LEARNABLE RIESZ TRANSFORM FOR COMPOSITE SCALE-ROTATION EQUIVARIANT SPATIAL TRANSFORMERS

Anonymous authors
Paper under double-blind review

ABSTRACT

Building models robust to transformations such as rotation, scale, and translation is a challenge in machine learning and computer vision. Existing approaches often provide only partial and discrete equivariance (group equivariance) or rely on supervision or very abundant data to learn equivariant representations. To achieve fine-grained equivariance from low data, we combine and improve over both approaches. We propose a novel, learnable, Riesz-transform-based architecture that achieves built-in group equivariance for translation, rotation, and scale. We combine it with a Spatial Transform Network (STN) tailored for the sequential estimation of composite transformations, reducing the combinatorial data requirements for learning fine-grained equivariance. Improved generalization guarantees and extensive experiments demonstrate that our approach brings improvements over state-of-the-art methods in unsupervised representation learning and object discovery, even more so in low-data regimes.

1 INTRODUCTION

Humans excel at recognizing objects from very limited examples, demonstrating robust generalization across arbitrary positions, orientations, and scales. Achieving similar robustness in machine learning models remains a significant challenge, motivating the development of approaches and architectures that inherently respect geometric symmetries, starting with convolutional neural networks (CNNs, e.g. O'shea & Nash (2015)). Handling symmetries may require either invariance and equivariance. An image classification task typically requires invariance: if an image is rotated, its label (e.g. "cat") remains the same. In contrast, tasks like object detection and semantic segmentation require equivariance: if an image is rotated, the predicted bounding boxes or masks should rotate accordingly. Typically, a CNN has translation equivariance by design, and when combined with pooling, it yields translation invariance.

Equivariance is often necessary for tasks requiring precise spatial understanding, such as object detection and segmentation, or latent representation learning. Rotation and translation equivariant architectures have been developed to address these needs (Cohen & Welling, 2016; Zitnick et al., 2022), but rarely address scale and are often limited to discrete groups. For rotation equivariance, this may mean handling only (multiple of) 90-degree rotations. For translation, it may mean equivariance only to shifts by multiples of the stride (or pooling size), in a deep CNN.

An alternative solution to build equivariant models is to learn to be equivariant (or invariant) using data, which occurs when the model is exposed to a diverse set of transformations during training. This is typically done using approaches like Spatial Transformer Networks (STN, Jaderberg et al. (2015)) that learn to estimate a transformation that brings the input into a canonical form. These work well when they can leverage a supervision signal and use data augmentation. However, this approach is limited by the range of transformations seen during training and may not generalize well to unseen transformations. Also, in some cases, supervision might be unavailable and adequate data augmentation may not be possible. This is the often the case in unsupervised learning and especially in unsupervised object discovery. This task involves identifying objects (or recurring patterns, as no supervision is available) in an image collection, where each image may contain multiple objects, each subject to its own transformation.

In this work, we focus on reducing the data requirements for learning equivariant models, especially targeting the case of unsupervised object discovery. Starting from the realization that achieving fine-grained equivariance requires, in fine, both architectural choices (for built-in discrete equivariance) and learning from data (for refining the equivariance), we propose improvements in both aspects, jointly. Our main contributions are as follows:

- Leveraging the fact that learning/tuning is necessary, we propose an equivariant architecture that embeds a part of learning. Named **Lea**rnable **R**iesz-transform **N**etwork (LeaRN), it provides equivariance to scale in addition to rotation.
- Realizing that the transformations estimated by STNs are composite in nature (e.g., translation followed by scaling and rotation), we propose a sequential estimation of these transformations thus reducing the combinatorial need for training examples.

In addition, we (i) provide theoretical guarantees on the generalization capabilities of our architecture through a tighter PAC-Bayesian bound. (ii) We demonstrate the interest of our approach through extensive experiments and ablation studies on autoencoding and unsupervised object discovery, showing significant improvements over state-of-the-art methods, especially in low-data regimes. (iii) We further introduce a metric to quantitatively assess the consistency and accuracy of rotational angle predictions in unsupervised settings (where there is no ground truth angle).

2 Related Work

CNNs introduced translation equivariance through weight sharing, significantly reducing parameters compared to fully connected architectures (LeCun et al., 1998). This principle of encoding inductive biases aligned with data symmetries has become foundational to equivariant learning. Spatial Transformer Networks extended this by learning spatial transformations directly from data (Jaderberg et al., 2015), though they face limitations in unsupervised scenarios. Cohen and Welling (Cohen & Welling, 2016) generalized CNNs to arbitrary groups through G-CNNs, achieving discrete rotation equivariance via lifting convolutions. This approach has proven effective in fluid dynamics simulation and biological applications (Bekkers et al., 2018; Andrearczyk et al., 2019)). Lafarge et al. (2021) refined G-CNNs by preserving directional information across layers for hierarchical structure detection. Steerable CNNs further extended these ideas to continuous groups using symmetry-constrained kernels on feature vector fields (Weiler et al., 2018; Weiler & Cesa, 2019; Cesa et al., 2022)). Alternative approaches include spherical networks that leverage sparse Clebsch-Gordan matrices for computational efficiency (Zitnick et al., 2022; Passaro & Zitnick, 2023).

In unsupervised settings, several approaches have emerged to learn equivariant representations without labels. TARGET-VAE explicitly encodes pose and orientation through equivariant architectures (Nasiri & Bepler, 2022) but encounters difficulties with discrete rotations requiring interpolation. IRL-INR addresses rotation equivariance via specialized latent modules with contrastive losses (Kwon et al., 2023), though generalization to unseen rotations remains limited due to data dependency. Kaba et al. (2023) propose two complementary frameworks: an "equivariant-by-design" approach that generalizes TARGET-VAE to unsupervised translation and rotation tasks, and an "optimization-based" method formulating equivariance as contrastive moment alignment. However, the latter primarily succeeds in supervised downstream applications. CODAE extends optimization-based equivariance by enforcing structured latent representations through group moment matching (Cha et al., 2025), though early-stage initialization with random noise samples poses significant challenges. Roto-Scale Equivariance. Despite its practical importance, joint rotation and scale equivariance has received limited attention. Among the notable work

Recent unsupervised equivariant learning has focused primarily on architectural innovations, yet classical image processing offers underexplored complementary insights. The Riesz transform demonstrates natural equivariant properties and effectively captures edge structures (Joyseeree et al., 2019; Depeursinge et al., 2011), with steerable Riesz wavelets enabling rotation-sensitive feature extraction (Depeursinge et al., 2013). However, these methods have been applied mainly to supervised downstream tasks (Barisin et al., 2024b;a), leaving their potential for upstream representation learning untapped. We leverage the Riesz transform's inherent equivariant properties for unsupervised representation learning. One of the key contributions of our paper is to extend the idea of unsupervised equivariant feature representation to object discovery and STNs has been a key component for

benchmark object discovery models like SPACE (Lin et al., 2020), SPAIR (Crawford & Pineau, 2019), GNM (Jiang & Ahn, 2020), and GMAIR (Zhu et al., 2022), which motivates us to build an "equivariant by design" network that can be integrated to such STN based models.

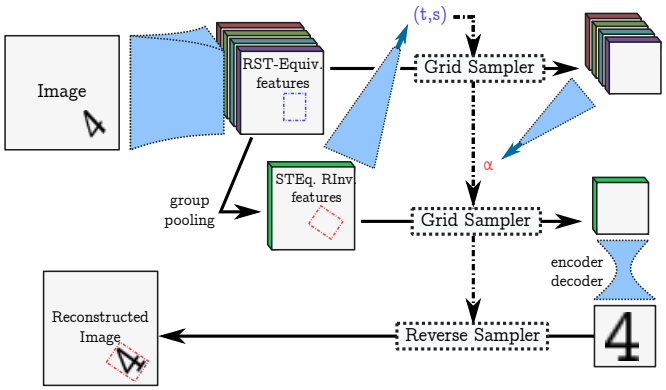

Figure 1: Proposed LeaRN-EqSTN architecture for unsupervised representation learning. Efficient end-to-end learning (learned blocks in dotted, blue) is achieved by leveraging Riesz-enhanced rotation-equivariant features and by disentangling pose estimation via a two-step estimation. This reduces the combinatorial need for training example to produce finely aligned crops for appearance learning.

3 Background

The main components of the proposed equivariant architecture illustrated in Fig. 1 are hereafter described. Detailed explanations of their roles are provided in the Appendix B. Let us remind here that a function f is said equivariant to a transformation group G, if $f(g \cdot x) = \rho(g) \cdot f(x)$ for all $g \in G$ and input x, where $\rho(g)$ is a corresponding transformation in the output space.

3.1 Spatial Transformer Networks

Spatial Transformer Networks (STNs) are designed to enhance neural networks by introducing learnable spatial transformations that improve robustness to geometric variations in input data, such as rotations, translations, scaling, and shearing, thereby enabling better generalization in tasks like image classification and object detection. They apply a parameterized transformation $T_\theta \in \mathbb{R}^{2\times 3}$, typically an affine matrix, mapping input coordinates (h_s, w_s) to output coordinates (h_t, w_t). Input feature maps $U \in \mathbb{R}^{H\times W\times C}$ are transformed into output feature maps $V \in \mathbb{R}^{H'\times W'\times C}$ using bilinear interpolation. A localization network predicts θ via backpropagation, optimizing the task-specific loss $L_{\text{task}}(V, y)$, enabling dynamic feature alignment and better generalization.

STN Limitations: STNs struggle to generalize beyond the training distribution (Oliver et al., 2018). Unlike architectures with built-in equivariance, they must explicitly learn each transformation from data. This becomes problematic in complex scenarios involving multiple objects or composite transformations, where the exponential growth of transformation space makes comprehensive training impractical, highlighting their dependence on diverse training data.

3.2 Group Convolutions and Equivariant Feature Learning

Whereas standard convolutions handle by design shifts well but struggle with rotations and scales, group convolutional layers address these issues by embedding symmetry directly into the network architecture. More specifically, they use the symmetry group P_r, which combines translations in $\mathbb{Z}^2$ with r discrete rotations by $\alpha_k = k \cdot \frac{2\pi}{r}$, $k \in \{0, 1, \ldots, r-1\}$. Unlike STNs, group convolutions achieve rotation equivariance for an input $\mathcal{I}$ through the operation $(\mathcal{I} \star \psi)(g) = \sum_{x\in\mathbb{Z}^2} \mathcal{I}(x)\psi(g^{-1}x)$, where $g \in P_r$. This applies kernel ψ at r discrete orientations, producing r feature maps that transform predictably under input rotations. Steerable CNNs extend this to continuous rotations under

$SO(2)$, using steerable filters $\Psi(x) = \sum_{m=1}^{M} c_m \psi_m(x)$, where χ_m are basis functions (e.g., circular harmonics). These filters adapt to any rotation α via $\psi'(x) = \psi(\rho_\alpha^{-1} x)$, efficiently representing rotations without discretizing angles. This makes steerable CNNs more flexible and computationally efficient than group CNNs for continuous symmetries, ideal for tasks where orientation is arbitrary. For feature learning we work with a rotation equivariant variant of ResNet (ReResNet, Han et al. (2021)), here group convolutions operate on a cyclic group G_N, representing N-fold discrete rotations (e.g., $\{0^\circ, \frac{360^\circ}{N}, \ldots, \frac{360^\circ(N-1)}{N}\}$). These convolutions produce feature maps with orientation channels for each group element $g \in G_N$, satisfying rotation equivariance: $\mathcal{A}(g, x) = f_{\text{ReResNet}}(\mathcal{I}_{\text{aug}}(x), g)$, where rotating the input by $g' \in G_N$ yields $\mathcal{A}(g, x) = \mathcal{A}(g'g, x)$. To achieve rotation-invariant features, group-invariant global pooling is applied: $\mathcal{A}_{\text{invariant}}(x) = \frac{1}{|G_N|} \sum_{g \in G_N} \mathcal{A}(g, x)$. This aggregates features across all rotations, ensuring consistent representations regardless of orientation.

