Appendix

A. Impact of Stride on Shift-Equivariance

To understand why the stride is the root of the issue regarding shift-equivariance, let the downsampling be reformulated as an operator (e.g., max, average, convolution) applied with a stride of 1 followed by subsampling depending on the subsampling factor. In this case, the subsampling scheme is hard-wired and returns the values at the indices modulo the scale factor. For instance, consider a 1D signal $f[n] = n$ and a max-pooling operation with a window size of 2. We first compute the stride-1 pooled signal: $g[n] = \max\big(f[n],\ f[n{+}1]\big)$, and then sub-sample by taking every other value:

$$s[k] = g[2k] = \big[\max(f[0], f[1]),\ ,\ldots\big] = [1, 3, \ldots]$$

By shifting the input by one, we get $f'[n] = f[n{+}1]$, which yields

$$s'[k] = g'[2k] = \big[\max(f'[0], f'[1]),\ \ldots\big] = [2, 4, \ldots]$$

If pooling were shift-equivariant we would have $s'[k] = s[k{+}1]$, but $s'[0] = 2 \neq 3 = s[1]$, demonstrating that fixed-index sub-sampling breaks shift-equivariance. This fixed strategy, independent of the input representation, cannot be shift-invariant/equivariant.

B. Proofs of Claim 1-3

Claim 1 The following proof is an extension of the real-valued demonstration provided in [44, Claim 1].

Proof. Let $\hat{\mathbf{z}} \triangleq \mathbf{T}_N\mathbf{z}$ be a shifted version of $\mathbf{z} \in \mathbb{C}^N$. Recall $\mathrm{PD}(\mathbf{z})$ and $\mathrm{PD}(\hat{\mathbf{z}})$ are defined as:

$$\mathrm{PD}(\mathbf{z}) \triangleq \mathrm{Poly}_{k^*}(\mathbf{z}) \tag{15}$$

$$\mathrm{PD}(\hat{\mathbf{z}}) \triangleq \mathrm{Poly}_{\hat{k}^*}(\hat{\mathbf{z}}) \tag{16}$$

where $k^* = \arg\max_{k\in\{0,1\}} p_{\mathbf{z}}\left(K = k|\mathbf{z}\right)$, and $\hat{k}^* = \arg\max_{k\in\{0,1\}} p_{\mathbf{z}}\left(K = k|\hat{\mathbf{z}}\right)$. Claim 1 states that $p_{\mathbf{z}}$ is shift-permutation-equivariant i.e. that $\hat{k}^* = \pi(k^*) = 1 - k^*$. So, from Eq. (8), we show that:

$$\begin{aligned}\mathrm{PD}\left(\mathbf{T}_N\,\mathbf{z}\right) &= \begin{cases}\mathrm{Poly}_1(\mathbf{z}) & \text{if } k^* = 1\,,\\ \mathbf{T}_M\mathrm{Poly}_0(\mathbf{z}) & \text{if } k^* = 0\,.\end{cases}\\ &= \left((1-k^*)\mathbf{T}_M + k^*\mathbf{I}\right)\cdot\mathrm{PD}(\mathbf{z})\end{aligned} \tag{17}$$

with $M = \lfloor N/2 \rfloor$, showing that PD satisfies the shift-equivariance definition. □

Claim 2 The following proof is an extension of the real-valued demonstration provided in [44, Claim 3].

Proof. Denote a feature map $\mathbf{z}$ and its shifted version $\hat{\mathbf{z}} \triangleq \mathbf{T}_N\mathbf{z}$, such as using Eq. (10) we got:

$$p_{\mathbf{z}}(K = \pi(k)|\mathbf{T}_N\mathbf{z}) = \frac{\exp\left(f(\mathrm{Poly}_k(\mathbf{T}_N\mathbf{z}))\right)}{\sum\limits_{j\in\{0,1\}}\exp\big(f(\mathrm{Poly}_j(\mathbf{T}_N\mathbf{z}))\big)}\,. \tag{18}$$

From Eq. (8) and given f shift-invariant, we obtain:

$$f(\mathrm{Poly}_{\pi(k)}(\mathbf{T}_N\mathbf{z})) = f(\mathbf{T}_M\mathrm{Poly}_k(\mathbf{z})) = f(\mathrm{Poly}_k(\mathbf{z})) \tag{19}$$

Finally, we have:

$$p_{\mathbf{z}}(K = \pi(k)|\mathbf{T}_N\mathbf{z}) = \frac{\exp\left(f(\mathrm{Poly}_k(\mathbf{z}))\right)}{\sum\limits_{j\in\{0,1\}}\exp\big(f(\mathrm{Poly}_j(\mathbf{z}))\big)} \tag{20}$$

$$= p_{\mathbf{z}}(K = k|\mathbf{z}) \tag{21}$$

□

Claim 3 The following proof is inspired by the real-valued demonstration provided in [44, Claim 4]. Nevertheless, we provide a simpler and more general demonstration.

