Circular-Symmetric Correlation Layer

Bahar Azari

Deniz Erdoğmuş

azari.b@northeastern.edu

Erdogmus@ece.neu.edu

Department of Electrical and Computer Engineering Northeastern University, USA Boston, MA 02115

Abstract

Despite the vast success of standard planar convolutional neural networks, they are not the most efficient choice for analyzing signals that lie on an arbitrarily curved manifold, such as a cylinder. The problem arises when one performs a planar projection of these signals and inevitably causes them to be distorted or broken where there is valuable information. We propose a Circular-symmetric Correlation Layer (CCL) based on the formalism of roto-translation equivariant correlation on the continuous group $S^1 \times \mathbb{R}$, and implement it efficiently using the well-known Fast Fourier Transform (FFT) algorithm. We showcase the performance analysis of a general network equipped with CCL on a popular autonomous driving dataset, nuScenes (Caesar et al., 2020), for semantic segmentation of 3D point clouds obtained from LiDAR sweeps from their 360° —panoramic projections.

1 Introduction

Planar convolutional neural networks, widely known as CNNs, are characterized by pattern-matching kernels that can identify motifs in the signal residing on a 2D plane. However, various applications exist in which signals lie on some curved planes, e.g., temperature and climate data on the surface of the (spherical) earth, or 360° —panoramic images obtained from LiDAR sweeps for semantic segmentation in autonomous driving applications.

Analyzing signals in these applications is achievable by using the planar projection of them. Specifically, for 360° —panoramic image processing, which is the interest of this study, the image is usually unwrapped to a standard 2D image to be treated as an input feature map. However, the resulting arbitrary breakage of the signal at the boundary may be destructive in object-detection tasks in terms of both information lost at the boundary and lack of equivariance to noticeable shifts. (see figure. 1).

A convolution kernel produces a single value associated with the region in the image covered by it at a specific shift. However, the area at the boundaries of the image is neglected as the kernel shift needs to stop at a margin equal to half of the size of the kernel. This is detrimental for panoramic image processing because of potentially valuable information that exists in the border of the image (The car in figure. 1). In addition, the ever-shrinking size of the middle layer feature maps prevents the formation of

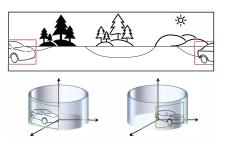


Figure 1: Object breakage in 360°-panoramic image unwrapping. **Top:** The car has been subjected to image cut. **Bottom:** Cognition tasks should be invariant to shifting of the object on the surface the cylinder.

Machine Learning for Autonomous Driving Workshop at the 35th Conference on Neural Information Processing Systems (NeurIPS 2021).

deeper networks. Zero-padding applied to the out-of-image regions solves the latter issue, but the introduced distortion propagates inward from the boundaries as we go deeper in the CNN. Other proxy techniques trying to alleviate the border information loss problem exists, such as input padding (e.g., Shi et al. 2015), but they increase the computational time and memory consumption.

Furthermore, a commonly neglected shortcoming of CNN becomes noticeable in the case of panoramic image processing where desired outputs should be immune to arbitrarily large rolling of the input image. This limitation is related to what is known as *invariance* and *equivariance* properties of the neural network as a function. For defining these properties, we consider a family, or a "group", of transformations (e.g., rotations, or translations) of input and output to a given layer of the neural network. The elements of the group can "act" on the input and output of each layer in some specific way. The neural network is invariant to the action of the group if transformations of the input do not change the output. Otherwise, it is equivariant if as we transform the input, the output is transformed according to some other action of the group. The convolution layers are empirically known to be invariant to small translations of their input image, but they are not completely immune to large shifts nonetheless (Goodfellow et al., 2009; Schmidt and Roth, 2012; He et al., 2015; Lenc and Vedaldi, 2015; Jaderberg et al., 2015; Cohen and Welling, 2016; Dieleman et al., 2016). In panoramic image processing, which can be achieved by a distortion-less projection of a cylindrical image onto a rectangular grid, arbitrary rotation around the principal axis of a cylindrical image manifests itself as a horizontal translation in a 2D grid. Therefore, utilizing planar CNN, with the aforementioned limitation, will not guarantee invariance to such transformation when required. Figure. 1(Bottom) shows an example of this phenomenon. The object (car) identification task should be invariant to rotation around the principal axis. These issues have been heretofore addressed by creating more training data using multiple circularly shifted versions of the original data, i.e., data augmentation (see Lo et al. 2002). Although this approach seems adequate to some extent, it increases the training time by inflicting sample complexity, does not always guarantee invariance (Elesedy and Zaidi 2021), and could have an adversary effect on the kernels' representational capacity as it exposes them to more border areas.