3.3 Riesz Transform

Given a real-valued input image $I \in L^2(\mathbb{R}^2)$, the first-order Riesz transform is defined via Fourier transform $\mathcal{F}$ as $\mathcal{R}_j I(x) = \mathcal{F}^{-1}\left[-i\frac{\xi_j}{|\xi|} \cdot \mathcal{F}I\right], \quad j \in \{1, 2\}$, where $\xi = (\xi_1, \xi_2) \in \mathbb{R}^2$ are frequency coordinates and $|\xi| = \sqrt{\xi_1^2 + \xi_2^2}$. The transform produces a vector field $\mathcal{R}I = (\mathcal{R}_1 I, \mathcal{R}_2 I)$, whose components correspond to frequency-normalized directional derivatives. It is **rotation-equivariant** ($\mathcal{R}[I(\rho_\alpha x)] = \rho_\alpha \mathcal{R}I(x)$, **scale-equivariant** ($\mathcal{R}[I(sx)](x) = \mathcal{R}I(sx)$, $s > 0$), and **translation-equivariant** ($\mathcal{R}[I(x - t)](x) = \mathcal{R}I(x - t)$, $t \in \mathbb{R}^2$), for all $\rho_\alpha \in \mathsf{SO}(2)$, where R_α denotes a 2D rotation matrix of angle α. This makes it a natural primitive for constructing equivariant representations. For any $\alpha \in [0, 2\pi)$, the Riesz transform admits **steerable synthesis**: $\mathcal{R}_\alpha I(x) := \mathbf{u}_\alpha^\top \mathcal{R}I(x), \quad$ with $\mathbf{u}_\alpha = (\cos\alpha, \sin\alpha)^\top$. To capture second-order geometric structures such as curvature, we further define frequency-domain operators: $\mathcal{R}_{xx} I(x) = \mathcal{F}^{-1}\left[-\frac{\xi_1^2}{|\xi|^2}\hat{I}(\xi)\right]$, $\mathcal{R}_{yy} I(x) = \mathcal{F}^{-1}\left[-\frac{\xi_2^2}{|\xi|^2}\hat{I}(\xi)\right]$, $\mathcal{R}_{xy} I(x) = \mathcal{F}^{-1}\left[-\frac{\xi_1\xi_2}{|\xi|^2}\hat{I}(\xi)\right]$ where $\hat{I}$ is the Fourier transform of image I. These can be interpreted as anisotropic second-order filters aligned to principal directions of local image structure. Together, the first and second-order Riesz components form a rotation-equivariant feature algebra under the natural group action. The Riesz transform is also an **isometry** in $L^2(\mathbb{R}^2)$, meaning it preserves the L^2-norm of the input image, $\|\mathcal{R}_j I\|_{L^2} = \|I\|_{L^2}, \quad j = 1, 2$. This property ensures that the transform does not amplify or diminish the energy of the image, making it stable for feature extraction in neural networks. In the Fourier domain, the Riesz transform corresponds to a projection onto the $m = \pm 1$ modes of the $SO(2)$ Fourier (see Appendix B for details).

3.4 Aliasing in convolutional architectures

In CNNs, downsampling such as by a factor of 2 via stride-2 convolutions, reduces an $M \times M$ feature map to $M/2 \times M/2$, causing aliasing as frequencies beyond the new Nyquist bound ($M/4$) fold into lower frequenciesGruver et al. (2022); Edixhoven et al. (2023).The discrete Fourier transform of the downsampled image introduces distortions disrupting translation equivariance, as translations yield incorrect phase shifts. In group-equivariant architectures like ReResNet, which ensure rotational equivariance for the cyclic group G_N, feature maps $\mathcal{A}(g, x) = f_{\text{ReResNet}}(\mathcal{I}(x), g)$ should satisfy $\mathcal{A}(g, x) = \mathcal{A}(g'g, x)$ under rotation by $g' \in G_N$. Aliasing distorts these maps, as high-frequency terms introduce inconsistent transformations, breaking rotational equivariance. Non-linearities like ReLU exacerbate this by generating higher harmonics (see Appendix C).

4 Contributions

We start by introducing our learnable Riesz-transform Network, that provides a broad built-in group equivariance, together with its theoretical properties. We then describe how a cascade of STNs can reduce the combinatorial data requirements for learning fine-grained equivariance. Finally, we present our full LeaRN-EqSTN architecture, combining these components for unsupervised representation learning and object discovery.

4.1 LEARNABLE STEERABLE RIESZ AND GENERALIZATION

To construct a feature extraction mechanism that is equivariant to scale and rotations, we improve the Riesz with a **Lea**rnable steerable **R**iesz transform **N**etwotk (**LeaRN**) that combines the input image, first-order, and second-order Riesz transforms, modulated by learnable spectral scaling factors in the frequency domain. Specifically, let $s_\ell : \mathbb{R}^2 \to \mathbb{R}^+$, for $\ell = 0, 1, \ldots, L$, be radially symmetric functions (i.e., $s_\ell(\xi) = s_\ell(|\xi|)$) expanded using a Laplacian basis to efficiently capture peaked frequency responses thanks to heavy-tailed Laplacian distributions: $s_\ell(|\xi|) = \sum_{m=1}^{M} w_{\ell m} \exp\left(-b_m|\xi - \xi_m|\right)$ where $w_{\ell m} \in \mathbb{R}^+$ are learnable weights, $b_m > 0$ are fixed scale parameters, and $\xi_m \in \mathbb{R}^+$ are fixed frequency centers, ensuring radial symmetry since the basis depends only on $|\xi|$. We introduce here a feature transform, denoted as $\mathcal{T} : L^2(\mathbb{R}^2) \to L^2(\mathbb{R}^2)$, where features $\mathcal{A}$ are transformed into features $\mathcal{T}[\mathcal{A}]$ in the Fourier domain as follows: $\mathcal{T}[\mathcal{A}] = \left[s_0(|\xi|)\mathcal{F}(\mathcal{A}), \left\{s_k(|\xi|)\mathcal{F}(\mathcal{R}_1(\mathcal{R}_{\alpha_k}\mathcal{A}))\right\}_k, \left\{s_j(|\xi|)\mathcal{F}(\mathcal{R}_2(\mathcal{R}_{\alpha_j}\mathcal{A}))\right\}_j, \left\{s_{jk}(|\xi|)\mathcal{F}(\mathcal{R}_{jk}\mathcal{A})\right\}_{j,k}\right]$, where, $\mathcal{R}_j$, $j = 1, 2$, are first-order Riesz transforms with , $\mathcal{R}_{jk}f = \mathcal{R}_j(\mathcal{R}_k f)$, $j, k \in \{x, y\}$, are second-order Riesz transforms, $\mathcal{R}_{\alpha_k}, \mathcal{R}_{\alpha_j}$ denote rotations by angles $\alpha_k, \alpha_j \in [0, 2\pi)$, with N, M the number of discretized angles, $s_\ell(|\xi|)$, $\ell = 0, 1, \ldots, N + M + 3$, are radially symmetric scaling functions. To promote sparsity in the weights and peaked frequency responses that emphasize specific frequency bands, we introduce an L^1 regularization term over the weights: $R_1(\Theta) = \sum_{\ell,m} |w_{\ell m}|$. To ensure smoothness, we add an L^2 regularization term on the gradient of s_ℓ: $R_2(\Theta) = \sum_\ell \|\nabla s_\ell\|_2^2$.

Aliasing: The steerable Riesz transform, prevents aliasing from geometric transformations by computing equivariant features in the frequency domain at full resolution before any downsampling. Since it corresponds to structured, continuous operations in the Fourier domain, they do not introduce artificial discontinuities or high-frequency spikes before downsampling that would otherwise cause aliasing (see Appendix C.2 for proof).

Generalizability: Given an equivariant network incorporating steerable group convolutions, we propose augmenting the network with a learnable steerable Riesz transform upstream to the initial group convolution layers in the aim to build a more generalizable architecture able to well approximate equivariant feature maps ϕ (see definition 1). Theorem 2 about Riesz Equivariant feature bounds confirms generalizability on the basis of homogeneous bounds for equivariant networks:

Theorem 1. *[Homogeneous Bounds for Equivariant Networks Behboodi et al. (2022)] For any equivariant network f, with high probability we have:*

$$\mathcal{L}(f_{\mathbf{W}}) \leq \hat{\mathcal{L}}_\gamma(f_{\mathbf{W}}) + \tilde{\mathcal{O}}\left(\sqrt{\left(\frac{\prod_l \|\mathbf{W}_l\|_2^2}{\gamma^2 m \eta}\right)\left(\sum_{l=1}^{L}\sqrt{M(l,\eta)}\right)^2 \sum_l \frac{\sum_{\psi,i,j} \|\widehat{\mathbf{W}}_l(\psi,i,j)\|_F^2 / \dim_\psi}{\|\mathbf{W}_l\|_2^2}}\right)$$

where $\mathcal{L}(f_W)$ represents the true expected loss and $\hat{\mathcal{L}}_\gamma(f_W)$ is the empirical margin loss. γ denotes the classification margin parameter, while $\eta \in (0, 1)$ is a perturbation probability parameter. L is the number of network layers, with W_l representing the weight matrices at layer l. $B = \max\left(1, \prod_l \|W_l\|_2\right)$ bounds the network output, and $\beta = \prod_l \|W_l\|_2$ is the product of spectral norms. $\hat{W}_l(\psi, i, j)$ are the kernel parameters in the Fourier domain for irreducible representation ψ, with $\dim \psi$ being the dimension of that representation and $M(l, \eta) := \log\left(\frac{\sum_{l=1}^{L}\sum_\psi m_{l,\psi}}{1-\eta}\right) \max_\psi\left(5 m_{l-1,\psi} m_{l,\psi} c_\psi\right)$.

Definition 1. An ideal equivariant feature map $\phi : L^2(\mathbb{R}^2, \mathbb{R}^{c_{\text{in}}}) \to L^2(\mathrm{SO}(2), \mathbb{R}^{c_{\text{out}}})$ satisfies the following property: for any $x \in L^2(\mathbb{R}^2, \mathbb{R}^{c_{\text{in}}})$ and $h \in \mathrm{SO}(2)$, with the action $x_h(y) = x(h^{-1}y)$ for $y \in \mathbb{R}^2$, the feature map is equivariant under $\mathrm{SO}(2)$, i.e.,

$$\phi(x_h)(g) = \phi(x)(hg)$$

for all $g \in \mathrm{SO}(2)$.

