widetilde{R}^{-1}R})}}[s](P_{\widetilde{R}^{-1}R}).\end{aligned}$$

■

Theorem 1 *If $s\in C^\infty(S^2)$ and $so\in C^\infty(SO(3))$, $\forall\widetilde{R}\in SO(3)$, we have*

$$\Psi\left[\pi_{\widetilde{R}}^{S}[s]\right]=\pi_{\widetilde{R}}^{SO}\left[\Psi[s]\right],\tag{9}$$

$$\Phi\left[\pi_{\widetilde{R}}^{SO}[so]\right]=\pi_{\widetilde{R}}^{SO}\left[\Phi[so]\right].\tag{10}$$

Proof 3 *According to Lemmas 1 and 2, $\forall R,\widetilde{R}\in SO(3)$,*

$$\begin{aligned}&\Psi\left[\pi_{\widetilde{R}}^{S}[s]\right](R)\\=&\chi^{(A_R)}\left[\pi_{\widetilde{R}}^{S}[s]\right](P_R)\\=&\Bigg(w_1+w_2\frac{\partial}{\partial x_1^{(A_R)}}+w_3\frac{\partial}{\partial x_2^{(A_R)}}+w_4\frac{\partial}{\partial x_1^{(A_R)}}\frac{\partial}{\partial x_1^{(A_R)}}\\&+w_5\frac{\partial}{\partial x_1^{(A_R)}}\frac{\partial}{\partial x_2^{(A_R)}}+w_6\frac{\partial}{\partial x_2^{(A_R)}}\frac{\partial}{\partial x_2^{(A_R)}}\Bigg)\left[\pi_{\widetilde{R}}^{S}[s]\right](P_R)\\=&\Bigg(w_1+w_2\frac{\partial}{\partial x_1^{(A_{\widetilde{R}^{-1}R})}}+w_3\frac{\partial}{\partial x_2^{(A_{\widetilde{R}^{-1}R})}}\\&+w_4\frac{\partial}{\partial x_1^{(A_{\widetilde{R}^{-1}R})}}\frac{\partial}{\partial x_1^{(A_{\widetilde{R}^{-1}R})}}+w_5\frac{\partial}{\partial x_1^{(A_{\widetilde{R}^{-1}R})}}\frac{\partial}{\partial x_2^{(A_{\widetilde{R}^{-1}R})}}\\&+w_6\frac{\partial}{\partial x_2^{(A_{\widetilde{R}^{-1}R})}}\frac{\partial}{\partial x_2^{(A_{\widetilde{R}^{-1}R})}}\Bigg)[s]\left(P_{\widetilde{R}^{-1}R}\right)\\=&\chi^{(A_{\widetilde{R}^{-1}R})}[s](P_{\widetilde{R}^{-1}R})\\=&\pi_{\widetilde{R}}^{SO}\left[\Psi[s]\right](R).\end{aligned}\tag{11}$$

So (9) is satisfied. As for (10),

$$\begin{aligned}&\Phi\left[\pi_{\widetilde{R}}^{SO}[so]\right](P_R,A_R)\\=&\int_{SO(2)}\chi_A^{(A_R)}\left[so\left(\widetilde{R}^{-1}P,A_{\widetilde{R}^{-1}R}A\right)\right]\Big|_{P=P_R}d\nu(A)\\=&\int_{SO(2)}\chi_A^{(A_R)}\left[\pi_{\widetilde{R}}^{S}\left[so\left(P,A_{\widetilde{R}^{-1}R}A\right)\right]\right]\Big|_{P=P_R}d\nu(A)\\=&\int_{SO(2)}\chi_A^{(A_{\widetilde{R}^{-1}R})}[so]\left(P_{\widetilde{R}^{-1}R},A_{\widetilde{R}^{-1}R}A\right)d\nu(A)\\=&\pi_{\widetilde{R}}^{SO}\left[\int_{SO(2)}\chi_A^{(A_R)}[so](P_R,A_RA)\,d\nu(A)\right]\\=&\pi_{\widetilde{R}}^{SO}\left[\Phi[so]\right](P_R,A_R).\end{aligned}$$