Proof. First, let $N = 2M$, note that for any vector $\mathbf{y} \in \mathbb{C}^M$,

$$\mathrm{IPoly}_1(\mathbf{T}_M\mathbf{y}) = \mathbf{T}_N\mathrm{IPoly}_0(\mathbf{y}) \tag{22}$$

$$\mathrm{IPoly}_0(\mathbf{y}) = \mathbf{T}_N\mathrm{IPoly}_1(\mathbf{y}) \tag{23}$$

Then, denote a feature map $\mathbf{z}$ and its shifted version $\hat{\mathbf{z}} \triangleq \mathbf{T}_N\mathbf{z}$, and $k^* = \arg\max_{k\in\{0,1\}} p_\theta(K = k|\mathbf{z})$. We define, $\mathbf{u} \triangleq \mathrm{PU}\circ\mathrm{PD}(\mathbf{z})$ and $\hat{\mathbf{u}} \triangleq \mathrm{PU}\circ\mathrm{PD}(\mathbf{T}_N\mathbf{z})$. Since $p_{\mathbf{z}}$ is shift-permutation-equivariant, the selection index for $\mathbf{T}_N\mathbf{z}$ is switched to $\hat{k}^* = \pi(k^*) = 1 - k^*$. From Eq. (17), we have,

$$\hat{\mathbf{u}} = \mathrm{IPoly}_{\hat{k}^*}\left(\left((1-k^*)\mathbf{T}_M + k^*\mathbf{I}\right)\cdot\mathrm{PD}(\mathbf{z})\right) \tag{24}$$

In the case $k^* = 0$, we have,

$$\begin{aligned}\hat{\mathbf{u}} &= \mathrm{IPoly}_1\left(\mathbf{T}_M\mathrm{Poly}_0(\mathbf{z})\right) = \mathbf{T}_N\mathrm{IPoly}_0\left(\mathrm{Poly}_0(\mathbf{z})\right)\\ &= \mathbf{T}_N\mathbf{u},\end{aligned} \tag{25}$$

and for the case $k^* = 1$,

$$\begin{aligned}\hat{\mathbf{u}} &= \mathrm{IPoly}_0\left(\mathrm{Poly}_1(\mathbf{z})\right) = \mathbf{T}_N\mathrm{IPoly}_1\left(\mathrm{Poly}_1(\mathbf{z})\right)\\ &= \mathbf{T}_N\mathbf{u},\end{aligned} \tag{26}$$

showing that $\mathrm{PU}\circ\mathrm{PD}$ is shift-equivariant. □

C. Gumbel Softmax

We first provide a step-by-step integration process to highlight how the Gumbel Max-Trick works [20, 39]. We remind that the Cumulative Density Function (CDF) of the standard Gumbel distribution is $F(z) = \exp\left(-\exp(-z)\right)$ and its corresponding Probability Density Function (PDF) is $f(z) = \exp\left(-z - \exp(-z)\right)$.

The probability $\mathbb{P}\left(j = \arg\max_i (g_i + \log \pi_i)\right)$ defined in Eq. (12) where $x_i = \log \pi_i$ can be rewritten as:

$$\prod_{i \neq j} \mathbb{P}\left(g_j + \log \pi_j > g_i + \log \pi_i\right) . \tag{27}$$

This probability can be expressed with the Gumbel PDF $f(.)$, where $g_j = t$:

$$\int_{-\infty}^{\infty} \prod_{i \neq j} \mathbb{P}\left(g_i < t + \log(\pi_j / \pi_i)\right) f(t)\, dt . \tag{28}$$

Using the Gumbel CDF, $F(.)$, we obtain:

$$\int_{-\infty}^{\infty} \prod_{i \neq j} \exp\left(-\exp\left(-t - \log(\pi_j/\pi_i)\right)\right) f(t)\, dt , \tag{29}$$

that can be rewritten as:

$$\int_{-\infty}^{\infty} \exp\left(-\sum_{i \neq j} \frac{\pi_i}{\pi_j} \exp(-t)\right) f(t)\, dt . \tag{30}$$

Replacing the Gumbel PDF leads to the following:

$$\int_{-\infty}^{\infty} \exp\left(-\frac{\exp(-t)}{\pi_j} \sum_{i \neq j} \pi_i\right) \exp\left(-t - \exp(-t)\right) dt . \tag{31}$$

Rearranging terms gives:

$$\int_{-\infty}^{\infty} \exp\left(-t - \frac{1}{\pi_j} \sum_i \pi_i \exp(-t)\right) dt . \tag{32}$$

The following change of variable $y = -\frac{1}{\pi_j} \sum_i \pi_i \exp(-t)$ leads to the final results:

$$\mathbb{P}\left(j = \arg\max_i (g_i + \log \pi_i)\right) = \frac{\pi_j}{\sum_i \pi_i} . \tag{33}$$

The Gumbel-Max Trick allows for sampling from a categorical distribution during the forward pass through a neural network, as $z = \text{one\_hot}\left(\arg\max_i [g_i + \log \alpha_i]\right) \sim \text{Categorial}\left(\frac{\alpha_i}{\sum_j \alpha_j}\right)$ [40]. Nevertheless, since the $\arg\max$ function is not differentiable, we replace it with the Softmax function. A temperature factor λ allows us to control how closely the Gumbel-softmax distribution approximates the categorical distribution [40].

D. Polarimetric Decomposition

D.1. Coherent Polarimetric Decomposition

A common tool for studying the Sinclair matrix $\mathbf{S}$ is based on coherent decompositions, which involve of canonical objects. The underlying reason is that significant dispersion and anisotropy phenomena are expected from coherent (or artificial) targets (e.g., cars, airplanes, trucks, buildings), unlike non-deterministic targets such as vegetation (e.g., crops, forests). Therefore, these decompositions can help us to distinguish them better. Two well-known representations are usually used: the Pauli and the Krogager decomposition [32].

D.1.1. Pauli decomposition

In the context of monostatic SAR, the Pauli decomposition [10, 33] expresses $\mathbf{S}$ as:

$$\mathbf{S} = \frac{\alpha}{2}\begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix} + \frac{\beta}{2}\begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix} + \frac{\gamma}{2}\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix} , \tag{34}$$

where each term $\alpha = S_{HH} + S_{VV}$, $\beta = S_{HH} - S_{VV}$ and $\gamma = 2S_{HV}$ represent the part of the response of a plate observed at normal incidence or a sphere, the characteristic of a horizontal metallic dihedral and the scattering matrix of a metallic dihedral oriented at 45° with respect to the radar line of sight respectively. Then, we define the Pauli vector as $\mathbf{k} = \frac{1}{\sqrt{2}} (\alpha, \beta, \gamma)^T$ which is usually used to estimate the coherence matrix in Eq. (39) and to plot an RGB image by taking the module of each component.