Theorem 2. *[Riesz Equivariant feature bounds] For a learnable steerable Riesz Transform, $\mathcal{R}_{net}(x) \in L^2(\mathbb{R}^2)^2$, and $\phi(x)$ as defined above satisfying the assumption that the Fourier coefficients of $\phi(x)$, are significant only for $|m| \leq 1$, when lifted to the $L^2(\mathrm{SO}(2))$ space as $\tilde{x}$, $\tilde{\mathcal{R}}_{net}(x)$, the following inequality $\left\|\tilde{\mathcal{R}}_{net}(x) - \phi(x)\right\|_2 < \|\tilde{x} - \phi(x)\|_2$ holds.*

Through our proof (D) we argue that if the Riesz representation tightens the homogeneous bound it would in turn improve the model's generalization.

4.2 LEARNING COMPOSITE TRANSFORMATIONS

Geometric transformations have an impact on training data efficiency. We focus here on rigid transformations. As illustrated in Figure 2, we consider three approaches to handling transformations:

Built-in equivariance: Implementing the downstream task F with inherent equivariance ($F(\rho_\alpha x) = \rho'_\alpha F(x)$) enables generalization from minimal examples. However, practical implementations often use discrete rotation groups (e.g., P_4), requiring exposure to samples covering the quotient space ($\alpha \in [0, \frac{\pi}{2}]$). This is illustrated in Fig. 2(a-b), where ConvNets and ReResNets provide only discrete equivariance, necessitating training across variations within each equivalence class.

Learning equivariance: When F lacks equivariance, we can learn a transformer function: $G(x) := F(\rho_{f(x)}x)$, where f predicts the transformation parameter. For rotated inputs, $G(\rho_\alpha x) = F(\rho_{f(\rho_\alpha x)}\rho_\alpha x)$. An optimal f would predict $-\alpha + K$, yielding $G(\rho_\alpha x) = F(\rho_K x)$, requiring only one canonical rotation to learn F. This corresponds to Fig. 2(c-d), where Canonicalization and STN approaches capture appearance but require extensive training data for transformation estimation.

Decomposing composite transformations: For compositions like $T_t\rho_\alpha$, our two-step approach maximizes data efficiency: $H(x) := F(\rho_{f^\rho(y)}y)$ where $y(x) = T_{f^T(x)}x$. With optimal transformation predictors, $H(T_t\rho_\alpha x)$ removes all variation factors. Making f^T translation-equivariant and rotation-invariant, while making f^ρ rotation-equivariant (Fig. 2(e-f)), allows independent learning of fine translations and continuous rotations, breaking the combinatorial requirement for training variations.

4.3 LEARNABLE STEERABLE RIESZ-ENHANCED EQUIVARIANT STN BASED NETWORK: FROM VARIATIONAL AUTOENCODING TO OBJECT DISCOVERY

In this section we introduce the hybrid rotation, scale and translation equivariant architecture as illustrated in Figure 1. It combines our learnable steerable Riesz transform with STNs and ReResNet to learn object representations in an unsupervised way while maintaining perfect equivariance.

Given an input image $x \in \mathbb{R}^{H\times W\times C}$, the goal is to produce a reconstructed image $\hat{x}$ while learning features that are equivariant to transformations $g \in G$, where G represents the group of 2D translations, scales and rotations. First, input image x goes through the encoder which has the **LeaRN** upstream the ReResNet followed by the STN. It extracts the features, followed by the group convolutions to extract Roto-Scale-Translation Equivariant features (RST-Equiv. features). Group invariant pooling then aggregates it to produce rotation-invariant features $\mathcal{A}_{\text{inv}} \in \mathbb{R}^{H'\times W'\times K}$: $\mathcal{A}_{\text{inv}}(t, s) = \max_r \mathcal{A}_{\text{RST}}(t, s, r)$. These rotation-invariant features are then processed by an STN to estimate the translation and scale parameters (t, s). The STN outputs a translation matrix $T(t) \in \mathbb{R}^{2\times 3}$, which is applied to the input feature map to align it spatially. The scaled and translated features $\mathcal{A}_{\text{trans}}$ are obtained by applying $T(t)$ to $\mathcal{A}_{\text{RST}}$ via a grid sampler, which according to the transformation produces a glimpse of the object. Next, the scaled and translated glimpse $\mathcal{A}_{\text{trans}} \in \mathbb{R}^{H'\times W'\times K\times R}$ is used to estimate the rotation parameter α probabilistically. The glimpse is passed with α back to the STN, producing aligned glimpse $\mathcal{A}_{\text{aligned}}$, which is rotation, scale and translation-equivariant. This network that sequentially estimates the transformation parameters is what we call **EqSTN**. The aligned features from the network are passed through a glimpse network to reconstruct glimpse, which when passed through a reverse sampler reconstructs the full image $\hat{x}$ with the correct position, scale and orientation of the object. This two-step disentanglement of translation, scale and rotation together with the Riesz network (**LeaRN-EqSTN**) reduces the combinatorial complexity of learning $SE(2)$-equivariant features, as explained before. We implement our architecture in VAE (Kingma et al.) and glimpse-based object discovery (Karazija et al., 2021) settings.

VAE: We use a gaussian mixture VAE (Dilokthanakul et al., 2016; Yang et al., 2019) for autoencoding settings. This setting is not arbitrary, it has been chosen specifically to help build the object detection setting, which is a glimpse-based architecture with VAE at its core. But, our approach, since it inherently provides an inductive bias to the feature learning, can essentially be also included in, for instance for UNet-based image generative models like Diffusion or Conditional Flow Matching. We structure latent space with: z_{pres} (object presence), z_{what} (object appearance), z_{cls} (class probabilities), and z_α (rotation angle), while the scale and translation are predicted deterministically as described in

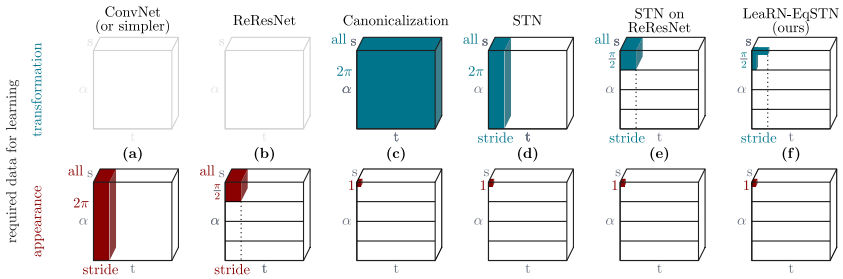

Figure 2: Illustration of the variations necessary in the training set to learn the transformation estimator (if present) and the downstream "appearance" model.

the above section. The rotation latent is modeled using a circular normal i.e. a Von Mises distribution (Davidson et al., 2018): $p(\mathbf{z}_\alpha) = \text{VM}(\mu_\alpha, \kappa)$.

Object Discovery: Our object discovery model is in the likes of SPAIR (Crawford & Pineau, 2019), GMAIR (Zhu et al., 2022), and SPACE (Lin et al., 2020). Our encoding model follows the same variational settings as above. The input is divided into $H \times W$ grid. For each cell (i, j), it learns: $z^{ij}_{\text{what}} \in \mathbb{R}^A$, $z^{ij}_{\text{depth}} \in \mathbb{R}$, $z^{ij}_{\text{pres}} \in [0, 1]$, $z^{ij}_{\text{where}} \in \mathbb{R}^5$, $z^{ij}_{\text{cls}} \in [0, 1]^C$, and $z^{ij}_{\alpha} \in [-\pi, \pi]$. Rotation-invariant variables (z_{pres}, z_{where}, z_{depth}, z_{cls}) enable stable object discovery, while rotation-equivariant z_α captures angular information. We have a version with and without a background model, the former being inspired from SPACE together with an additional latent category variable modeled after Gaussian mixtures as in GMAIR (see Appendix E). We call this model **SPAGMACE**. We introduce a fake bounding-box loss $\mathcal{L}_{fakebbox} = \sum_i \sum_j (1 - \gamma^{ij})^2 \cdot z^{ij}_{pres}$ to discourage transparent detections with non-zero presence probability.

5 Experimental Evaluation

Our primary task of focus is the one of object discovery synthesized in Tab.2. To compare our model in a simpler setting and isolate the effect of different components, we also compare it in a more traditional autoencoding setting (images with a single object) in Tab.3, and on classification in Tab.5.

Model	Accuracy (%)
GroupConv (GCNN)	82.76
Standard CNN	70.41
Riesz + CNN	89.78
Riesz + GCNN	93.45
Steerable Riesz + GCNN	95.76
LeaRN + GCNN	98.44
Standard DFT + CNN	73.47
Standard DFT + GCNN	83.58

Table 1: Classification accuracies of different models.

5.1 Datasets

In our experimental framework, we employ a variety of datasets tailored to unsupervised representation learning. For the VAE studies, we use the rotated MNIST (Rotated-MNIST) dataset to create a roto-scale MNIST. Each variant incorporates images rotated randomly according to $\mathcal{U}(0, 2\pi)$ and random scaling sampled from $\mathcal{U}(0.5, 2.5)$. Additionally, our work includes three real-world datasets. Firstly, the Tomotwin-100 Cryo-EM benchmark dataset (Jeon et al., 2024) is used. We extract 10 classes out of the available 100, focusing on non-rotation symmetric objects, and select the SNR 0.01 version to highlight the noise-filtering capability of our learnable Riesz. We also engage with the WHOI-Plankton dataset (Orenstein et al., 2015). Given its noticeable class imbalance, we reweigh it into 10 equally balanced classes featuring 1000 training and 200 test images, apply a circular crop, and resize them to 32×32 pixels following the procedure established by the state-of-the-art model IRL-INR. Lastly, among the MVTEC dataset (Bergmann et al., 2019), we leverage three classes with specific orientations: screw, hazelnut, and cable.

For object discovery, a synthetic dataset Multi-TranslationRotationScaling-MNIST (MTRS-MNIST) is crafted to evaluate equivariance regarding scale, rotation, and translation. More precisely, we

	Low Data			More Data		
	AP↗	mAP↗	RoE↘	AP↗	mAP↗	RoE↘
MTRS-MNIST						
SPAIR	75.4	×	×	81.43	×	×
GMAIR	75.48	51.23	×	82.04	57.13	×
GNM	79.14	×	×	85.97	×	×
LeaRN-EqSTN+GMAIR	**94.32**	**70.44**	**0.35**	**95.77**	**72.11**	**0.39**
(ablations)						
EqSTN+GMAIR	82.59	61.87	0.55	85.14	65.3	0.49
ReResNet+GMAIR	78.26	58.75	×	80.34	61.86	×
GMAIR+$\mathcal{L}_{fakebbox}$	78.74	58.34	×	80.79	60.97	×
MVTec D2S						
SPACE	75.34	×	×	82.9	×	×
SPAGMACE	74.15	42.24	×	84.09	43.8	×
GNM	81.48	×	×	**87.37**	×	×
LeaRN-EqSTN+SPAGMACE	**84.34**	**61.70**	**0.61**	85.97	**67.25**	**0.56**
LeaRN-EqSTN+SPACE	84.19	×	0.62	85.83	×	0.56
MVTec Screws						
SPACE	67.55	×	×	72.17	×	×
SPAGMACE	71.43	59.76	0.47	74.54	59.97	0.47
GNM	71.85	×	×	71.91	×	×
LeaRN-EqSTN+SPAGMACE	77.42	**62.12**	**0.44**	78.49	**61.78**	**0.45**
LeaRN-EqSTN+SPACE	**79.42**	×	×	**79.92**	×	×
MQRT-MNIST						
LeaRN-EqSTN+GMAIR	94.8	67.33	0.69	96.0	70.58	0.61

Table 2: Object discovery models comparison on detection and orientation metrics across datasets.