The derivation from the third line to the fourth line is due to (11). So (10) is satisfied. ■

Theorem 2 *If $s\in C^\infty(S^2)$, $\forall\widetilde{R}\in SO(3)$, we have*

$$T\left[\pi_{\widetilde{R}}^{S}[s]\right]=\pi_{\widetilde{R}}^{SO}\left[T[s]\right].$$

Proof 4 *According to Theorems 1, we have*

$$\begin{aligned}T\left[\pi_{\widetilde{R}}^{S}[s]\right]=&\Phi^{(L)}\left[\cdots\sigma\left(\Phi^{(1)}\left[\sigma\left(\Psi\left[\pi_{\widetilde{R}}^{S}[s]\right]\right)\right]\right)\right]\\=&\Phi^{(L)}\left[\cdots\sigma\left(\Phi^{(1)}\left[\sigma\left(\pi_{\widetilde{R}}^{SO}\left[\Psi[s]\right]\right)\right]\right)\right]\\=&\Phi^{(L)}\left[\cdots\sigma\left(\Phi^{(1)}\left[\pi_{\widetilde{R}}^{SO}\left[\sigma\left(\Psi[s]\right)\right]\right]\right)\right]\\=&\Phi^{(L)}\left[\cdots\sigma\left(\pi_{\widetilde{R}}^{SO}\left[\Phi^{(1)}\left[\sigma\left(\Psi[s]\right)\right]\right]\right)\right]\\=&\pi_{\widetilde{R}}^{SO}\left[\Phi^{(L)}\left[\cdots\sigma\left(\Phi^{(1)}\left[\sigma(\Psi[s])\right]\right)\right]\right]\\=&\pi_{\widetilde{R}}^{SO}\left[T[s]\right].\end{aligned}$$

■

Lemma 3 *$\forall P \in \Omega$ and $\boldsymbol{w} \in \mathbb{R}^5$,*

$$\boldsymbol{w}^T D_P = \boldsymbol{w}^T \hat{D}_P + O(\rho).$$

Proof 5 *According to (13) in the main body, we have*

$$F_P = V_P D_P + O(\rho^3),$$

and then

$$\begin{aligned} D_P =&(V_P^T V_P)^{-1} V_P D_P + (V_P^T V_P)^{-1} V_P O(\rho) \\ =&\hat{D}_P + (V_P^T V_P)^{-1} V_P O(\rho^3). \end{aligned}$$

Actually,

$$V_P = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{i1} & x_{i2} & \frac{1}{2}x_{i1}^2 & x_{i1}x_{i2} & \frac{1}{2}x_{i2}^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} = XC,$$

where

$$X = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ \frac{x_{i1}}{\rho} & \frac{x_{i2}}{\rho} & \frac{x_{i1}^2}{2\rho^2} & \frac{x_{i1}x_{i2}}{\rho^2} & \frac{x_{i2}^2}{2\rho^2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

and

$$C = \begin{bmatrix} \rho I_2 & 0 \\ 0 & \rho^2 I_3 \end{bmatrix}.$$

Obviously $X = O(1)$, so we have

$$\begin{aligned} (V_P^T V_P)^{-1} V_P O(\rho^3) =& C^{-1}(X^T X)^{-1} X^T O(\rho^3) \\ =& \begin{bmatrix} \frac{I_2}{\rho} & 0 \\ 0 & \frac{I_3}{\rho^2} \end{bmatrix} O(\rho^3) \\ =& \begin{bmatrix} O(\rho^2)\mathbf{1}_2 \\ O(\rho)\mathbf{1}_3 \end{bmatrix}, \end{aligned}$$

i.e., $\forall \boldsymbol{w} \in \mathbb{R}^5$,

$$\begin{aligned} \boldsymbol{w}^T D_P =& \boldsymbol{w}^T \hat{D}_P + \boldsymbol{w}^T (V_P^T V_P)^{-1} V_P O(\rho^3) \\ =& \boldsymbol{w}^T \hat{D}_P + O(\rho). \end{aligned}$$