Nevertheless, because the building block of CNN (i.e., convolution or cross-correlation layer) has the potential equivariance property, we may exploit them to construct a network suitable for the translation-invariant tasks such as object detection in figure. 1. Therefore, for a systematic treatment of analyzing the 360° —panoramic data, we propose a circular-symmetric correlation Layer (CCL) based on the formalism of roto-translation equivariant correlation on the continuous group $S^1 \times \mathbb{R}$ — a group constructed of the unit circle and the real line. We implement this layer efficiently using the well-known Fast Fourier Transform (FFT) and discrete cosine transform (DCT) algorithms. We discuss how the FFT yields the exact calculation of the correlation along the panoramic direction due to its circular symmetry and guarantees the invariance to circular shift. The DCT provides an improved approximation to transnational symmetry compared to what we observe in CNNs. We showcase the performance analysis of a general network equipped with CCL on nuScenes (Caesar et al., 2020), a public large-scale dataset for autonomous driving, for LiDAR semantic segmentation. Our contributions are as follows:

- Theoretical treatment of circular-symmetric correlation on the surface of a cylinder.
- Efficient implementation of CCL based on FFT and DCT.
- Experimental results showing competitive performance of neural networks equipped with CCL.

2 Related Work

The outstanding ability of CNN in processing spatially and temporally correlated signals comes from the fact that it exploits the translational symmetry and equivariance property of its correlation layers. In other words, a trained kernel should be able to detect a particular pattern regardless of its specific location in the image. Due to this compelling property, there has been an increasing attempt to generalize the idea of CNN to other spaces and symmetry groups (Gens and Domingos, 2014; Olah, 2014; Dieleman et al., 2015; Guttenberg et al., 2016; Dieleman et al., 2016; Cohen and Welling, 2017; Ravanbakhsh et al., 2016; Zaheer et al., 2017; Ravanbakhsh et al., 2017; Worrall et al., 2017; Maron et al., 2020; Dym and Maron, 2021). Theoretical guarantee for generalization benefit of equivariant models was treated in (Elesedy and Zaidi, 2021).

Most of these studies focus on discrete groups. For example, the investigation of discrete 90° rotations acting on planar images in the work of (Cohen and Welling, 2016), permutations of nodes in graphs in (Maron et al., 2019), or permutations of points in the point cloud in (Zaheer et al., 2017). Recent works (such as (Cohen et al., 2018, 2019)) have been investigating equivariance to continuous groups and generalized the CNN to various spaces. (Kondor and Trivedi, 2018) and (Cohen et al., 2018) use the generalized Fourier transform for group correlation and provided a formalism to efficiently implement these layers. Circular symmetry, which is the interest of this paper, has also been empirically studied in (Schubert et al., 2019; Papadakis et al., 2010; Kim et al., 2020), but none of these works addressed the issue in a formal analytic way.

3 Circular-Symmetric Correlation Layer

To learn a function that predicts a quantity based on a spatially correlated signal such as an image, we need to perform cross-correlation (correlation, in short). Specifically, we slide a kernel (filter) throughout the signal and measure the similarity. We have the familiar case of a classical planar \mathbb{R}^2 correlation, in which the output value at translation $x \in \mathbb{R}^2$ is computed as an inner product between the input and a kernel, translated to x. However, correlation is not limited to signals on \mathbb{R}^2 , and in our case, we are interested in images on the surface of a cylinder. We begin our discussion by introducing the correlation on the surface of a cylinder. To do so, we start with defining its mathematical building blocks.

3.1 Preliminaries and Notation

Cylinder We consider the lateral surface of a cylinder, a manifold, which is constructed by the combination of two other manifolds – a circle and a line segment ¹. The unit circle S^1 , defined as the set of points $z \in \mathbb{R}^2$ with norm 1, is a one-dimensional manifold that can be parameterized by polar coordinate $\varphi \in [0, 2\pi]$. Cartesian product of S^1 with a line \mathbb{R} (or, a line segment (-a, a)) constructs a two-dimensional manifold, known as a cylinder $\mathbb{X} = S^1 \times \mathbb{R}$ (or, $S^1 \times (-a, a)$ in case of having a line segment). We characterize the set of points on the lateral surface of the cylinder by cylindrical coordinates $\varphi \in [0, 2\pi]$ and $z \in \mathbb{R}$ and define circular-symmetric signals and convolution kernels as continuous functions on this surface $f : \mathbb{X} \mapsto \mathbb{R}^K$, where K is the number of channels.