	More Data			Low Data		
	NMI	ARI	SSIM	NMI	ARI	SSIM
SR-MNIST						
EqSTN+GMVAE	0.81	0.70	0.92	0.68	0.59	0.81
TARGET VAE	0.78	0.65	0.86	0.63	0.51	0.77
IRL-INR	0.83	0.78	0.94	0.65	0.51	0.69
LeaRN-EqSTN+GMVAE	**0.88**	**0.80**	**0.95**	**0.78**	**0.62**	**0.83**
LeaRN+TARGET VAE	0.82	0.72	0.93	0.69	0.60	0.82
Tomtwin						
EqSTN+GMVAE	0.62	0.55	0.72	0.55	0.46	0.63
TARGET VAE	0.60	0.51	0.67	0.50	0.40	0.60
IRL-INR	—	—	—	—	—	—
LeaRN-EqSTN+GMVAE	**0.72**	**0.62**	**0.74**	**0.67**	**0.48**	**0.65**
LeaRN+TARGET VAE	0.64	0.56	0.72	0.58	0.47	0.64
Plankton						
EqSTN+GMVAE	0.64	0.57	0.75	0.57	0.48	0.66
TARGET VAE	0.61	0.53	0.70	0.50	0.41	0.62
IRL-INR	0.65	0.63	0.76	0.51	0.41	0.56
LeaRN-EqSTN+GMVAE	**0.75**	**0.65**	**0.77**	**0.69**	**0.50**	**0.67**
LeaRN+TARGET VAE	0.65	0.58	0.75	0.56	0.49	0.66
MVTec						
EqSTN+GMVAE	0.97	0.95	0.88	0.94	0.91	0.75
TARGET VAE	0.98	0.94	0.82	0.9	0.87	0.67
IRL-INR	0.92	0.88	0.75	0.84	0.78	0.61
LeaRN-EqSTN+GMVAE	**0.98**	**0.96**	**0.89**	**0.95**	**0.92**	**0.76**
LeaRN+TARGET VAE	0.97	0.95	0.88	0.94	0.91	0.75

Table 3: VAE clustering and reconstruction metrics across datasets.

generate composite images by randomly choosing 1-4 digits and subjecting them to random rotations from $\mathcal{U}(0, 2\pi)$ and scales from $\mathcal{U}(0.5, 2.5)$, positioning them arbitrarily on a 128x128 pixel black background, thus creating diverse configurations to test the model's adaptability to various orientations, scales, and placements. Additional datasets include MVTec D2S (Follmann et al., 2018) and MVTec Screws (MVTec., 2022). To evaluate model performance under varying data settings, we construct two versions of each dataset one with the whole dataset that we call More Data (MD) and one with 25% of the whole data, that we refer as Low Data (LD). Moreover, a specialized MQTR-MNIST variant focuses on classes 3 and 4, where the training set images are confined to rotations within $[0, \frac{\pi}{4}]$, while the test set spans rotations beyond $\frac{\pi}{4}$.

5.2 Experiments and Results

We compare our proposed model LeaRN-EqSTN+GMVAE to benchmark unsupervised equivariant models for representation learning IRL-INR (Kwon et al., 2023), TARGET-VAE (Nasiri & Bepler, 2022) and EqSTN+GMVAE, unlike (Kaba et al., 2023). Quantitative results of (Cha et al., 2025), are poor hence we show them in Appendix F.1. In addition, we evaluate the performance of our Learnable Riesz transform when plugged to TARGET-VAE. Evaluation focuses on reconstruction fidelity (SSIM, Wang et al. (2004)) and clustering effectiveness (NMI, Strehl & Ghosh (2002), ARI (Rand, 1971)), with latent space clustering using k-means. All models use a fixed latent dimension of 32; our model uses $N = 8$ discrete rotations, same as TARGET-VAE P_8, a detailed ablation study is provided in Appendix F.1.

We evaluate the object discovery models on their accuracy in predicting bounding boxes, rotation equivariance, and data efficiency for recognizing objects across rotations, scales, and translations without extensive data augmentation. We compare our model LeaRN-EqSTN+GMAIR GNM, GMAIR and SPAIR on MTRS-MNIST, and SPACE, GNM, LeaRN-EqSTN+SPACE, LeaRN-EqSTN+SPAGMACE on the other two datasets with background clutter (since they can handle backgrounds). Object discovery performance is measured using Average Precision (AP) and class-specific mean Average Precision (mAP) (Buckland & Gey, 1994; Ruppert, 2004). Secondly, to demonstrate rotation-equivariance, we design a custom metric that assesses the consistency of the predicted angles, focusing on relative consistency rather than absolute accuracy, as there is no predefined reference angle for the predictions. The normalized entropy of the angle difference between predicted and actual angles modulo 2π, is here used as a metric. For

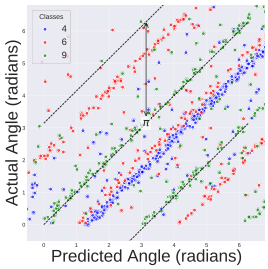

Figure 3: Correlation between predicted rotations and true object orientations.

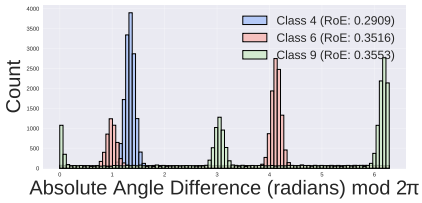

Figure 4: Histogram of differences between predicted and true object orientations (modulo 2π).

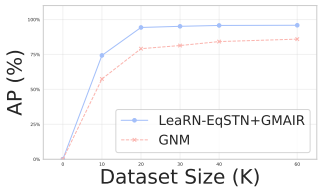

Figure 5: Effect of dataset size on class-independent bounding box predictions for object discovery.

this purpose, interval $[0, 2\pi]$ is divided into n regular bins, the corresponding discrete probability density function $p_{i=1,..,n}$ is estimated from statistical frequencies, and its Shannon entropy $H(p) = -\sum_{i=1}^{n} p_i \log_2 p_i$ is assessed. The metric named *Rotation-offset Entropy* (RoE) is formed by normalizing entropy: RoE $= {}^{H}/_{\log_2(n)}$. RoE ranges from 0 to 1, indicating the level of uncertainty in angle difference, where 1 represents high disorder and 0 represents low disorder. Finally, we evaluate the model's generalization capabilities by training and testing on quarterly rotated MNIST (MQRT MNIST), and testing its predictions on out of distribution orientations.

Results: In VAE experiments, LeaRN-EqSTN+GMVAE achieves better performance against all models in terms of NMI, ARI, and SSIM metrics, with notably superior performance in low-data regimes. In object discovery experiments, LeaRN-EqSTN+GMAIR and LeaRN-EqSTN+SPAGMACE have the highest AP scores across all the evaluated datasets, with the highest classification mAP scores, improved performance scaling with increased data availability, and consistent angle predictions with low Rotation-offset Entropy scores. Our architecture exhibits robust generalization capabilities, maintaining the best performance even with limited training data. We also report on average of all datasets, the lowest Lie derivative equivariance error (LEE) (Table: 4 for learned equivariance as introduced in Gruver et al. (2022). Further, our ablation studies confirm the effectiveness of our integrated approach, with the complete model consistently outperforming partial variants across all metrics, validating our architectural design choices.

Table 4: Equivariance Error Comparison(↓)

Model	**LEE**
TARGET-VAE	0.184
LeaRN+TARGET-VAE	0.162
EqSTN+GMVAE	0.159
LeaRN-EqSTN+GMVAE	**0.097**

6 Conclusion

In this work, we introduce a learnable steerable Riesz transform integrated with Spatial Transformer Networks (STNs) and equivariant architectures to effectively model composite transformations in an unsupervised learning framework. By incorporating sequential transformation estimation, our approach improves model generalization with limited data and reduces data complexity. Experiments on VAEs and object discovery tasks demonstrate enhanced data efficiency, faster training, and improved transformation awareness compared to traditional methods. Our findings highlight the potential of combining learned and enforced equivariance, paving the way for more robust and data efficient unsupervised learning models. Future work could focus on leveraging the Riesz transform's SO(3) representation for seamless integration with SO(3)-equivariant CNNs.

References

Vincent Andrearczyk, Julien Fageot, Valentin Oreiller, Xavier Montet, and Adrien Depeursinge. Exploring local rotation invariance in 3d cnns with steerable filters. In *International Conference on Medical Imaging with Deep Learning*, pp. 15–26. PMLR, 2019.

Tin Barisin, Jesus Angulo, Katja Schladitz, and Claudia Redenbach. Riesz feature representation: Scale equivariant scattering network for classification tasks. *SIAM Journal on Imaging Sciences*, 17(2):1284–1313, 2024a.

Tin Barisin, Katja Schladitz, and Claudia Redenbach. Riesz networks: Scale-invariant neural networks in a single forward pass. *Journal of Mathematical Imaging and Vision*, 66(3):246–270, 2024b.

Arash Behboodi, Gabriele Cesa, and Taco S Cohen. A pac-bayesian generalization bound for equivariant networks. *Advances in Neural Information Processing Systems*, 35:5654–5668, 2022.

Erik J Bekkers, Maxime W Lafarge, Mitko Veta, Koen AJ Eppenhof, Josien PW Pluim, and Remco Duits. Roto-translation covariant convolutional networks for medical image analysis. In *Medical Image Computing and Computer Assisted Intervention–MICCAI 2018: 21st International Conference, Granada, Spain, September 16-20, 2018, Proceedings, Part I*, pp. 440–448. Springer, 2018.

Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9592–9600, 2019.

Anamaria Bjelopera, Emil Dumic, and Sonja Grgic. Classification of image degradation using riesz transform. In *2016 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pp. 1–4, 2016. doi: 10.1109/IWSSIP.2016.7502741.

Michael Buckland and Fredric Gey. The relationship between recall and precision. *Journal of the American society for information science*, 45(1):12–19, 1994.

Gabriele Cesa, Leon Lang, and Maurice Weiler. A program to build e (n)-equivariant steerable cnns. In *International conference on learning representations*, 2022.

Jaehoon Cha, Jinhae Park, Samuel Pinilla, Kyle L Morris, Christopher S Allen, Mark I Wilkinson, and Jeyan Thiyagalingam. Discovering fully semantic representations via centroid-and orientation-aware feature learning. *Nature Machine Intelligence*, pp. 1–8, 2025.

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

Eric Crawford and Joelle Pineau. Spatially invariant unsupervised object detection with convolutional neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 3412–3420, 2019.