D.1.2. Krogager Decomposition

A refined alternative approach, proposed by [26–28] considers a scattering matrix as the combination of the responses of a sphere, an oriented diplane, and a helix:

$$\mathbf{S} = e^{j\varphi}\left(e^{j\varphi_s} k_s \mathbf{S}_s + k_d \mathbf{S}_d(\vartheta) + k_h \mathbf{S}_h(\vartheta)\right) , \tag{35}$$

where $\mathbf{S}_s = \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix}$, $\mathbf{S}_d(\vartheta) = \begin{pmatrix}\cos 2\vartheta & \sin 2\vartheta\\ \sin 2\vartheta & -\cos 2\vartheta\end{pmatrix}$ and $\mathbf{S}_h(\vartheta) = e^{\pm 2j\vartheta}\begin{pmatrix}1 & \pm j\\ \pm j & -1\end{pmatrix}$, where the ± sign in the helix component varies the left or right-handedness, and it must be accounted for during the estimation of its components. The identification of the parameters is generally performed in the right-left circular basis:

$$\begin{cases} k_s = |S_{RL}| , & \\ k_h = |S_{RR}| - |S_{LL}| ,\ k_d = |S_{LL}| & \text{if } |S_{RR}| > |S_{LL}| , \\ k_h = |S_{LL}| - |S_{RR}| ,\ k_d = |S_{RR}| & \text{otherwise.} \end{cases} \tag{36}$$

The condition $|S_{RR}| > |S_{LL}|$ denotes the presence of a left-handed helix contribution. The coefficients k_s, k_d, and k_h represent the amplitude of each canonical scattering mechanism contributing to the initially measured scattering matrix $\mathbf{S}$. From the Krogager decomposition, we can deduce the vector $\mathbf{h} = \left[k_d \;\; k_h \;\; k_s\right]^T$, which is usually used to plot an RGB image.

D.1.3. Cameron Decomposition

The Cameron decomposition is a coherent target decomposition that exploits monostatic reciprocity (which forces the off-diagonal terms of the Sinclair matrix to be equal) and mirror symmetry (the existence of a symmetry axis perpendicular to the radar line of sight) to derive a compact complex descriptor of scattering [6–8]. Starting from the Pauli scattering vector defined in Section D.1.1, the antisymmetric component vanishes under reciprocity, and one represents the normalized complex parameter as:

$$z \;=\; \frac{k_2 + j\,k_3}{k_1} \tag{37}$$

In the complex z–plane, eleven prototype values z_p are assigned to various scattering responses. Each resolution cell is then classified by:

$$\hat{p} \;=\; \arg\min_{p} |z - z_p| \tag{38}$$

As such, both the amplitude and the dominant coherent mechanism are estimated per pixel. A classification procedure can be defined based on the z–plane.

D.2. Non-Coherent Polarimetric Decomposition

Although coherent decompositions help describe artificial targets, they are not suited to analyze random scattering effects (forests, fields, vegetation, etc.). To fill this lack, the Pauli vector $\mathbf{k}$ is usually modeled by the multivariate, centered, circular complex Gaussian distribution $\mathcal{CN}(\mathbf{0}, \mathbf{T})$ which is fully characterized by the covariance matrix $\mathbf{T} = E\left[\mathbf{k}\,\mathbf{k}^H\right]$. This covariance matrix is estimated locally on a PolSAR image through the Sample Covariance Matrix (SCM) $\hat{\mathbf{T}}$ computed on a set of N samples taken in a spatial boxcar:

$$\hat{\mathbf{T}} = \frac{1}{N}\sum_{i=1}^{N} \mathbf{k}_i\,\mathbf{k}_i^H\,. \tag{39}$$

The entropy H indicates the randomness of the overall backscattering phenomenon: $H = \sum_{i=1}^{3} p_i \log p_i$ with the pseudo-probabilities $p_i = \lambda_i / \left(\sum_{j=1}^{3} \lambda_j\right)$ where λ_i are the eigenvalues of the SCM $\hat{\mathbf{T}}$. Entropy H varies between 0 and 1. A low entropy indicates that the observed target is pure and the backscattering is deterministic. This is reflected by a single non-zero normalized eigenvalue close to 1. When the entropy is high, it reflects the completely random nature of the observed target. This occurs when the pseudo-probabilities are identical. The angle α is defined as $\alpha = \sum_{i=1}^{3} p_i\,\alpha_i$ where $\alpha_i = \arccos(|e_{i1}|)$ and where e_{i1} is the first component of the i-th eigenvector of the SCM. It varies between 0° and 90° and characterizes the type of the dominant scattering mechanism (surface diffusion, dihedral diffusion). The relationship between entropy, α angle, and scattering mechanisms is represented in Figure 2. A classification procedure can be defined based on the entropy and alpha parameters. Indeed, by considering the two-dimensional $H-\alpha$ space, all random scattering mechanisms can be represented [16]. Therefore, a pixel belonging to a region of the $H-\alpha$ plane allows a physical interpretation of the average scattering mechanism.