Rotation and Translation on Cylinder surface The set of rotations around and translations along the z-axis is a subgroup of SE(3), the "special Euclidean group", denoted as $\mathcal{G} \leq \text{SE}(3)$ and is isomorphic to \mathbb{X} , i.e., $\mathcal{G} = S^1 \times \mathbb{R}$. The action of an element ξ in \mathcal{G} is a pair (R_{ψ}, ν) , where R_{ψ} belongs to a subgroup of the "special orthogonal group" SO(3) representing a rotation by ψ around z-axis, and a translation by $\nu \in \mathbb{R}$ along z-axis. The representation of \mathcal{G} corresponds to the set of all 4×4 transformation matrices of the form

$$\mathcal{G} = \left\{ \begin{pmatrix} R\psi & 0 \\ 0 & \nu \\ 0 & 0 & 0 & 1 \end{pmatrix} \middle| \psi \in [0, 2\pi] \text{ and } \nu \in \mathbb{R} \right\},\tag{1}$$

where R_{ψ} is a 3D rotation matrix. In this study, we consider filters and functions on the cylindrical surface corresponding to applying the roto-translation operator L_{ξ} which takes a function $f: \mathbb{X} \mapsto \mathbb{R}^K$ and produces a shifted version by rotating it around and translating it along the principal axis:

$$[L_{\xi}f](x) = f(\xi^{-1}x).$$
 (2)

As we explained earlier, since \mathcal{G} is a group and groups contain inverses, for $\xi, \xi' \in \mathcal{G}$ we have $L_{\xi\xi'} = L_{\xi}L_{\xi'}$. We show this using inverse and associative properties of groups:

$$[L_{\xi\xi'}f](x) = f((\xi\xi')^{-1}x) = f(\xi'^{-1}(\xi^{-1}x))$$

= $[L_{\xi'}f](\xi^{-1}x) = [L_{\xi}L_{\xi'}f](x)$. (3)

¹It is either an infinite line or a line segment without its endpoints which is also a manifold.

3.2 Correlation on Cylinder

To define the correlation we begin with the established definition of the inner product. The inner product on the vector space of cylindrical signals is characterized:

$$\langle f, h \rangle = \int_{\mathbb{X}} \sum_{k=1}^{K} f_k(x) h_k(x) dx, \tag{4}$$

where the integration measure dx denotes the Haar measure (invariant integration measure) on the lateral surface of the cylinder and it is equal to $d\varphi dz$ in cylindrical coordinate. Due to the invariance of the measure, the value of the integral of a function affected by any $\xi \in \mathcal{G}$ remains the same, namely, $\int_{\mathbb{X}} f(\xi x) dx = \int_{\mathbb{X}} f(x) dx$ for all $\xi \in \mathcal{G}$. Using the inner product in (4), we define the correlation of signals and filters on the surface of the cylinder. Given a point on the cylinder $x \in \mathbb{X}$, a transformation on the subgroup of SE(3), $\xi \in \mathcal{G}$, and functions f(x) and h(x), the correlation is defined:

$$[f \star h](\xi) = \langle L_{\xi}f, h \rangle = \int_{\mathbb{X}} \sum_{k=1}^{K} f_k(\xi^{-1}x) h_k(x) dx.$$
 (5)

Note that the correlation in (5) is also equivalent to $\langle f, L_{\xi^{-1}} h \rangle$ as the value of the correlation at a shift ξ is equal to the inner product of f and h, where either f is shifted by ξ , or h is shifted by the inverse of ξ (ξ^{-1}). Therefore, if we express the point x as $x = (\varphi, z)$, the transformation as $\xi = (\psi, \nu)$, and the Haar measure as $dx = d\varphi dz$, the correlation in (5) can be rewritten as:

$$[f \star h](\xi) = \langle L_{\xi}f, h \rangle = \int_{\mathbb{R}} \int_{0}^{2\pi} \sum_{k=1}^{K} f_{k}(\varphi - \psi, z - \nu) h_{k}(\varphi - \psi, z - \nu) d\varphi dz, \tag{6}$$

where the integral with respect to φ is the circular cross-correlation. It is worthwhile to mention that the resulting correlation function lies on the group \mathcal{G} which is isomorphic to the space \mathbb{X} that the initial functions have lied on, namely $S^1 \times \mathbb{R}$.

3.3 Equivariance of Correlation Layers

For the correlation in (6), defined in terms of the roto-translation operator L_{ξ} , we can show the crucial equivariance property known for all convolution and correlation layers. We express mathematically what we informally stated earlier.

Group actions: For a set of points \mathbb{X} , we have a group \mathcal{G} that acts on \mathbb{X} . This means that for each element $\xi \in \mathcal{G}$, there exist a transformation $T_{\xi} : \mathbb{X} \to \mathbb{X}$ corresponding to group action $x \mapsto T_{\xi}(x)$. We showed this simply as ξx to simplify notation. As we have seen earlier, the action of \mathcal{G} on \mathbb{X} extends to functions on \mathbb{X} (induced action) and that is what we have denoted as the operator $L_{\xi} : f \mapsto f'$ which is $f'(x) = [L_{\xi}f](x) = f(\xi^{-1}x)$.