Tim R Davidson, Luca Falorsi, Nicola De Cao, Thomas Kipf, and Jakub M Tomczak. Hyperspherical variational auto-encoders. In *34th Conference on Uncertainty in Artificial Intelligence 2018, UAI 2018*, pp. 856–865. Association For Uncertainty in Artificial Intelligence (AUAI), 2018.

Adrien Depeursinge, Antonio Foncubierta-Rodriguez, Dimitri Van de Ville, and Henning Müller. Lung texture classification using locally–oriented riesz components. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2011: 14th International Conference, Toronto, Canada, September 18-22, 2011, Proceedings, Part III 14*, pp. 231–238. Springer, 2011.

Adrien Depeursinge, Antonio Foncubierta-Rodriguez, Dimitri Van de Ville, and Henning Müller. Rotation–covariant texture learning using steerable riesz wavelets. *IEEE Transactions on Image Processing*, 23(2):898–908, 2013.

Nat Dilokthanakul, Pedro AM Mediano, Marta Garnelo, Matthew CH Lee, Hugh Salimbeni, Kai Arulkumaran, and Murray Shanahan. Deep unsupervised clustering with gaussian mixture variational autoencoders. *arXiv preprint arXiv:1611.02648*, 2016.

Tom Edixhoven, Attila Lengyel, and Jan C van Gemert. Using and abusing equivariance. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 119–128, 2023.

Martin Engelcke, Oiwi Parker Jones, and Ingmar Posner. Genesis-v2: Inferring unordered object representations without iterative refinement. *Advances in Neural Information Processing Systems*, 34:8085–8094, 2021.

Patrick Follmann, Tobias Bottger, Philipp Hartinger, Rebecca Konig, and Markus Ulrich. Mvtec d2s: densely segmented supermarket dataset. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 569–585, 2018.

David Gilbarg, Neil S Trudinger, David Gilbarg, and NS Trudinger. *Elliptic partial differential equations of second order*, volume 224. Springer, 1977.

Nate Gruver, Marc Finzi, Micah Goldblum, and Andrew Gordon Wilson. The lie derivative for measuring learned equivariance. *arXiv preprint arXiv:2210.02984*, 2022.

Jiaming Han, Jian Ding, Nan Xue, and Gui-Song Xia. Redet: A rotation-equivariant detector for aerial object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2786–2795, 2021.

Forrest Hoffman. An introduction to fourier theory. *Extraído el*, 2, 1997.

Max Jaderberg, Kavita Simonyan, Andrew Zisserman, and Koray Kavukcuoglu. Spatial transformer networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 2017–2025, 2015.

Minkyu Jeon, Rishwanth Raghu, Miro Astore, Geoffrey Woollard, Ryan Feathers, Alkin Kaz, Sonya M. Hanson, Pilar Cossio, and Ellen D. Zhong. Cryobench: Diverse and challenging datasets for the heterogeneity problem in cryo-em. In *Advances in Neural Information Processing Systems*, 2024.

Jindong Jiang and Sungjin Ahn. Generative neurosymbolic machines. *Advances in Neural Information Processing Systems*, 33:12572–12582, 2020.

Ranveer Joyseeree, Sebastian Otálora, Henning Müller, and Adrien Depeursinge. Fusing learned representations from riesz filters and deep cnn for lung tissue classification. *Medical image analysis*, 56:172–183, 2019.

Sékou-Oumar Kaba, Arnab Kumar Mondal, Yan Zhang, Yoshua Bengio, and Siamak Ravanbakhsh. Equivariance with learned canonicalization functions. In *International Conference on Machine Learning*, pp. 15546–15566. PMLR, 2023.

Laurynas Karazija, Iro Laina, and Christian Rupprecht. Clevrtex: A texture-rich benchmark for unsupervised multi-object segmentation. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.

Diederik P Kingma, Max Welling, et al. Auto-encoding variational bayes.

Sehyun Kwon, Joo Young Choi, and Ernest K Ryu. Rotation and translation invariant representation learning with implicit neural representations. In *International Conference on Machine Learning*, pp. 18037–18056. PMLR, 2023.

Maxime W Lafarge, Erik J Bekkers, Josien PW Pluim, Remco Duits, and Mitko Veta. Roto-translation equivariant convolutional networks: Application to histopathology image analysis. *Medical Image Analysis*, 68:101849, 2021.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Eung-Hyun Lee. Aliasing error in muliti-channel sampling. 2004.

Zhixuan Lin, Yi-Fu Wu, Skand Vishwanath Peri, Weihao Sun, Gautam Singh, Fei Deng, Jindong Jiang, and Sungjin Ahn. Space: Unsupervised object-oriented scene representation via spatial attention and decomposition. In *International Conference on Learning Representations*, 2020.

MVTec. Mvtec screws dataset. *https://www.mvtec.com/company/research/datasets/mvtec-screws*, 2022. URL `https://www.mvtec.com/company/research/datasets/mvtec-screws`.

Alireza Nasiri and Tristan Bepler. Unsupervised object representation learning using translation and rotation group equivariant vae. *Advances in Neural Information Processing Systems*, 35: 15255–15267, 2022.

Avital Oliver, Augustus Odena, Colin A Raffel, Ekin Dogus Cubuk, and Ian Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. *Advances in neural information processing systems*, 31, 2018.

Eric C Orenstein, Oscar Beijbom, Emily E Peacock, and Heidi M Sosik. Whoi-plankton-a large scale fine grained visual recognition benchmark dataset for plankton classification. *arXiv preprint arXiv:1510.00745*, 2015.

Keiron O'shea and Ryan Nash. An introduction to convolutional neural networks. *arXiv preprint arXiv:1511.08458*, 2015.

Marc-Antoine Parseval. Mémoire sur les séries et sur l'intégration complète d'une équation aux différences partielles linéaires du second ordre, à coefficients constants. *Mém. prés. par divers savants, Acad. des Sciences, Paris,(1)*, 1(638-648):42, 1806.

Saro Passaro and C Lawrence Zitnick. Reducing so (3) convolutions to so (2) for efficient equivariant gnns. In *International conference on machine learning*, pp. 27420–27438. PMLR, 2023.

C Rafael. Gonzalez, richard e. woods. digital image processing. prentice hall. 2002.

William M Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336):846–850, 1971.

David Ruppert. The elements of statistical learning: data mining, inference, and prediction, 2004.

Alexander Strehl and Joydeep Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of machine learning research*, 3(Dec):583–617, 2002.

Michael Unser and Dimitri Van De Ville. Wavelet steerability and the higher-order riesz transform. *IEEE Transactions on Image Processing*, 19(3):636–652, 2010. doi: 10.1109/TIP.2009.2038832.

van A Van der Schaaf and JH van van Hateren. Modelling the power spectra of natural images: statistics and information. *Vision research*, 36(17):2759–2770, 1996.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Maurice Weiler and Gabriele Cesa. General e (2)-equivariant steerable cnns. *Advances in neural information processing systems*, 32, 2019.

Maurice Weiler, Fred A Hamprecht, and Martin Storath. Learning steerable filters for rotation equivariant cnns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 849–858, 2018.

Linxiao Yang, Ngai-Man Cheung, Jiaying Li, and Jun Fang. Deep clustering by gaussian mixture variational autoencoders with graph embedding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 6440–6449, 2019.

Richard Zhang. Making convolutional networks shift-invariant again. In *International conference on machine learning*, pp. 7324–7334. PMLR, 2019.

Weijin Zhu, Yao Shen, Mingqian Liu, and Lizeth Patricia Aguirre Sanchez. Gmair: Unsupervised object detection based on spatial attention and gaussian mixture model. *Computational Intelligence and Neuroscience*, 2022(1):7254462, 2022.

Larry Zitnick, Abhishek Das, Adeesh Kolluru, Janice Lan, Muhammed Shuaibi, Anuroop Sriram, Zachary Ulissi, and Brandon Wood. Spherical channels for modeling atomic interactions. *Advances in Neural Information Processing Systems*, 35:8054–8067, 2022.

Xueyan Zou, Fanyi Xiao, Zhiding Yu, Yuheng Li, and Yong Jae Lee. Delving deeper into anti-aliasing in convnets. *International Journal of Computer Vision*, 131(1):67–81, 2023.

A APPENDIX

In this document, we present the materials that were summarized due to space limitation in the main text. It is organized as follows:

- **Appendix A** contains some details to help for the introduction of the LeaRN-EqSTN model, details about spatial transformers, group convolutional layers, steerable CNNs and Riesz transform.
- **Appendix B** reports on the aliasing phenomenon in CNNs and the Riesz transform approach to solving it.
- **Appendix C** includes a proof of Theorem 2 highlighting the impact of the Riesz transform in aligning representations with equivariant features, and details the integration of the Riesz transform within a group-equivariant CNN.
- **Appendix D** focuses on unsupervised detection in presence of multiple objects, showing the potential of transformation invariance and equivariance in terms of training data-efficiency, together with additional training time measurements.
- **Appendix E** provides the empirical and qualitative results, with ablation studies to demonstrate the effectiveness of each component in our model.

B BACKGROUND WITH SOME DETAILS

Related details about the concepts on which our architecture (illustrated in Fig. 1 of the main paper) is based on, are given in this section.

B.1 SPATIAL TRANSFORMER NETWORKS

An STN transforms an input feature map $U \in \mathbb{R}^{H\times W\times C}$ (e.g., an image with height H, width W, and channels C) into an output feature map $V \in \mathbb{R}^{H'\times W'\times C}$. It consists of three components: a *localization network*, a *grid generator*, and a *sampler*.Jaderberg et al. (2015)

- **The localization Network**, $f_{\text{loc}} : \mathbb{R}^{H\times W\times C} \to \mathbb{R}^k$ predicts the parameters of spatial transformation. For a 2D affine transformation ($k = 6$): the parameters are $\theta = \{a_{11}, a_{12}, a_{21}, a_{22}, t_x, t_y\}$ and the transformation matrix is:

$$T_\theta = \begin{bmatrix} a_{11} & a_{12} & t_x \\ a_{21} & a_{22} & t_y \\ 0 & 0 & 1 \end{bmatrix}$$

  This matrix operates in homogeneous coordinates to map input coordinates (x_s, y_s) to output coordinates (x_t, y_t).

- **The Grid Generator** computes a sampling grid by applying the inverse transformation T_θ^{-1} to output coordinates (x_t, y_t):

$$\begin{bmatrix} x_s \\ y_s \\ 1 \end{bmatrix} = T_\theta^{-1} \begin{bmatrix} x_t \\ y_t \\ 1 \end{bmatrix}$$

  For an affine transformation:

$$\begin{bmatrix} x_s \\ y_s \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}^{-1} \left(\begin{bmatrix} x_t \\ y_t \end{bmatrix} - \begin{bmatrix} t_x \\ t_y \end{bmatrix} \right)$$

  Coordinates are typically normalized to $[-1, 1]$ for consistency across image sizes.

- **The Sampler** uses bilinear interpolation to compute image $V(x_t, y_t)$ from image U at source coordinates (x_s, y_s):

$$V(x_t, y_t) = \sum_{n=1}^{H} \sum_{m=1}^{W} U(m, n) \cdot \max(0, 1 - |x_s - m|) \cdot \max(0, 1 - |y_s - n|)$$

  The interpolation kernel $\max(0, 1-|x_s-m|)\cdot\max(0, 1-|y_s-n|)$ ensures differentiability, enabling gradient flow. This is applied independently to each channel.