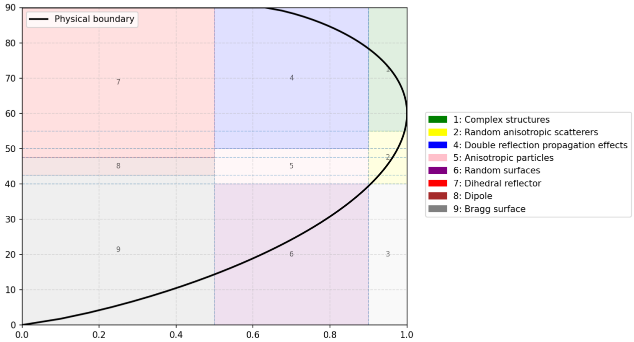

Figure 2. $H-\alpha$ plane separated into areas 1 to 9, each corresponding to a specific scattering mechanism, with the entropy at the x-axis and the scattering angle at the y-axis. The black line represents the boundary of physically possible $H-\alpha$ couples.

E. Datasets

E.1. San Francisco Polarimetric SAR ALOS-2

The San Francisco Polarimetric SAR ALOS-2 dataset[3] is an open-source PolSAR dataset. More precisely, it is an L-band polarimetric SAR image with a ground range resolution approximating 10m that has been acquired by the satellite ALOS-2: due to the penetration capability of the L-band wave into forest, vegetation, snow, and soil medium, ALOS-2 brought precious information on the earth surface objects [54]. Note that the San Francisco Polarimetric SAR ALOS-2 dataset is a *full polarimetric SAR* data, i.e., the four channels of the image represent the four elements of the Sinclair matrix. Due to the monostatic SAR context, we transform it into a three-channel image. The dataset is built by cropping the $22,608 \times 8080$ image into 64×64 non-overlapping tiles, resulting in a dataset of $44,478$ three-

[3] https://ietr-lab.univ-rennes1.fr/polsarpro-bio/san-francisco/

channel tiles. Tiles are randomly assigned to the training, validation, and test folds with 70% for training, 15% for validation, and 15% for test. Note that the image in the experiments is a crop of the original image to reduce computation cost, resulting in a 4200×2000 image.

E.2. PolSF

The PolSF dataset is a segmentation dataset segmentation mask [37]. The segmentation mask comprises 6 classes (plus one unlabeled class) corresponding to various terrain types (e.g., water, forests, urban areas). The dataset is strongly imbalanced as some classes are more represented than others. Similar to the San Francisco Polarimetric SAR ALOS-2 dataset, the PolSF dataset is constructed by cropping the 4200×2000 image into 64×64 non-overlapping tiles, resulting in a dataset of 3397 three-channel tiles. Tiles are randomly assigned with a sampling weight based on their majority class to the training, validation, and test folds, with 70% allocated to training, 15% to validation, and 15% to test.

E.3. S1SLC_CVDL

The S1SLC_CVDL[4] is an open-source PolSAR dataset [42]. It is a C-band polarimetric SAR image with a ground range resolution [19]. This dataset comprises $276,571$ two-channel images semantically annotated in 7 different classes. Note that the S1SLC˙CVDL dataset is a *dual-polarimetric SAR* dataset, i.e., the two channels of the images represent the S_{HH} and S_{HV} elements of the Sinclair matrix. Like the PolSF dataset, tiles are randomly assigned with a sampling weight based on their majority class to the training, validation, and test folds with 70% allocated to training, 15% to validation, and 15% to test.

Model	Trainable Parameters	Inference Time (ms)
ResNet LPS MLP	$2,447,560$	48.82 ± 0.08
ResNet LPS PolyDec	$2,446,100$	38.56 ± 0.17
ResNet LPS Norm	$2,445,995$	18.70 ± 0.07
ResNet APS	$1,226,467$	7.56 ± 0.30

Table 4. Trainable parameters and inference time for shift-equivariant CVNN variants on the S1SLC_CVDL dataset. Inference time is reported on 50 iterations. CVNN Learnable Polyphase Sampling (LPS) variants are more costly than the shift-equivariant Adaptive Polyphase Sampling (APS) model, both in number of trainable parameters and in inference time.

F. Experimental setup

F.1. Classification

For the training of the ResNet, we have used the AdamW optimizer [38] with a weight decay of 10^{-5} and a learning rate varying between 10^{-2} and 10^{-3} depending on the projection function. Indeed, memory-intensive projection functions like $\mathrm{PolyDec}$ and MLP necessitated reducing the batch size and, by extension, the learning rate. The model is trained for 100 epochs. For both RVNNs and CVNNs, we set the number of layers in the encoder to 4, with an initial number of channels set to 16. To address the class imbalance issue, we have implemented a complex-valued version of Focal Loss [36] for CVNNs and utilized the real-valued version of Focal Loss for RVNNs. Finally, we have used the ReLU activation for RVNNs, and $\mathrm{modReLU}$ for CVNNs. For the shift-equivariant models relying on the Gumbel Softmax, we have set the initial temperature value 10^{-5}. Table 4 shows that the shift-equivariant CVNN LPS variants have a similar amount of trainable parameters, but differ greatly during inference time. We see a clear correlation between the number of trainable parameters and the inference time: as such, the Norm variant is the least costly in time and memory, while the MLP variant is the most demanding one. Interestingly, we notice that the projection layer does not have a great impact on the number of trainable parameters (switching from LPS to APS has a far greater impact on this aspect).