Equivariance: Equivariance is the potential property of a map between functions on a pair of spaces with respect to a group acting on these spaces through the group action.

Definition 1. Let X_1 , X_2 be two sets with group \mathcal{G} acting on them. Consider V_1 and V_2 as the corresponding vector spaces of functions defined on these sets, and L_{ω} and L'_{ω} as the induced actions of \mathcal{G} on functions. We say that a map $\Phi: V_1 \to V_2$ is \mathcal{G} -equivariant if

$$\Phi(L_{\omega}(f)) = L'_{\omega}(\Phi(f)) \quad \forall f \in V_1, \ \forall \omega \in \mathcal{G}.$$

Considering that the map in our case corresponds to the cross-correlation function we have defined on the cylindrical surface in (5), its equivariance with respect to the action of the group $\mathcal{G} = S^1 \times \mathbb{R}$ can be demonstrated as follows:

Theorem 2. Cross-correlation on lateral surface of a cylinder is equivariant to the action of the group $S^1 \times \mathbb{R}$.

Proof. Given that the group G of transformations on the cylinder surface is isomorphic to the set of points on the cylindrical manifold, we have:

$$[h \star L_{\omega} f](\xi) \stackrel{\text{by (5)}}{=} \langle L_{\xi} h, L_{\omega} f \rangle = \langle L_{\omega^{-1}} L_{\xi} h, f \rangle$$

$$\stackrel{\text{by (3)}}{=} \langle L_{\omega^{-1} \xi} h, f \rangle = [h \star f](\omega^{-1} \xi)$$

$$\stackrel{\text{by (2)}}{=} [L_{\omega} [h \star f]](\xi),$$

where $[h\star.](\xi)$ is the cross-correlation function, and L_{ω} is a transformation operator. Note that in our case $L_{\omega}=L'_{\omega}$. Equivariance can be represented graphically by commutative diagram as:

$$\begin{array}{ccc}
f & \xrightarrow{L_{\omega}} & L_{\omega}f \\
[h \star .](\xi) & & \downarrow [h \star .](\xi) \\
[h \star f](\xi) & \xrightarrow{L_{\omega}} & [L_{\omega}[h \star f]](\xi)
\end{array}$$

3.4 Implementing CCL using FFT

Computing cross-correlation and convolution using the Fast Fourier Transform (FFT) is known to be more efficient than their direct calculation. This is an important result of the Convolution theorem, according to which, the cross-correlation between two signals is equal to the product of the Fourier transform of one signal multiplied by the complex conjugate of Fourier transform of the other signal, or mathematically, $\widehat{f * g} = \widehat{f} \odot \widehat{g}$, where \odot is the element-wise product. Fourier transform is a linear projection of a function onto a set of orthogonal basis functions. For the real line (\mathbb{R}) and the circle (S^1), these basis functions are the familiar complex exponentials $\exp(\imath n\theta)$, where $\imath = \sqrt{-1}$.

The input of the CCL is the spatial signal f on \mathbb{X} , sampled on a discrete grid of the cylindrical coordinate (φ,z) . This signal is periodic in φ due to the 2D image being wrapped around a cylindrical manifold, and it is finite along z. Therefore, the convolution theorem holds for the dimension along unwrapped φ , and it is appropriate to use FFT for implementing the correlation in this dimension. However, we do not have the same periodicity in the z dimension. Hence, we use another set of basis functions (i.e., cosine waves), and as a consequence, we use discrete cosine transform (DCT) in the z dimension. As opposed to FFT, which is related to Fourier series coefficients of a periodically extended sequence, DCT Muchahary et al. (2015) is associated with Fourier series coefficients of a periodically and symmetrically extended sequence, yields a continuous extension at the boundaries. As shown in figure. 2 for a 360° -panoramic image, by applying FFT along unwrapped φ a circular symmetry is evoked along the horizontal axis, and by applying DCT along z dimension a reflection symmetry is evoked along the vertical axis, which imply smooth boundaries in both dimensions. We will show in the experiments that the usage of DCT in this setting benefits the overall performance of the deep learning module in terms of vertical translation. We compute DCT by using N-FFT Makhoul (1980) in which a signal $f = \left[f_n|_{n=1}^N\right]$ is organized as $\hat{f} = \left[f_{2n-1}|_{n=1}^{N/2}, f_{N-2n+2}|_{n=1}^{N/2}\right]$ and FFT is applied to the resulting signal:

$$\mathrm{DCT}(f) = \Re \Big(\big[2e^{\frac{-j\pi n}{2N}} \big|_{n=1}^{N} \big] \odot \mathrm{FFT}(\hat{f}) \Big),$$

where \odot and \Re denote element-wise multiplication and real part, respectively, and $e^{\frac{-j\pi n}{2N}}$ is a half-sample shift. The CCL class computation graph is summarized in algorithm. 1.