TRAINING

The STN is trained end-to-end by minimizing a task-specific loss whose gradients with respect to parameters θ are backpropagated through the sampler and grid generator to the localization network, enabling the STN to learn a geometric transformation optimizing the current task. For example, the localization network is intended to deliver the following matrix:

$$T_\theta = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

when a rotation by angle α is applied, transforming target points with coordinates $(x_t, y_t)^T$ into source coordinates $(x_s, y_s)^T$. The sampler interpolates from source image U to warp it into image V.

LIMITATIONS

STNs face several challenges:

1. *Lack of Equivariance*: Unlike convolutional neural networks (CNNs) with translational equivariance, STNs must learn transformations explicitly, requiring diverse training data to cover the transformation space (e.g., the 6D affine group).
2. *Exponential Complexity*: In scenarios with multiple objects, the transformation space grows as $|\text{Aff}(2)|^N$, where N is the number of objects, making comprehensive training infeasible.
3. *Data Dependence*: Generalization depends on training data diversity. Without examples of certain transformations (e.g., large rotations), STNs may fail on out-of-distribution data.
4. *Stability*: The matrix $\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$ must be non-singular for T_θ^{-1} to exist. Unstable θ predictions can cause numerical issues.

COMPARISON TO EQUIVARIANT ARCHITECTURES

Equivariant networks encode symmetries (e.g., translation or rotation) directly, reducing data requirements. For example, a CNN satisfies $f(U(\cdot - \mathbf{t})) = f(U)(\cdot - \mathbf{t})$ for translations $\mathbf{t} = (t_x, t_y)^T$. STNs approximate this by learning t_x, t_y, which is less robust without sufficient data. STNs provide a flexible, differentiable framework for learning spatial transformations, integrating localization, grid generation, and sampling. However, their reliance on data-driven learning limits generalization in complex or under-sampled transformation spaces. Hybrid approaches combining STNs with equivariant architectures may address these challenges.

B.2 GROUP CONVOLUTIONS

Standard convolutional neural networks (CNNs) are effective for handling translations due to their translational equivariance but struggle with rotations, as they do not inherently recognize that a rotated object is equivalent to its unrotated counterpart. Group convolutional layers Cohen & Welling (2016) and steerable CNNs address this limitation by embedding rotational symmetries into the network architecture, reducing the need for extensive data augmentation. This section provides a mathematical overview of these methods, focusing on their mechanisms and advantages.

GROUP CONVOLUTIONAL LAYERS

Group convolutional layers leverage the symmetry group P_r, which combines translations in $\mathbb{Z}^2$ with r discrete rotations by angles $\theta_k = k \cdot \frac{2\pi}{r}$, where $k \in \{0, 1, \ldots, r-1\}$. The group P_r represents transformations that include both a translation by vector $x \in \mathbb{Z}^2$ and a rotation by angle α.

GROUP CONVOLUTION

For an input feature map $f : \mathbb{Z}^2 \to \mathbb{R}$ and a kernel $\psi : \mathbb{Z}^2 \to \mathbb{R}$, the group convolution over P_r is defined as:

$$(f \star \psi)(g) = \sum_{x \in \mathbb{Z}^2} f(x)\psi(g^{-1}x),$$

where $g \in P_r$ is a group element combining a translation and a rotation. The inverse g^{-1} applies the reverse transformation (e.g., rotating by $-\theta_k$ and translating backward by x). For each g, the kernel is transformed (e.g., rotated by θ_k), and the convolution produces a feature map indexed by both translation and rotation.

EQUIVARIANCE

The group convolution ensures *equivariance* to P_r-transformations. If the input f is transformed by $h \in P_r$, i.e., $f'(x) = f(h^{-1}x)$, the output feature map transforms predictably:

$$(f' \star \psi)(g) = (f \star \psi)(h^{-1}g).$$

For example, with $r = 4$ (rotations by $0°, 90°, 180°, 270°$), kernel ψ is applied at each rotation, producing r feature maps. If the input rotates by $90°$, the feature maps shift cyclically, preserving the rotational structure.

B.3 STEERABLE CNNS

Steerable CNNs extend rotational equivariance to the continuous rotation group $SO(2)$, handling arbitrary angles efficiently using *steerable filters*.

STEERABLE FILTERS

A steerable filter $\Psi : \mathbb{R}^2 \to \mathbb{R}$ is expressed as a linear combination of basis functions:

$$\Psi(x) = \sum_{m=1}^{M} c_m \psi_m(x),$$

where $\psi_m : \mathbb{R}^2 \to \mathbb{R}$ are basis functions (e.g., circular harmonics) and c_m are learnable coefficients. When rotated by $\alpha \in SO(2)$, the filter transforms as:

$$\Psi'(x) = \Psi(\rho_\alpha^{-1}x),$$

where ρ_α is the 2D rotation matrix:

$$\rho_\alpha = \begin{bmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{bmatrix}.$$

The basis functions ψ_m are chosen such that $\Psi'(x)$ can be expressed as a linear combination of the same ψ_m, with transformed coefficients c'_m.

CONVOLUTION WITH STEERABLE FILTERS

The convolution with a steerable filter Weiler & Cesa (2019); Weiler et al. (2018); Andrearczyk et al. (2019); Cesa et al. (2022) produces a feature map indexed by both spatial position $x \in \mathbb{R}^2$ and rotation angle α:

$$(f \star \Psi)(x, \alpha) = \int_{\mathbb{R}^2} f(y)\Psi(\rho_\alpha^{-1}(x - y))\, dy.$$

This feature map is equivariant to rotations: if the input is rotated by β, the output shifts in the α-dimension:

$$(f(\rho_\beta^{-1}\cdot) \star \psi)(x, \alpha) = (f \star \psi)(\rho_\beta^{-1}x, \alpha + \beta).$$

COMPARISON TO STANDARD CNNS

Standard CNNs are translation-equivariant but not rotation-equivariant. A rotated input produces a different feature map, requiring data augmentation to learn rotational invariance. Group and steerable CNNs address rotational equivariance directly to ensure that a rotated input produces a predictably transformed output, enhancing robustness with less training data.

B.4 Riesz Transform and Properties

The Riesz transform is a multidimensional generalization of the Hilbert transformHoffman (1997); Gilbarg et al. (1977), operating on functions defined over $\mathbb{R}^d$. For a function $f : \mathbb{R}^2 \to \mathbb{R}$ (e.g., an image channel), the 2D Riesz transform is defined as a vector-valued operator $\mathcal{R} = (\mathcal{R}_1, \mathcal{R}_2)$ where each component $\mathcal{R}_j$ $(j = 1, 2)$ is given by:

$$\mathcal{R}_j f(x) = \lim_{\epsilon \to 0} \frac{1}{\pi} \int_{|y|>\epsilon} \frac{y_j}{|y|^3} f(x-y)\, dy, \quad j = 1, 2,$$

where $x = (x_1, x_2)$, $y = (y_1, y_2)$, and $|y| = \sqrt{y_1^2 + y_2^2}$. This integral is understood in the principal value sense to handle the singularity at $y = 0$.

In the frequency domain, the Riesz transform is more conveniently expressed. Let $\mathcal{F}(\xi)$ denote the Fourier transform of f, defined as:

$$\mathcal{F}(\xi) = \int_{\mathbb{R}^2} f(x) e^{-i\xi \cdot x}\, dx, \quad \xi = (\xi_1, \xi_2) \in \mathbb{R}^2.$$

The Riesz transform $\mathcal{R}_j f$ in the frequency domain is:

$$\widehat{\mathcal{R}_j f}(\xi) = i \frac{\xi_j}{|\xi|} \mathcal{F}(\xi), \quad |\xi| = \sqrt{\xi_1^2 + \xi_2^2}, \quad j = 1, 2.$$

The factor $i\frac{\xi_j}{|\xi|}$ acts as a directional derivative in the frequency domain, emphasizing the j-th direction while normalizing by the frequency magnitude. In practice, a small ϵ (e.g., 10^{-8}) is added to $|\xi|$ to avoid division by zero, as implemented in the Higher-order Riesz transforms can also be defined. For example, second-order terms such as $\mathcal{R}_{xx}$, $\mathcal{R}_{yy}$, and $\mathcal{R}_{xy}$ are computed as:

$$\widehat{\mathcal{R}_{xx} f}(\xi) = -\frac{\xi_1^2}{|\xi|^2} \mathcal{F}(\xi), \quad \widehat{\mathcal{R}_{yy} f}(\xi) = -\frac{\xi_2^2}{|\xi|^2} \mathcal{F}(\xi), \quad \widehat{\mathcal{R}_{xy} f}(\xi) = -\frac{\xi_1 \xi_2}{|\xi|^2} \mathcal{F}(\xi).$$

These terms capture second-order directional information, analogous to second derivatives, and are used in our model to enrich the feature set.

The Riesz transform possesses several properties that make it particularly suitable for rotation-equivariant feature extractionUnser & Van De Ville (2010):

Isometry: The Riesz transform is an isometry in $L^2(\mathbb{R}^2)$, meaning it preserves the L^2-norm of the input signal:

$$\|\mathcal{R}_j f\|_{L^2} = \|f\|_{L^2}, \quad j = 1, 2.$$

This property ensures that the transform does not amplify or diminish the energy of the signal, making it stable for feature extraction in neural networks.

Rotation Equivariance: The Riesz transform is equivariant under the action of the rotation group $SO(2)$. For a rotation $g \in SO(2)$, represented by a 2D rotation matrix, and a rotated function $f_g(x) = f(g^{-1}x)$, the Riesz transform satisfies:

$$\mathcal{R}_j(f_g) = \rho(g)(\mathcal{R}_j f),$$

where $\rho(g)$ is the representation of g that rotates the vector field $(\mathcal{R}_1 f, \mathcal{R}_2 f)$. Specifically, if g rotates by angle α, then:

$$(\mathcal{R}_1(f_g), \mathcal{R}_2\mathcal{R}(f_g)) = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} (\mathcal{R}_1 f, \mathcal{R}_2 f).$$

This equivariance ensures that the Riesz transform's output transforms predictably under rotations, a critical property for building rotation-invariant models.

Steerability: The Riesz transform components $\mathcal{R}_1 f$ and $\mathcal{R}_2 f$ form a steerable basis, meaning that directional derivatives at any angle α can be obtained as a linear combination:

$$\text{steered}_\alpha = (\cos\alpha)\mathcal{R}_1 f + (\sin\alpha)\mathcal{R}_2 f.$$

This steerable property allows the model to compute responses at arbitrary orientations without needing to discretely rotate filters, reducing computational overhead. In our implementation, the computes both fixed orientations ($\alpha_k = \frac{k\pi}{8}$, $k = 0, \ldots, 7$) and learnable orientations, enhancing flexibility while maintaining equivariance. **Harmonic Analysis Connection**: In the Fourier domain, the Riesz transform corresponds to a projection onto the $m = \pm 1$ modes of the $SO(2)$ Fourier expansion. For a frequency ξ and angular mode m, the Riesz transform acts as:

$$\widehat{\mathcal{R}_1 f}(\xi, m) \propto \hat{f}(\xi, m-1) + \hat{f}(\xi, m+1),$$

emphasizing directional gradients that are inherently rotation-equivariant. This connection to harmonic analysis underpins its utility in group-equivariant neural networks.