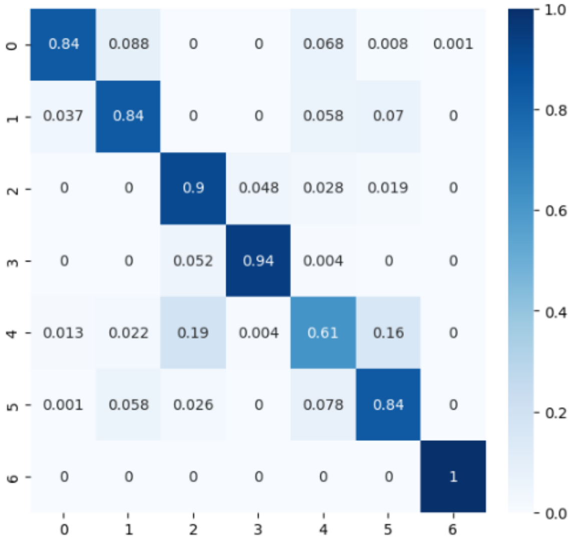

Figure 3. Confusion matrix of a CVNN LPS MLP for classification on S1SLC˙CVDL

F.2. Semantic segmentation

For the training of the UNet, we have chosen the AdamW optimizer [38] with a weight decay of 5×10^{-4} and a learning rate of 10^{-3}. The model is trained for 500 epochs. For both RVNNs and CVNNs, we set the number of layers in the encoder and the decoder to 4, with an initial number of channels set to 16. The rest of the setup is similar to the procedure presented in Appendix F.1. Table 5 shows that the

[4] https://ieee-dataport.org/open-access/s1slccvdl-complex-valued-annotated-single-look-complex-sentinel-1-sar-dataset-complex

Model	Trainable Parameters	Inference Time (ms)
UNet LPS MLP	$3,297,660$	29.20 ± 0.21
UNet LPS PolyDec	$3,296,200$	23.05 ± 0.07
UNet LPS Norm	$3,296,095$	14.10 ± 0.27
UNet APS	$2,075,588$	11.75 ± 0.40

Table 5. Trainable parameters and inference time for shift-equivariant CVNN variants on the PolSF dataset. Inference time is reported on 50 iterations. CVNN Learnable Polyphase Sampling (LPS) variants are more costly than the shift-equivariant Adaptive Polyphase Sampling (APS) model, both in number of trainable parameters and in inference time.

shift-equivariant CVNN LPS variants have a similar amount of trainable parameters, but differ greatly during inference time. We see a clear correlation between the number of trainable parameters and the inference time: as such, the Norm variant is the least costly in time and memory, while the MLP variant is the most demanding one. Interestingly, we notice that the projection layer does not have a great impact on the number of trainable parameters (switching from LPS to APS has a far greater impact on this aspect).

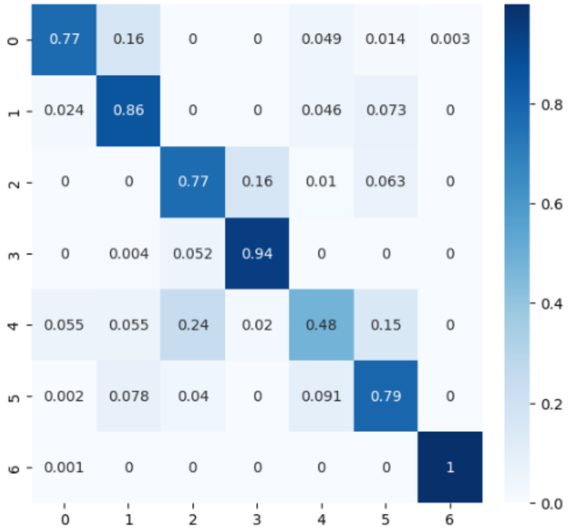

Figure 4. Confusion matrix of a CVNN LPF for classification on S1SLC˙CVDL

F.3. Reconstruction

For the training of the AutoEncoder, we have chosen the AdamW optimizer [38] with a weight decay of 0 and a learning rate of 5×10^{-4}. The model is trained for 50 epochs. We set the number of layers in the encoder and the decoder to 2, with an initial number of channels to 64. As expected from the unsupervised nature of the reconstruction task, we have used the Mean Squared Error loss function. Finally, we have used the ReLU activation for RVNNs, and $\mathrm{modReLU}$ for CVNNs. For the shift-equivariant models relying on the Gumbel Softmax, we have set the initial temperature value to 0.001, the gamma value to 0.1, and the minimum value to 10^{-5}. Table 6 shows that the shift-equivariant CVNN LPS variants have a similar amount of trainable parameters, but differ greatly during inference time. We see a clear correlation between the number of trainable parameters and the inference time: as such, the Norm variant is the least costly in time and memory, while the MLP variant is the most demanding one. Interestingly, we notice that the projection layer does not have a great impact on the number of trainable parameters (switching from LPS to APS has a far greater impact on this aspect).

Model	Trainable Parameters	Inference Time (ms)
AE LPS MLP	$2,898,626$	88.98 ± 0.03
AE LPS PolyDec	$2,898,042$	75.29 ± 0.05
AE LPS Norm	$2,898,000$	46.50 ± 0.02
AE APS	$1,750,348$	24.23 ± 0.01

Table 6. Trainable parameters and inference time for shift-equivariant CVNN variants on the San Francisco Polarimetric SAR ALOS-2 dataset. Inference time is reported on 50 iterations. CVNN Learnable Polyphase Sampling (LPS) variants are more costly than the shift-equivariant Adaptive Polyphase Sampling (APS) model, both in number of trainable parameters and in inference time.

G. Additional Results

G.1. Classification

In addition to the results presented in Section 4.2, we include the confusion matrices of the CVNN LPS MLP and the CVNN LPF to showcase the impact of our method against a non-shift-equivariant CVNN. As we can observe from Figures 3 and 4, the confusion matrix of the CVNN LPS MLP is better. The semantic classes of the S1SLC˙CVDL dataset are defined as follows: Agricultural fields, Forest and Woodlands, High-Density Urban Areas, High Rise Buildings, Low-Density Urban Areas, Industrial Regions, and Water Regions. The confusion matrix from Figure 3 shows a better distinction between urban areas than the confusion matrix from Figure 4: High-Density Urban Areas, Low-Density Urban Areas, and Industrial Regions. This result is crucial as it highlights that our proposed approach increases the viability of neural networks for real-life applications.