3.5 Computational Complexity of CCL vs. CNN

The computational complexity of applying a Conv2d filter with kernel size K in a CNN to a panoramic image of size $H \times W$ is $O(WHK^2)$, whereas for CCL is $O(WH\log_2 W)$ and is independent of the kernel size. For a typical panoramic image of width W=1024 with kernel size K=5, $\log_2 W < K^2$ and therefore CCL performs relatively faster. This becomes more evident when using larger kernel sizes (e.g., in favor of a shallower network) or in deeper layers where kernel size becomes relatively larger w.r.t. image dimensions after applying pooling layers.

Algorithm 1: CCL class

```
 \begin{split} \textbf{Class} & \, \texttt{CCL}(C_{\text{IN}}, C_{\text{OUT}}, k_{\text{H}}, k_{\text{W}}, s_{\text{H}}, s_{\text{W}}) \colon \\ & \quad \textbf{Parameters:} \, \, W \in \mathbb{R}^{C_{\text{OUT}} \times C_{\text{IN}} \times k_{\text{H}} \times k_{\text{W}}}, b \in \mathbb{R}^{C_{\text{OUT}}} \\ & \quad \textbf{Input:} \, \, X \in \mathbb{R}^{C_{\text{IN}} \times H \times W} \\ & \quad \textbf{Output:} \, \, F \in \mathbb{R}^{C_{\text{OUT}} \times \left\lceil \frac{H}{s_{\text{H}}} \right\rceil \times \left\lceil \frac{W}{s_{\text{W}}} \right\rceil} \\ & \quad \textbf{Def Forward}(X) \colon \\ & \quad X \leftarrow \texttt{FFT}_h \big( \texttt{DCT}_v(X) \big) \\ & \quad W \leftarrow \texttt{FFT}_h \big( \texttt{DCT}_v(W) \big) \\ & \quad F \leftarrow \sum_{C_{\text{IN}}} X \odot W \\ & \quad F \leftarrow \texttt{FFT}_h^{-1} \big( F \big) \\ & \quad F \leftarrow \texttt{DCT}_v^{-1} \big( F \big) \\ & \quad \textbf{return subsample} \big( F, s_{\text{H}}, s_{\text{W}} \big) \end{split}
```

4 Experiments

We demonstrate the accuracy and effectiveness of the CCL layer in comparison with the standard convolution layer by evaluating it over a couple of well-known datasets such as MNIST and CIFAR10. We then provide an application example for adopting CCL in designing neural networks for LiDAR semantic segmentation in autonomous driving.

4.1 Invariance Analysis of Networks Built with CCL

We first evaluate the equivariance performance of a neural network equipped with CCL to rotations of the input along the z-axis. We propose a version of MNIST and CIFAR10 datasets called Rolled MNIST (\mathcal{R} MNIST) and Rolled CIFAR10 (\mathcal{R} CIFAR10), respectively, wrapped around a cylindrical surface as shown in figure. 3. In these datasets, we augment the actual MNIST and CIFAR10 datasets with the horizontally rolled version of the original images using random samples of $\varphi \in [0, 2\pi]$ (see figure. 3). Therefore, for a standard image size of 28×28 , the rotation by $\pi/2$ is equivalent to shifting the image horizontally by $\pi/2 \times 28/2\pi = 7$. Consequently, the boundary cut of the image can pass through and destruct the consistency of the object (e.g., the digits in the MNIST dataset or the animal in the CIFAR10 dataset).

We perform three testing experiments using the actual datasets or their rolled versions and report the results in table. 1. Note that in our experiments, we do not aim for optimizing the architectures for the best accuracy. Our goal is to demonstrate how a neural network equipped with CCL can outperform regular CNN in terms of accuracy for tasks requiring equivariance. In the case of training and testing with the original MNIST and CIFAR10, the performance of a neural network using CCL is comparable to its CNN counterpart, although the CCL network slightly outperforms. However, if we train these two neural networks on the original non-augmented datasets and test them on the \mathcal{R} MNIST and \mathcal{R} CIFAR10, we see a considerable performance drop for CNN. The reason is that CNNs cannot handle a considerable degree of image translation. The accuracy of the CNN improves

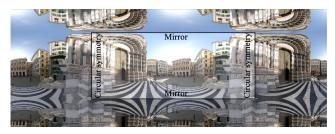


Figure 2: 360° – panoramic image with circular symmetry along the horizontal axis (unwrapped φ) and reflection symmetry along the vertical axis which are evoked by FFT and DCT, respectively.