C Aliasing in Standard CNNs and the Riesz transform to solving it

Aliasing occurs when a signal is sampled at a rate insufficient to capture its highest frequency components, causing high frequencies to be misrepresented as lower frequenciesRafael (2002). In CNNs, aliasing arises during downsampling operations (e.g., max-pooling or strided convolutions), which reduce the spatial resolution of feature maps. This violates the Nyquist-Shannon sampling theorem, which states that a signal must be sampled at least twice its highest frequency to be reconstructed accurately. CNNs rely on **translation equivariance**, meaning that if an input image is shifted (translated), the output feature maps shift accordingly without changing their structure. Aliasing disrupts this property by introducing distortions, particularly incorrect phase shifts in the frequency domain, leading to inconsistent feature representations. This affects the network's ability to generalize across translated inputs, critical for tasks like image recognition.Zou et al. (2023); Gruver et al. (2022); Zhang (2019)

In practice, digital processing is applied on images discretized on a grid, here assumed to be $M \times M$, and the **Discrete Fourier Transform** (DFT) is defined in the discretized frequency domain as

$$F[k,l] = \sum_{m,n=0}^{M-1} f[m,n] e^{-2\pi i(mk+nl)/M}, \quad k,l \in \{0,\ldots,M-1\}.$$

to compute the frequency components of digital image $f[m,n] = f\left(\frac{m}{M}, \frac{n}{M}\right), \quad m,n \in \{0,\ldots,M-1\}$.). Each index pair (k,l) corresponds to a frequency $\xi = (k/M, l/M)$. The discretization of the frequency domain involves a periodization of the image in the spatial domain, and the highest frequency that can be represented is $f_s = M/2$ (half the Nyquist Frequency): any frequency beyond this limit is aliased in $[-M/2, M/2]^2$.

C.1 Aliasing Due to Downsampling

Downsampling by a factor of 2 reduces the grid from $M \times M$ to $M/2 \times M/2$ by assuming here that M is an even integer. That typically occurs in CNNs in stride-2 convolution or pooling, effectively halving the spatial resolution. This is equivalent to reducing the sampling rate, which lowers the Nyquist frequency to $f'_{NS} = M/2$. The DFT of downsampled image $f_{\text{down}}[m,n] = f[2m, 2n], \quad m,n \in \{0,\ldots,(M/2)-1\}$, is:

$$F_{\text{down}}[k,l] = \sum_{m,n=0}^{M/2-1} f[2m,2n] e^{-2\pi i(mk+nl)/(M/2)}, \quad k,l \in \{0,\ldots,(M/2)-1\}.$$

To understand aliasing, we need to relate F_{down} to the original DFT F. The DFT sums over the reduced grid $m,n \in \{0,\ldots,(M/2)-1\}$. The downsampled DFT can be rewritten using the original image's DFT. The well-known standard aliasing formula gives:

$$F_{\text{down}}[k,l] = \frac{1}{4}\left(F[k,l] + F[k+M/2,l] + F[k,l+M/2] + F[k+M/2,l+M/2]\right).$$

The factor $1/4$ arises because the DFT of a downsampled signal averages contributions from frequency components that are shifted by multiples of the new sampling frequency. The terms $F[k + M/2, l]$, $F[k, l + M/2]$, and $F[k + M/2, l + M/2]$ represent high-frequency components (beyond $M/4$) that "fold" into the lower frequency range $k, l \in \{0, \ldots, M/2 - 1\}$.

The high-frequency components (e.g., $F[k + M/2, l]$) are outside the limit $M/4$. These components alias into the lower frequency range, causing distortions because they are indistinguishable from lower frequencies in the downsampled signal.

Translation and Phase Errors

Now, consider a translated image $f[m - b_0, n - b_1]$. In the frequency domain, such a translation introduces a phase shift: component $F[k, l]$ is modulated as follows: $F[k, l]e^{-2\pi i(b_0 k + b_1 l)/M}$, and the DFT after downsampling becomes $\frac{1}{4}\sum_{p,q\in\{0,M/2\}} F[k + p, l + q]e^{-2\pi i(b_0 k + b_1 l)/M} e^{-2\pi i(b_0 p + b_1 q)/M}$.. For aliased terms (e.g., $p = M/2$), the modulation term expresses as $e^{-2\pi i b_0 (k+M/2)/M} = e^{-2\pi i b_0 k/M} e^{-\pi i b_0}$. where $e^{-\pi i b_0}$ is an additional factor due to alisasing, causing a phase error which disrupts translation equivariance. The downsampled image's frequency components are modulated incorrectly, meaning the network's response to a translated input is not a simple shift of the original response. Non-linearities (e.g., ReLU) worsen this by generating higher harmonics, increasing the impact of aliased frequencies.

C.2 The Riesz Transform Solution

The Riesz transform provides a steerable, multi-directional representation. This allows for selective filtering of high-frequency components in specific directions, which can help design anti-aliasing filters that preserve important features while suppressing aliasing artifacts Lee (2004); Bjelopera et al. (2016). Our scale adaptive transform enables adaptive sampling or filtering techniques that prioritize critical signal components, reducing the risk of aliasing in areas with rapid changes.

$$\widehat{\mathcal{R}_j(f(\cdot - b))}(\xi) = i\frac{\xi_j}{|\xi|}\hat{f}(\xi)e^{-2\pi i\xi\cdot b}.$$

Since:

$$\widehat{\mathcal{R}_j f}(\xi) = i\frac{\xi_j}{|\xi|}\hat{f}(\xi),$$

we have:

$$\widehat{\mathcal{R}_j(f(\cdot - b))}(\xi) = e^{-2\pi i\xi\cdot b}\widehat{\mathcal{R}_j f}(\xi).$$

In the discrete setting:

$$G_j[k, l] = i\frac{k}{\sqrt{k^2 + l^2}}F[k, l]e^{-2\pi i(b_0 k + b_1 l)/M} \quad (j = 1).$$

The Riesz transform preserves the phase shift exactly, as $\frac{\xi_j}{|\xi|}$ is independent of translation. In the spatial domain, this corresponds to:

$$(\mathcal{R}_j f)[m - b_0, n - b_1].$$

This means the Riesz-transformed feature map shifts with the input, satisfying translation equivariance at full resolution.

Downsample the Riesz-transformed signal $g_j[m, n] = (\mathcal{R}_j f)[m, n]$:

$$g_{j,\text{down}}[m, n] = g_j[2m, 2n], \quad m, n \in \{0, \ldots, M/2 - 1\}.$$

The DFT is:

$$G_{j,\text{down}}[k, l] = \sum_{m,n=0}^{M/2-1} g_j[2m, 2n]e^{-2\pi i(mk+nl)/(M/2)}.$$

Using the aliasing formula:

$$G_{j,\text{down}}[k, l] = \frac{1}{4}\sum_{p,q\in\{0,M/2\}} G_j[k + p, l + q].$$

Translated Signal For the translated signal:

$$G_j[k+p, l+q] = i\frac{k+p}{\sqrt{(k+p)^2+(l+q)^2}} F[k+p, l+q] e^{-2\pi i(b_0(k+p)+b_1(l+q))/M} \quad (j=1).$$

The phase term is:

$$e^{-2\pi i(b_0(k+p)+b_1(l+q))/M} = e^{-2\pi i(b_0k+b_1l)/M} e^{-2\pi i(b_0p+b_1q)/M}.$$

Substitute into the downsampled DFT:

$$G_{j,\text{down}}[k,l] = \frac{1}{4} \sum_{p,q\in\{0,M/2\}} \left[i\frac{k+p}{\sqrt{(k+p)^2+(l+q)^2}} F[k+p, l+q] e^{-2\pi i(b_0(k+p)+b_1(l+q))/M} \right].$$

Factor out the phase:

$$G_{j,\text{down}}[k,l] = \frac{1}{4} e^{-2\pi i(b_0k+b_1l)/M} \sum_{p,q\in\{0,M/2\}} \left[i\frac{k+p}{\sqrt{(k+p)^2+(l+q)^2}} F[k+p, l+q] e^{-2\pi i(b_0p+b_1q)/M} \right].$$

The downsampled DFT is:

$$G_{j,\text{down}}[k,l] = \frac{1}{4} e^{-2\pi i(b_0k+b_1l)/M} \sum_{p,q\in\{0,M/2\}} \left[i\frac{k+p}{\sqrt{(k+p)^2+(l+q)^2}} F[k+p, l+q] e^{-2\pi i(b_0p+b_1q)/M} \right].$$

The term $e^{-2\pi i(b_0k+b_1l)/M}$ is the correct phase shift for translation, factored out of the sum. The aliased terms (where $p = M/2$ or $q = M/2$) contribute additional phases, e.g.:

$$e^{-2\pi i b_0(M/2)/M} = e^{-\pi i b_0}.$$

The additional phase $e^{-\pi i b_0}$ modulates the amplitude of the aliased frequency components (e.g., $F[k + M/2, l]$). This phase does not affect the primary phase shift $e^{-2\pi i(b_0k+b_1l)/M}$. Instead, it scales the contribution of high-frequency components in the sum.

The Riesz transform's directional filter $\frac{k+p}{\sqrt{(k+p)^2+(l+q)^2}}$ encodes the signal's structure at full resolution.
- The phase shift is applied before downsampling, so aliasing only affects the combination of frequency components, not the translation equivariance.

In the spatial domain, the downsampled feature map is:

$$g_{j,\text{down}}[m,n] = g_j[2m, 2n] = (\mathcal{R}_j f)[2m, 2n].$$

For the translated signal:

$$g_{j,\text{down}}[m,n] = (\mathcal{R}_j f)[2m - b_0, 2n - b_1].$$

The downsampled feature map of the translated signal is a shifted version of the original downsampled feature map, preserving the translation structure. The aliased terms in the frequency domain affect the magnitude of the features (due to $e^{-\pi i b_0}$) but do not disrupt the phase shift, ensuring equivariance.

The Riesz transform's output G_j represents directional features that are inherently equivariant to translation. Downsampling introduces aliasing, but because the phase shift is encoded at full resolution, the aliased terms only contribute to the feature map's amplitude, not its phase structure. The directional filter ensures that the features transform predictably, even after aliasing.

The Riesz transform is computed at full resolution ($M \times M$), capturing all frequencies up to $M/2$. This prevents information loss before downsampling, unlike standard CNNs, where downsampling occurs directly on the signal or convolved features.

D RIESZ ENHANCED EQUIVARIANT NETWORKS GENERALISE BETTER

The section is devoted to the proof of Theorem 2 indicating that given an equivariant network incorporating steerable group convolutions, augmenting the network with a learnable steerable Riesz transform upstream to the initial group convolution layers leads to a more generalizable architecture able to well approximate equivariant feature maps ϕ (see definition 1).