G.2. Semantic Segmentation

In addition to the results presented in Section 4.3, we include the visualizations of the CVNN LPS $\mathrm{PolyDec}$ and the CVNN LPF to showcase the impact of our method against a non-shift-equivariant CVNN. As we can observe from Figures 5 and 6, the predicted segmentation mask of the CVNN LPS $\mathrm{PolyDec}$ is smoother and closer to the ground truth.

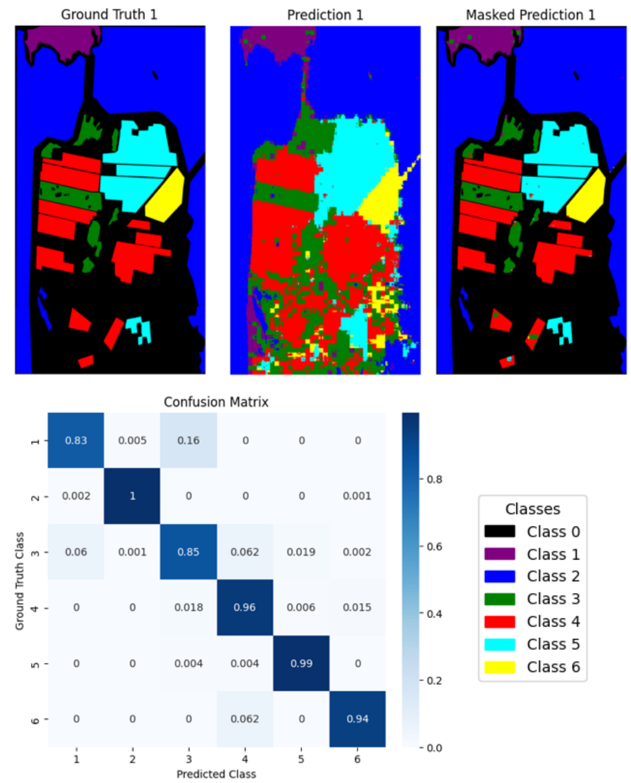

Figure 5. Results obtained with a CVNN LPS PolyDec for semantic segmentation on PolSF. Left: ground truth segmentation mask. Middle: the complete prediction (without a mask for the unlabeled class). Right: prediction (with a mask for the unlabeled class). Bottom: confusion matrix between ground truth and prediction

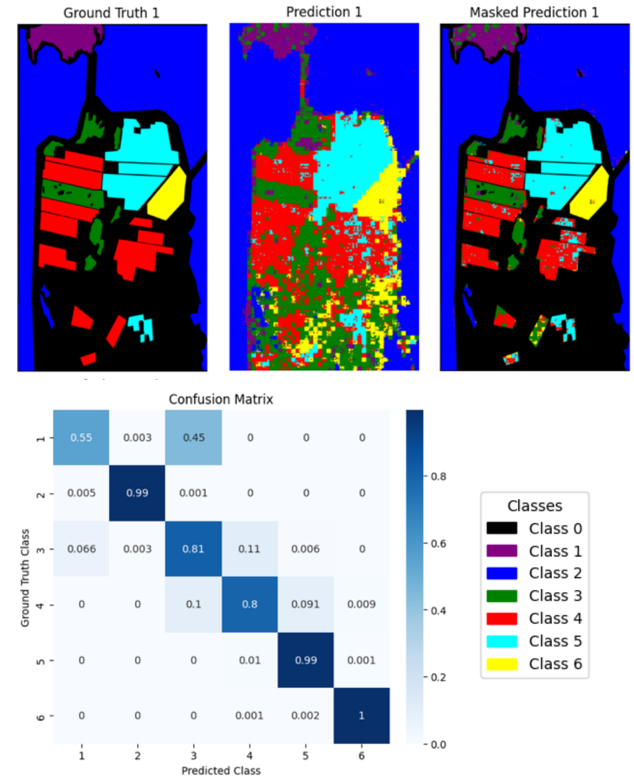

Figure 6. Results obtained with a CVNN LPF for semantic segmentation on PolSF. Left: ground truth segmentation mask. Middle: complete prediction (without a mask for the unlabeled class). Right: prediction (with a mask for the unlabeled class). Bottom: confusion matrix between ground truth and prediction

The semantic classes of the PolSF dataset are defined as follows: Mountain, Water, Vegetation, High-Density Urban, Low-Density Urban, and Developed Area. The confusion matrix from Figure 5 shows a better distinction between natural areas than from Figure 6: Mountain and Vegetation. This conclusion is further supported by latent spaces visualization from Figures 7 and 8. We also have fewer artifacts in the middle of correctly labeled zones (such as wrongfully predicting the presence of a forest in the middle of an urban area). As stated in G.1, this result highlights the advantages of our approach over traditional architectures regarding real-life applications.

G.3. Reconstruction

In addition to the results presented in Section 4.4, we include the visualizations of the CVNN LPS PolyDec and the CVNN LPF to showcase the impact of our method against a non-shift-equivariant CVNN. As we can observe from Figures 9 and 10, the reconstruction of the CVNN LPS PolyDec is almost perfect when compared to the ground truth.

The semantic classes of the $H - \alpha$ are shown in Figure 2 and defined as follows: Complex structures, Random anisotropic scatterers, Double reflection propagation effects, Anisotropic particles, Random surface, Dihedral reflector, Dipole, and Bragg surface. Similarly, the semantic classes of the Cameron are defined as follows: Non-reciprocal, Asymmetric, Left helix, Right helix, Symmetric, Trihedral, Dihedral, Dipole, Cylinder, Narrow dihedral, and Quarter-wave.

The various visualizations allow us to make a general observation: polarimetric decompositions (Pauli, Krogager, and $H - \alpha$) and reconstruction metrics (amplitude and angular distances) are incredibly better for the CVNN LPS PolyDec model 9 when compared to the CVNN LPF model 10. We believe that such results show promising perspectives regarding further experiments on the impact of CVNNs on the reconstruction of PolSAR images.