■

Theorem 3 *$\forall \widetilde{R} \in SO(3)$,*

$$\widetilde{\Psi}\left[\pi_{\widetilde{R}}^S[\boldsymbol{I}]\right] = \pi_{\widetilde{R}}^{SO}\left[\widetilde{\Psi}[\boldsymbol{I}]\right] + O(\rho), \tag{12}$$

$$\widetilde{\Phi}\left[\pi_{\widetilde{R}}^{SO}[\boldsymbol{F}]\right] = \pi_{\widetilde{R}}^{SO}\left[\widetilde{\Phi}[\boldsymbol{F}]\right] + O(\rho) + O\left(\frac{1}{N^2}\right), \tag{13}$$

where transformations acting on discrete inputs and feature maps are defined as $\pi_{\widetilde{R}}^S[\boldsymbol{I}](P) = \pi_{\widetilde{R}}^S[s](P)$ and $\pi_{\widetilde{R}}^{SO}[\boldsymbol{F}](P,i) = \pi_{\widetilde{R}}^{SO}[so](P, A_i)$, respectively.

Proof 6 *$\forall i = 0, 1, \cdots, N-1$, the operator $\chi^{(Z_i)}$ is a linear combination of differential operators and $\widetilde{\chi}^{(Z_i)}$ is a linear combination of corresponding numerical estimations, except a trivial scalar. According to Lemma 3, we have that $\forall P \in \Omega$,*

$$\chi^{(Z_i)}[s](P) = \widetilde{\chi}^{(Z_i)}[\boldsymbol{I}](P) + O(\rho),$$
$$\chi^{(Z_i)}\left[\pi_{\widetilde{R}}^S[s]\right](P) = \widetilde{\chi}^{(Z_i)}\left[\pi_{\widetilde{R}}^S[\boldsymbol{I}]\right](P) + O(\rho),$$

i.e.,

$$\Psi[s](P, Z_i) = \widetilde{\Psi}[\boldsymbol{I}](P, i) + O(\rho),$$
$$\Psi\left[\pi_{\widetilde{R}}^S[s]\right](P, Z_i) = \widetilde{\Psi}\left[\pi_{\widetilde{R}}^S[\boldsymbol{I}]\right](P, i) + O(\rho). \tag{14}$$

Easily, we have

$$\pi_{\widetilde{R}}^{SO}\left[\Psi[s]\right](P, Z_i) = \pi_{\widetilde{R}}^{SO}\left[\widetilde{\Psi}[\boldsymbol{I}]\right](P, i) + O(\rho). \tag{15}$$

From (9) we know that the left hand sides of (14) and (15) equal, hence the right hand sides of the two equations are the same, which results in (12).

As for (13),

$$\begin{aligned} &\Phi[so](P, Z_i) \\ =&\int_{SO(2)} \chi_Z^{(Z_i)}\ [so](P, Z_i Z) d\nu(Z) \\ =&\frac{\nu(SO(2))}{N} \sum_{j=0}^{N-1} \chi_{Z_j}^{(Z_i)}[so](P, Z_i Z_j) + O\left(\frac{1}{N^2}\right) \\ =&\frac{\nu(SO(2))}{N} \sum_{j=0}^{N-1} \left(\widetilde{\chi}_{Z_j}^{(Z_i)}[\boldsymbol{F}](P, i \oplus j) + O(\rho)\right) + O\left(\frac{1}{N^2}\right) \\ =&\widetilde{\Phi}[\boldsymbol{F}](P, i) + O(\rho) + O\left(\frac{1}{N^2}\right). \end{aligned}$$