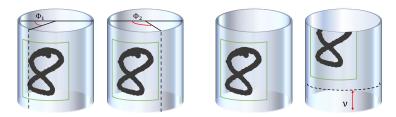


Figure 3: \mathcal{R} MNIST. For the dataset to be representative of all the define transformations mentioned in the paper, namely, rotation around the z-axis and translation along the z-axis, we randomly generated the discretised rolls ($\varphi_i \in [0, 2\pi]$ with step size of 1/28). **left**: panoramic boundary cut is rolled from φ_1 to φ_2 . **right**: The image is translated north by ν .

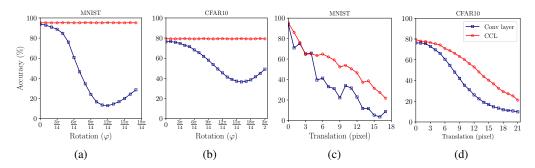


Figure 4: Accuracy of the neural networks trained on MNIST and CIFAR10 and tested on the rolled and translated version. The two left figures show the accuracy performance of the models versus the rotation of the images around the principal axis (z-axis). The two right figures show the accuracy performance of the models versus translation along the z-axis. We observe that the CCL layer is exactly equivariant to S^1 , and it demonstrates a greater degree of equivariance compared to its counterpart (conv2d) to the translation along the z-axis..

by training on the augmented version of the datasets, however, it is still considerably lower than that of CCL. Also, note that training with the augmented dataset is significantly slower as it contains several times more samples, i.e., for each rotation. To see the adopted architectures refer to table. 2. To make the learned representation invariant to the rotation around the *z*-axis, a global average pooling layer is used between the correlation and fully-connected layers (see Lin et al. 2013).

We show another set of results comparing the equivariance of neural networks adopting CCL layers and regular CNN. We adopt similar network architectures described in table. 2. CCL(M) corresponds to the usage of the CCL layer with an output channel size of M. For the regular CNN, we replace the CCL with the Conv2d layer and keep everything else the same. Figure. 4 shows the accuracy of the CCL neural network (red) and CNN (blue) trained on MNIST and CIFAR10 datasets and tested on their rolled and translated versions. The two left figures show the accuracy performance of the models versus different degrees of rotation of the images around the principal axis (z-axis). It is obvious that the CCL neural network trained only on the unperturbed data generalizes quite well in all the rotations of the test data, hence the flat red line. Nonetheless, CNN performance drops as the rotation value increases to the point where the image begins to roll back to its original position, hence the sharp drop of the blue line. The two right figures show the accuracy performance of the models versus translation along z-axis. For a finite signal, the equivariance property does not hold for the translations along the z-axis. Therefore, although the CCL layer is exactly equivariant to S^1 , it is not completely equivariant to vertical translation. However, networks equipped by the CCL layer demonstrate a greater degree of equivariance compared to their counterpart (conv2d), which is the consequence of using DCT in implementing the CCL layer. Specifically, because DCT exploits an even reflection symmetry of the images, objects remain more consistent along the upper and lower edges of the image (see figure. 2).

Table 1: Accuracy results for network using CCL and Conv2d layers.

			,		CIFAR10 RCIFAR10	,	
Conv2d	93.96	16.81	46.64	76.29	49.11	66.07	73.54
CCL	95.27	95.15	95.35	79.40	79.24	79.02	76.43

Table 2: Network architectures. $CCL(c_{OUT})$: c_{OUT} implies number of output channels. $FC(l_{IN}, l_{OUT})$: l_{IN} and l_{IN} imply input and output features dimensions, respectively. MaxPool(k, s): k and s imply kernel and stride sizes, respectively. AvgPool(k): k implies kernel sizes. The global average pooling makes the network invariant to the input roll.

Layer	MNIST	CIFAR10	nuScenes LiDAR
Input	$f \in \mathbb{R}^{1 \times 28 \times 28}$	$f \in \mathbb{R}^{3 \times 32 \times 32}$	$f \in \mathbb{R}^{4 \times 40 \times 360}$
1	CCL(8), ReLU	CCL(128), ReLU	CCL(32), ReLU
2	CCL(8), ReLU	CCL(128), ReLU	CCL(32), ReLU
3	MaxPool(2, 2)	MaxPool(2, 2)	CCL(32), ReLU
4	CCL(8), ReLU	CCL(128), ReLU	CCL(32), ReLU
5	CCL(8), ReLU	CCL(256), ReLU	CCL(6), Softmax
6	MaxPool(2, 2)	MaxPool(2, 2)	
7	CCL(10), ReLU	AvgPool(8)	
8	AvgPool(7), Softmax	FC(256, 120), ReLU	
9		FC(120, 84), ReLU	
10		FC(84, 10), Softmax	

For regular CNN, the CCL layers are replaced with Conv2d layers.