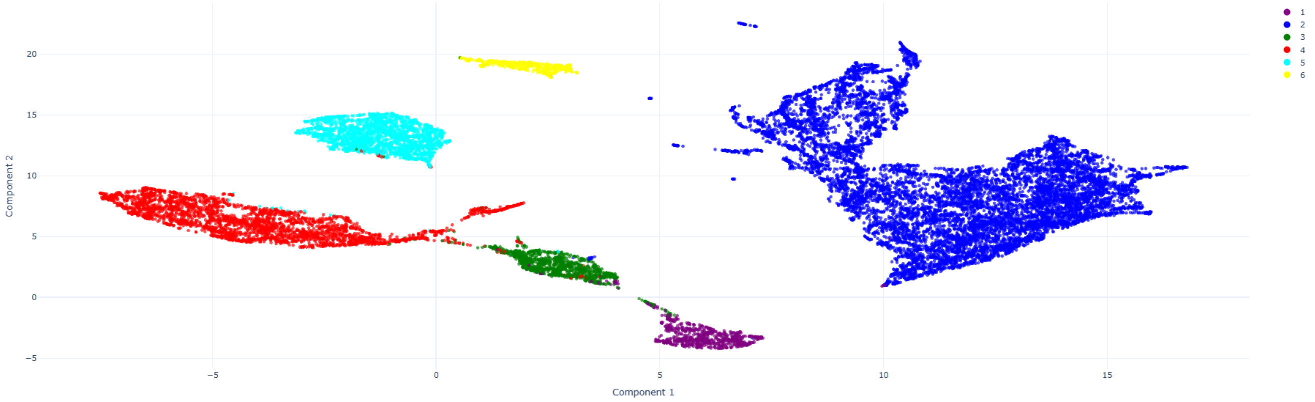

Figure 7. UMAP visualization of the latent space from CVNN LPS PolyDec on PolSF dataset.

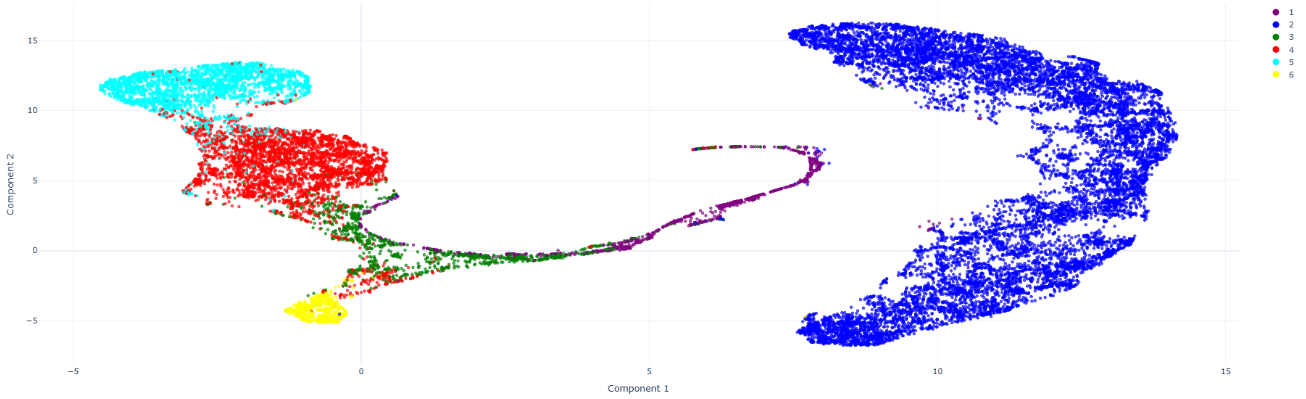

Figure 8. UMAP visualization of the latent space from CVNN LPF on PolSF dataset.