Then we can prove (13) analogously. ■

2 Equivariant Network Architectures

When the inputs and feature maps consist of multiple channels, we utilize multiple Ψ's and Φ's to process inputs and generate outputs. To be specific, for the input layer, where inputs s consist of M_s channels and the resulting feature maps $so^{(1)}$ consist of M_1 layer, we have

$$\begin{bmatrix} so_1^{(1)} \\ \vdots \\ so_{M_1}^{(1)} \end{bmatrix} = \sigma\left(\begin{bmatrix} \Psi_{11} & \cdots & \Psi_{1M_s} \\ \vdots & \ddots & \vdots \\ \Psi_{M_1 1} & \cdots & \Psi_{M_1 M_s} \end{bmatrix} \begin{bmatrix} s_1 \\ \vdots \\ s_{M_s} \end{bmatrix}\right).$$

For the following layer, where feature maps $so^{(l)}$ at the l-th layer consist of M_l channels, we have

$$\begin{bmatrix} so_1^{(l+1)} \\ \vdots \\ so_{M_{l+1}}^{(l+1)} \end{bmatrix} = \sigma\left(\begin{bmatrix} \Phi_{11}^{(l)} & \cdots & \Phi_{1M_l}^{(l)} \\ \vdots & \ddots & \vdots \\ \Phi_{M_{l+1}1}^{(l)} & \cdots & \Phi_{M_{l+1}M_l}^{(l)} \end{bmatrix} \begin{bmatrix} so_1^{(l)} \\ \vdots \\ so_{M_l}^{(l)} \end{bmatrix}\right).$$

Finally, we obtain a more general network architecture, and it is easy to verify that equivariance can still be preserved through this network.

Particularly, as for

$$\Phi[so](P_R, A_R) = \int_{SO(2)} \chi_A^{(A_R)} \ [so](P_R, A_R A) d\nu(A), \tag{16}$$

if we take $w_{A,i} = 0$ for any $A \in SO(2)$ and $i = 2, 3, \cdots, 6$, then (16) can be rewritten as

$$\Phi[so](P_R, A_R) = \int_{SO(2)} w_{A,1} so(P_R, A_R A) d\nu(A), \tag{17}$$

which is analogous to the conventional 1×1 convolution in planar CNNs.

3 Model Architectures and Training Details

In this section we provide network architectures and training details for reproducing our results in experiments. Each experiment is run for 5 times and implemented using Pytorch.

Spherical MNIST Classification

The small model consists of 4 convolution layers and 3 fully connected (FC) layers. The convolution layers have $8, 12, 16$ and 28 output channels, and the FC layers have $28, 28$ and 10 channels, respectively. The large model consists of 5 convolution layers and 3 fully connected (FC) layers. The convolution layers have $8, 12, 16, 24,$ and 48 output channels, and the FC layers have $48, 48$ and 10 channels, respectively. N is set to 16, and downsampling is performed after layer 2. In between convolution layers, we use batch normalization (Ioffe and Szegedy 2015) and ReLU nonlinearities.

The models are trained using the Adam algorithm (Kingma and Ba 2015). We use generalized He's weight initialization scheme introduced in (Weiler, Hamprecht, and Storath 2018) for the convolution layers and Xavier initialization (Glorot and Bengio 2010) for the FC layers. For N/R task, we use dropout for better generalization. We train using a batch size of 16 for 80 epochs, an initial learning rate of 0.01 and a step decay of 0.5 per 10 epochs. We use the cross-entropy loss for training the classification network.

Omnidirectional Image Segmentation

Following (Jiang et al. 2019), we preprocess the data into a spherical signal by sampling the original rectangular images at the latitude-longitudes of the spherical mesh vertex positions. The input RGB-D channels are interpolated using bilinear interpolation, and semantic labels are acquired using nearest-neighbor interpolation. The input and output spherical signals are at the level-5 resolution.