4.2 Application to LiDAR semantic segmentation

We evaluated the CCL in LiDAR semantic segmentation from a well-known autonomous driving dataset, nuScenes (Caesar et al., 2020). This LiDAR dataset consists of 40,000 LiDAR point clouds (34,000 train and 6,000 test), obtained from LiDAR sweeps, with 32 highly imbalanced semantic labels. The LiDAR's vertical field of view (FOV) is between -10° to 30° and its horizontal FOV is between -180° to 180° . We projected each point cloud to a panoramic image of height 40 = 30 - (-10) and width 360 = 180 - (-180) considering a vertical and horizontal resolution of 1° , and encoded 3D coordinates of each point as well as its LiDAR intensity in the four channels of this panoramic image, resulting in a 360° -panoramic image of size $4 \times 40 \times 360$ for each point cloud along with its ground-truth annotation. Furthermore, we merged the labels into six dominant classes of background, pedestrian, bicycle/motorcycle, movable object, bus/truck, and car. We adopted the architecture shown in table. 2 for both CCL and CNN with a weighted cross-entropy loss to account for class imbalance. We report balanced accuracy (average of the recalls for all classes) in table. 1, which shows the superior performance of CCL against CNN. In figure. 5, we show an instance of the segmentation performance for CCL and Conv2d CNN along with the ground truth (GT) labels. CCL performs better particularly on the vertical borders.

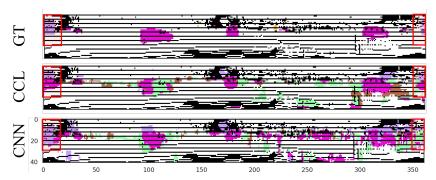


Figure 5: Segmentation results compared to ground truth (GT). CCL performs better particularly on the vertical borders.

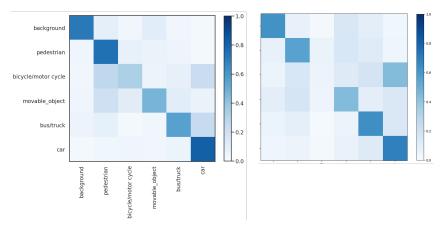


Figure 6: Confusion matrix, CCL (Left), Conv2d (Right)

The Confusion matrices associated with networks equipped with CCL and Conv2d are shown in figure. 6. The CCL network has done a better job classifying the labels, specifically regarding the "bicycle/motorcycle" label which is mainly confused with the "car" label.

5 Discussion and Conclusion

We have proposed a Circular-symmetric Correlation Layer (CCL) based on the formalism of roto-translation equivariant correlation on the continuous group $S^1 \times \mathbb{R}$, and implement it efficiently using the well-known FFT and DCT algorithm. Our numerical results demonstrate the effectiveness and accuracy obtained from adopting the CCL layer. A neural network equipped with CCL generalizes across rotations around the principal axis and outperforms its CNN counterpart. Note that the achieved gain is not at the expense of increasing the number of parameters (by zero- or input-padding of the input data) or data augmentation and hence longer training time and sample complexity. It is merely due to the intrinsic property of the CCL layer in mimicking the circular symmetry and reflection symmetry in the data.

References

Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 11618–11628. IEEE Computer Society, 2020.

Taco Cohen and Max Welling. Group equivariant convolutional networks. In *International conference* on machine learning, pages 2990–2999, 2016.

Taco S Cohen and Max Welling. Steerable cnns. In *International Conference on Learning Representations*, 2017.

Taco S Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical cnns. In *International Conference on Learning Representations*, 2018.

Taco S Cohen, Mario Geiger, and Maurice Weiler. A general theory of equivariant cnns on homogeneous spaces. *In Advances in Neural Information Processing Systems (NeurIPS)*, 32, 2019.

Sander Dieleman, Kyle W Willett, and Joni Dambre. Rotation-invariant convolutional neural networks for galaxy morphology prediction. *Monthly notices of the royal astronomical society*, 450(2): 1441–1459, 2015.

Sander Dieleman, Jeffrey De Fauw, and Koray Kavukcuoglu. Exploiting cyclic symmetry in convolutional neural networks. In *International conference on machine learning*, pages 1889–1898. PMLR, 2016.