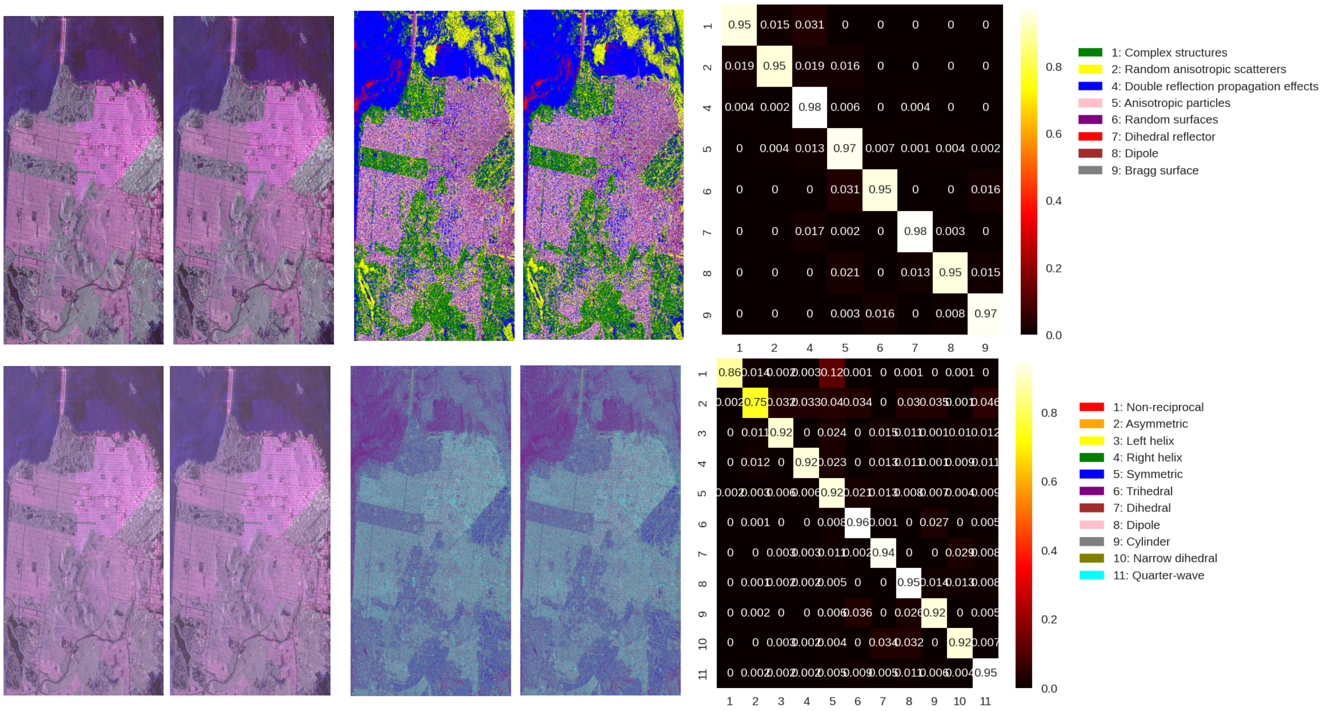

Figure 9. Reconstruction results obtained from CVNN LPS PolyDec on San Francisco Polarimetric SAR ALOS-2 dataset. Up-Left: Amplitude images of the original (left) and reconstructed (right) images with the Pauli basis. Up-Middle: Images of the original (left) and reconstructed (right) images with the $H-\alpha$ classification. Up-Right: Down-Right: Confusion matrix of the original (rows) and reconstructed (columns) $H-\alpha$ classes. Down-Left: Amplitude images of the original (left) and reconstructed (right) images with the Krogager basis. Down-Middle: Images of the original (left) and reconstructed (right) images with the Cameron classification. Down-Right: Confusion matrix of the original (rows) and reconstructed (columns) Cameron classes.

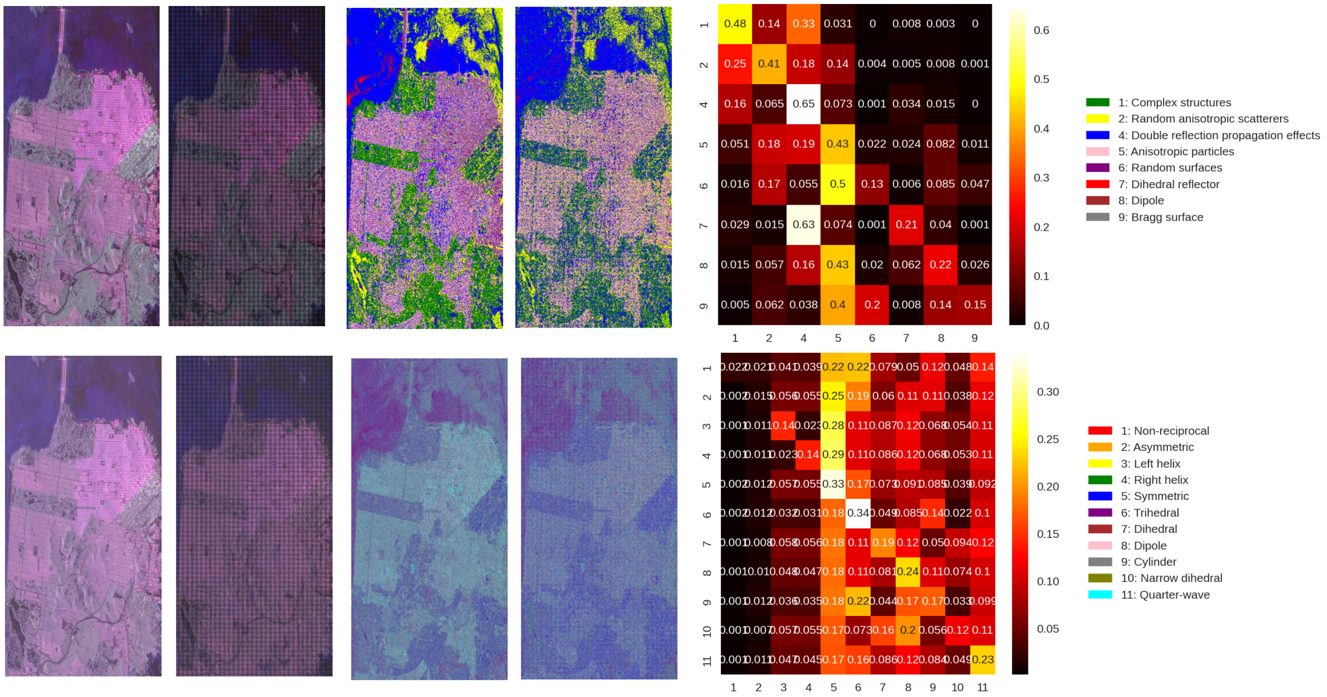

Figure 10. Reconstruction results obtained from CVNN LPF on San Francisco Polarimetric SAR ALOS-2 dataset. Up-Left: Amplitude images of the original (left) and reconstructed (right) images with the Pauli basis. Up-Middle: Images of the original (left) and reconstructed (right) images with the $H - \alpha$ classification. Up-Right: Down-Right: Confusion matrix of the original (rows) and reconstructed (columns) $H - \alpha$ classes. Down-Left: Amplitude images of the original (left) and reconstructed (right) images with the Krogager basis. Down-Middle: Images of the original (left) and reconstructed (right) images with the Cameron classification. Down-Right: Confusion matrix of the original (rows) and reconstructed (columns) Cameron classes.