The network architecture is a residual U-Net (He et al. 2016; Ronneberger, Fischer, and Brox 2015) using PDO-eS^{2}Convs, which consists of an encoder and a decoder. The encoder network takes as input a signal at resolution $r = 5$. We use a similar network architecture as that in (Jiang et al. 2019) and the details are shown in Table 1, and N is set to 8 except the last layer. We use a trivial PDO-eS^{2}Conv ($N = 1$) for the last layer to obtain 15 output channels. Note that we use 15 output channels since the 2D-3D-S dataset has two additional classes (invalid and unknown) that are not evaluated for performance.

Table 1: The architecture of PDO-eS^{2}CNN used in the 2D-3D-S image segmentation experiments. a, b, c and s stands for input channels, bottleneck channels, output channels, and strides, respectively. When $s = 2$, downsampling is performed using average pooling, and when $s = 0.5$, upsampling is applied using linear interpolation.

Level	a	Block	b	c	s	N
5	4	Encoder	-	16	2	8
4	16	Encoder	16	32	2	8
3	32	Encoder	32	64	2	8
2	64	Decoder	-	64	0.5	8
3	64	Decoder	-	64	1	8
3	64×2	Decoder	32	32	0.5	8
4	32	Decoder	-	32	1	8
4	32×2	Decoder	16	16	0.5	8
5	16	Decoder	-	16	1	8
5	16×2	Decoder	16	16	1	8
5	16	PDO-eS^{2}Conv	-	8	1	8
5	8×8	PDO-eS^{2}Conv	-	15	1	1

We use the Adam optimizer to train our network for 200 epochs, with an initial learning rate of 0.01 and a step decay of 0.9 per 20 epochs. Following (Jiang et al. 2019), we use the weighted cross-entropy loss for training, and the loss for each class is of the following weighting scheme:

$$w_c = \frac{1}{1.02 + \log(f_c)},$$

where w_c is the weight corresponding to class c, and f_c is the frequency by which class c appears in the training set. We use zero weight for the two dropped classes (invalid and unknown). The detailed statistics for this task is shown in Tables 2 and 3.

Atomization Energy Prediction

Following (Cohen et al. 2018), we represent each molecule as a spherical signal. Specifically, we define a sphere S_i around p_i for each atom i. The radius is kept uniform across atoms and molecules, and chosen minimal such that no intersections among spheres happen. We define potential functions $U_z = \sum_{j \neq i, z_j = z} \frac{z_i z}{|x - p_i|}$ and produce a T channel spherical signal for each atom in the molecule. Finally, we represent these signals on a level-3 mesh.

The architecture used on QM7 dataset is shown in Table 4 and N is set to 8. We share weights among atoms making filters permutation invariant, by pushing the atom dimension into the batch dimension. We use global spatial pooling and orientation pooling after the last PDO-eS^{2}Conv. Next, we use DeepSet (Zaheer et al. 2017) to refine the resulting feature vectors. Both PDO-eS^{2}CNN and DeepSet are jointly optimized. Following (Cohen et al. 2018), we firstly train a simple MLP only on the 5 frequencies of atom types in a molecule, and then train our main model on the residual. Specifically, we use the Adam optimizer to train this model using a batch size of 32 for 30 epochs, an initial learning rate of 0.001 and a step decay of 0.1 per 10 epochs.

Table 2: mAcc comparison on 2D-3D-S dataset. Per-class accuracy is shown when available.