- Nadav Dym and Haggai Maron. On the universality of rotation equivariant point cloud networks. In *International Conference on Learning Representations*, 2021.
- Bryn Elesedy and Sheheryar Zaidi. Provably strict generalisation benefit for equivariant models. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 2959–2969. PMLR, 2021.
- Robert Gens and Pedro M Domingos. Deep symmetry networks. *Advances in neural information processing systems*, 27:2537–2545, 2014.
- Ian Goodfellow, Honglak Lee, Quoc Le, Andrew Saxe, and Andrew Ng. Measuring invariances in deep networks. *Advances in neural information processing systems*, 22:646–654, 2009.
- Nicholas Guttenberg, Nathaniel Virgo, Olaf Witkowski, Hidetoshi Aoki, and Ryota Kanai. Permutation-equivariant neural networks applied to dynamics prediction. *arXiv* preprint *arXiv*:1612.04530, 2016.
- Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE transactions on pattern analysis and machine* intelligence, 37(9):1904–1916, 2015.
- Max Jaderberg, Karen Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *Proceedings of the 28th International Conference on Neural Information Processing Systems*, pages 2017–2025, 2015.
- Jinpyo Kim, Wooekun Jung, Hyungmo Kim, and Jaejin Lee. Cycnn: A rotation invariant cnn using polar mapping and cylindrical convolution layers. *arXiv* preprint arXiv:2007.10588, 2020.
- Risi Kondor and Shubhendu Trivedi. On the generalization of equivariance and convolution in neural networks to the action of compact groups. In *International Conference on Machine Learning*, pages 2747–2755, 2018.
- Karel Lenc and Andrea Vedaldi. Understanding image representations by measuring their equivariance and equivalence. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 991–999, 2015.
- Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. *arXiv preprint arXiv:1312.4400*, 2013.
- Shih-Chung B Lo, Huai Li, Yue Wang, Lisa Kinnard, and Matthew T Freedman. A multiple circular path convolution neural network system for detection of mammographic masses. *IEEE transactions on medical imaging*, 21(2):150–158, 2002.
- John Makhoul. A fast cosine transform in one and two dimensions. *IEEE Transactions on Acoustics*, *Speech, and Signal Processing*, 28(1):27–34, 1980.
- Haggai Maron, Heli Ben-Hamu, Nadav Shamir, and Yaron Lipman. Invariant and equivariant graph networks. In *International Conference on Learning Representations*, 2019.
- Haggai Maron, Or Litany, Gal Chechik, and Ethan Fetaya. On learning sets of symmetric elements. In *International Conference on Machine Learning*, pages 6734–6744. PMLR, 2020.
- Deboraj Muchahary, Abir J Mondal, Rajesh Singh Parmar, Amlan Deep Borah, and Alak Majumder. A simplified design approach for efficient computation of dct. In 2015 Fifth International Conference on Communication Systems and Network Technologies, pages 483–487. IEEE, 2015.
- Christopher Olah. Groups and group convolutions, 2014. URL https://colah.github.io/posts/2014-12-Groups-Convolution/.
- Panagiotis Papadakis, Ioannis Pratikakis, Theoharis Theoharis, and Stavros Perantonis. Panorama: A 3d shape descriptor based on panoramic views for unsupervised 3d object retrieval. *International Journal of Computer Vision*, 89(2):177–192, 2010.

- Siamak Ravanbakhsh, Jeff Schneider, and Barnabas Poczos. Deep learning with sets and point clouds. *arXiv preprint arXiv:1611.04500*, 2016.
- Siamak Ravanbakhsh, Jeff Schneider, and Barnabas Poczos. Equivariance through parameter-sharing. In *International Conference on Machine Learning*, pages 2892–2901. PMLR, 2017.
- Uwe Schmidt and Stefan Roth. Learning rotation-aware features: From invariant priors to equivariant descriptors. In 2012 IEEE Conference on Computer Vision and Pattern Recognition, pages 2050–2057. IEEE, 2012.
- Stefan Schubert, Peer Neubert, Johannes Pöschmann, and Peter Protzel. Circular convolutional neural networks for panoramic images and laser data. In 2019 IEEE Intelligent Vehicles Symposium (IV), pages 653–660. IEEE, 2019.
- Baoguang Shi, Song Bai, Zhichao Zhou, and Xiang Bai. Deeppano: Deep panoramic representation for 3-d shape recognition. *IEEE Signal Processing Letters*, 22(12):2339–2343, 2015.
- Daniel E Worrall, Stephan J Garbin, Daniyar Turmukhambetov, and Gabriel J Brostow. Harmonic networks: Deep translation and rotation equivariance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5028–5037, 2017.
- Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Russ R Salakhutdinov, and Alexander J Smola. Deep sets. In *Advances in neural information processing systems*, pages 3391–3401, 2017.