Model	Mean	beam	board	bookcase	ceiling	chair	clutter	column	door	floor	sofa	table	wall	window
UNet	50.8	17.8	40.4	59.1	91.8	50.9	46.0	8.7	44.0	94.8	26.2	68.6	77.2	34.8
UGSCNN (Jiang et al. 2019)	54.7	19.6	48.6	49.6	93.6	63.8	43.1	28.0	63.2	96.4	21.0	70.0	74.6	39.0
Icosahedral CNN (Cohen et al. 2019)	55.9	-	-	-	-	-	-	-	-	-	-	-	-	-
HexRUNet (Zhang et al. 2019)	58.6	**23.2**	56.5	**62.1**	**94.6**	66.7	41.5	18.3	**64.5**	96.2	**41.1**	**79.7**	77.2	41.1
PDO-eS^{2}CNN	**60.4**	22.2	**59.6**	59.7	93.5	**67.4**	**53.9**	**26.3**	64.1	**97.1**	30.8	75.4	**81.9**	**53.4**

Table 3: mIoU comparison on 2D-3D-S dataset. Per-class IoU is shown when available.

Model	Mean	beam	board	bookcase	ceiling	chair	clutter	column	door	floor	sofa	table	wall	window
UNet	35.9	8.5	27.2	30.7	78.6	35.3	28.8	4.9	33.8	89.1	8.2	38.5	58.8	23.9
UGSCNN (Jiang et al. 2019)	38.3	8.7	32.7	33.4	82.2	42.0	25.6	10.1	41.6	87.0	7.6	41.7	61.7	23.5
Icosahedral CNN (Cohen et al. 2019)	39.4	-	-	-	-	-	-	-	-	-	-	-	-	-
HexRUNet (Zhang et al. 2019)	43.3	10.9	39.7	37.2	**84.8**	**50.5**	29.2	11.5	45.3	**92.9**	**19.1**	49.1	63.8	29.4
PDO-eS^{2}CNN	**44.6**	**11.4**	**43.3**	**38.2**	83.9	50.3	**31.3**	**12.4**	**48.4**	90.0	18.1	**49.5**	**65.9**	**37.1**

Table 4: The architecture used in QM7 atomization energy prediction experiments. Downsampling is performed using average pooling.

PDO-eS^{2}CNN	Layer	Channels	Level
	PDO-eS^{2}Conv	16	3
	PDO-eS^{2}Conv	32	2
	PDO-eS^{2}Conv	64	1
	PDO-eS^{2}Conv	64	0
	Global orientation pooling		
	Global spatial pooling		
DeepSet	Layer	Input/Hidden	
	ϕ(MLP)	64/256	
	ψ(MLP)	64/512	

References

Cohen, T. S.; Geiger, M.; Kohler, J.; and Welling, M. 2018. Spherical CNNs. In *ICLR*.

Cohen, T. S.; Weiler, M.; Kicanaoglu, B.; and Welling, M. 2019. Gauge Equivariant Convolutional Networks and the Icosahedral CNN. In *ICML*, 1321–1330.

Glorot, X.; and Bengio, Y. 2010. Understanding the Difficulty of Training Deep Feedforward Neural Networks. In *AISTATS*, 249–256.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *CVPR*, 770–778.

Ioffe, S.; and Szegedy, C. 2015. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *ICML*, 448–456.

Jiang, C.; Huang, J.; Kashinath, K.; Prabhat; Marcus, P.; and Niessner, M. 2019. Spherical CNNs on Unstructured Grids. In *ICLR*.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In *ICLR*.

Ronneberger, O.; Fischer, P.; and Brox, T. 2015. U-net: Convolutional Networks for Biomedical Image Segmentation. In *MICCAI*, 234–241. Springer.

Weiler, M.; Hamprecht, F. A.; and Storath, M. 2018. Learning Steerable Filters for Rotation Equivariant CNNs. In *CVPR*, 849–858.

Zaheer, M.; Kottur, S.; Ravanbakhsh, S.; Poczos, B.; Salakhutdinov, R. R.; and Smola, A. J. 2017. Deep Sets. In *NeurIPS*, 3391–3401.

Zhang, C.; Liwicki, S.; Smith, W. A. P.; and Cipolla, R. 2019. Orientation-Aware Semantic Segmentation on Icosahedron Spheres. In *ICCV*, 3533–3541.