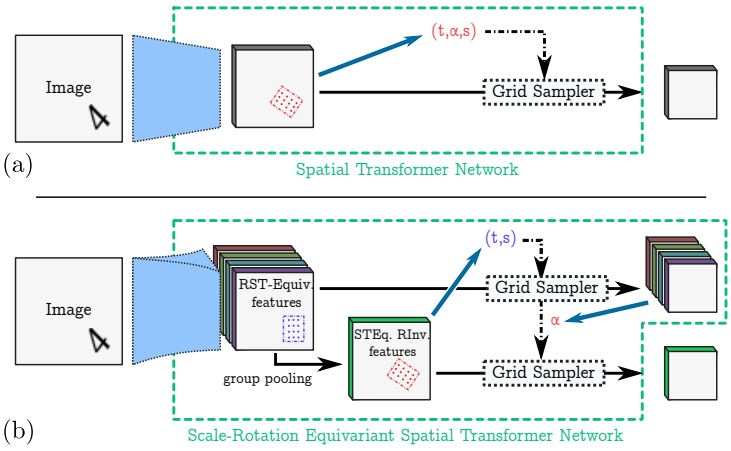

Figure 6: The figure illustrates the core difference between STN and EqSTN

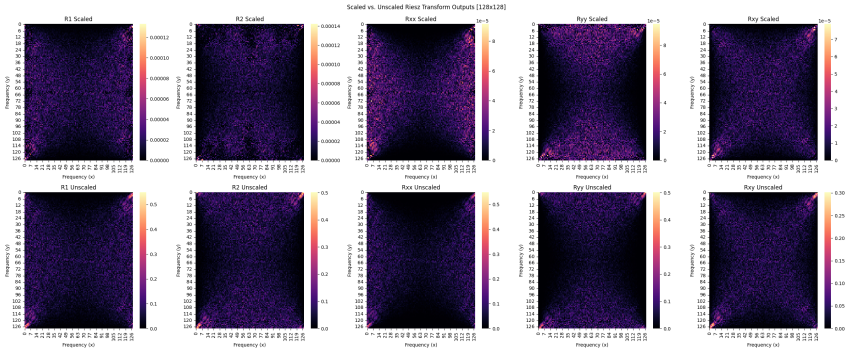

Figure 7: Spectral heat map representation of the frequency domain, with and without scaling of steerable Riesz

Assumption 1 (Spectral Property of ϕ)**.** Fourier modes $\hat{\phi}(\xi, m)_i$ are significant only for $|m| \leq 1$ i.e. $\hat{\phi}(\xi, m)_i \approx 0$ for $|m| > 1, \quad i = 1, \ldots, c_{\text{out}}$.

The equivariance constraint, defined as $\phi(x_h)(g) = \phi(x)(hg)$ for $x_h(y) = x(h^{-1}y)$, where $h \in SO(2)$, restricts $\phi(x)$ to transform predictably under rotations. For features such as gradients, this transformation manifests as a phase shift (e.g., a rotation of direction), which corresponds to Fourier modes $m = \pm 1$. Higher $|m|$ modes are associated with more intricate transformations, such as those of higher-order tensors, which are less prevalent in typical image features due to their increased rotational complexity.Hoffman (1997)

Group convolutional neural networks are specifically designed to exploit group symmetries, in this case, $SO(2)$. The feature map $\phi(x)$ is often conceptualized as an idealized output of a GCN layer, which applies filters that are equivariant under $SO(2)$. These filters typically generate low-m features, as higher-m modes necessitate more complex kernels, which are less common in the early layers of such networks. Consequently, Assumption 1 is naturally satisfied when $\phi(x)$ represents the output of such layers, capturing simple rotational patterns like edges or oriented textures.

In the context of natural images, such as photographs, the presence of edges and textures supports the validity of Assumption 1. For instance, the outline of a tree or a building rotates as a vector field, aligning with $m = \pm 1$ in the $SO(2)$-Fourier domain. These low-m modes effectively approximate the dominant features in such images, reinforcing the assumption that $\hat{\phi}$ is concentrated at $|m| \leq 1$.Van der Schaaf & van Hateren (1996)

However, Assumption 1 may not hold if $\phi(x)$ captures high-order symmetries, such as textures with rapid rotational variation. For example, starfish-like patterns exhibit complex rotational patterns that require $|m| > 1$. In such cases, the Riesz transform's concentration at $m = \pm 1$ may result in a larger distance $\|R(x) - \phi(x)\|_2$, potentially violating the inequality. This highlights the importance of aligning the feature map's spectral properties with the task and data at hand.

In summary, Assumption 1 generally holds when $\phi(x)$ is designed to capture low-order, rotation-equivariant features, such as gradients or edges, which are prevalent in natural images and early GCN layers. Its validity is less certain for intricate, high-frequency patterns, necessitating careful consideration of the feature map's design in applications where the inequality is applied.

We now analyze the theoretical advantage of incorporating the Riesz transform as a pre-processing step for equivariant networks. Consider two classifier architectures:

$$\text{Standard:} \quad f_1(x) = V(W_1 * x) \tag{1}$$
$$\text{Riesz-augmented:} \quad f_2(x) = V(W_2 * R(x)) \tag{2}$$

where W_1, W_2 are convolutional filters, $R(\cdot)$ is the Riesz transform, and V is a shared linear classifier that maps features to class logits.

For both architectures to perform equivalently, the convolutional layers must approximate the ideal feature map such that $W_1 * x \approx \phi(x)$ and $W_2 * R(x) \approx \phi(x)$. Our key insight is that the Riesz transform aligns the input representation with the symmetry structure of $\phi(x)$, reducing the complexity required in the subsequent convolutional layer.

Formally, we analyze the minimal-norm solutions,

$$W_1^* = argmin_{W \in \mathcal{W}} \|W\|_F, \;\; W_2^* = argmin_{W \in \mathcal{W}} \|W\|_F$$

If $\mathcal{R}(x)$ reduces the "distance" to $\phi(x)$, W_2 requires less magnitude what directly translates to tighter PAC-Bayesian generalization bounds, as evidenced by the coefficient $\sum_l \sum_{\psi,i,j} \frac{\|\hat{W}_l(\psi,i,j)\|_F^2}{\dim \psi} \|W_l\|_2^2$. This distance is effectively reduced, as shown by theorem 2 formulated and proved below:

For $x \in L^2(\mathbb{R}^2)$, $\mathcal{R}(x) \in L^2(\mathbb{R}^2)^2$, and $\phi(x) \in L^2(SO(2), \mathbb{R}^{c_{\text{out}}})$ satisfying Assumption 1, we have:

$$\|R(x) - \phi(x)\|_2 < \|x - \phi(x)\|_2,$$

where the norm is computed in $L^2(SO(2), \mathbb{R}^{c_{\text{out}}})$ using lifted functions $\tilde{x}$ and $\tilde{R}(x)$.

The derivation assumes that $\phi(x)$ satisfies an equivariance condition and a spectral assumption (Assumption 1), which we will specify.

Now, let $x \in L^2(\mathbb{R}^2)$ be a square-integrable, scalar-valued function over the plane, representing an input signal (e.g., an image). The L^2 norm is:

$$\|x\|^2_{L^2(\mathbb{R}^2)} = \int_{\mathbb{R}^2} |x(y)|^2 dy < \infty.$$

The feature map $\phi(x) \in L^2(SO(2), \mathbb{R}^{c_{\text{out}}})$ is a vector-valued function over the rotation group $SO(2)$, with c_{out} channels, representing a feature map (e.g., output of a neural network layer). The space $L^2(\mathrm{SO}(2), \mathbb{R}^{c_{\text{out}}})$ consists of functions $f : SO(2) \to \mathbb{R}^{c_{\text{out}}}$ with the norm:

$$\|f\|_2^2 = \sum_{i=1}^{c_{\text{out}}} \int_{SO(2)} |f_i(g)|^2 dg,$$

where dg is the Haar measure on $SO(2)$. Parametrizing $g \in \mathrm{SO}(2)$ as rotations by $\alpha \in [0, 2\pi)$, and normalizing the measure so $\int_{\mathrm{SO}(2)} dg = 2\pi$, we have:

$$\|f\|_2^2 = \sum_{i=1}^{c_{\text{out}}} \int_0^{2\pi} |f_i(\alpha)|^2 d\alpha.$$

Based on Assumption 1, Fourier components of $\phi(x)$ on $SO(2)$ are concentrated in low-frequency modes, i.e., $\hat{\phi}(m)_i \approx 0$ for $|m| > 1$, where:

$$\hat{f}(m)_i = \frac{1}{2\pi} \int_0^{2\pi} f_i(\alpha) e^{-im\alpha} d\alpha, \quad m \in \mathbb{Z}.$$

By Parseval's identityParseval (1806):

$$\|f\|_2^2 = 2\pi \sum_{i=1}^{c_{\text{out}}} \sum_{m \in \mathbb{Z}} |\hat{f}(m)_i|^2.$$

Since x and $R(x)$ are defined on $\mathbb{R}^2$, while $\phi(x)$ is on $SO(2)$, we must lift x and $R(x)$ to $L^2(SO(2), \mathbb{R}^{c_{\text{out}}})$ as $\tilde{x}$ and $\tilde{R}(x)$, respectively, to compute the norms consistently.

To lift $x : \mathbb{R}^2 \to \mathbb{R}$ to $\tilde{x} : SO(2) \to \mathbb{R}^{c_{\text{out}}}$, we use a bank of steerable filters $\{\psi_\alpha^i\}_{i=1}^{c_{\text{out}}}$, where $\psi_\alpha^i(y) = \psi^i(h_\alpha^{-1} y)$, and h_α is the rotation by α. Define:

$$\tilde{x}_i(\alpha) = \int_{\mathbb{R}^2} x(y) \psi_\alpha^i(y) dy.$$

Here, ψ^i is a base filter (e.g., a Gaussian or Gabor function), and ψ_α^i is its rotation by α. This lifting captures how x responds to rotated versions of ψ^i, producing a function on $SO(2)$ per channel.

Since $R(x) = (\mathcal{R}_1(x), \mathcal{R}_2(x))$ is a vector field, its lifting should reflect its directional nature. We can now define:

$$\tilde{R}(x)_i(\alpha) = \int_{\mathbb{R}^2} \left[\mathcal{R}_1(x)(y) \cos\alpha + \mathcal{R}_2(x)(y) \sin\alpha\right] \psi^i(y) dy,$$

where ψ^i is a scalar-valued filter. This projects $R(x)$ onto the direction $(\cos\alpha, \sin\alpha)$, aligning with the rotational structure of $SO(2)$. The choice of ψ^i (e.g., isotropic like a Gaussian) ensures the integral is well-defined in L^2.

Both $\tilde{x}$ and $\tilde{R}(x)$ are now in $L^2(\mathrm{SO}(2), \mathbb{R}^{c_{\text{out}}})$, compatible with $\phi(x)$.

Since $SO(2) \approx S^1$, functions in $L^2(\mathrm{SO}(2), \mathbb{R}^{c_{\text{out}}})$ have a Fourier series. For $f(\alpha)$, the coefficients are:

$$\hat{f}(m)_i = \frac{1}{2\pi} \int_0^{2\pi} f_i(\alpha) e^{-im\alpha} d\alpha,$$

and the norm is:

$$\|f\|_2^2 = 2\pi \sum_{i=1}^{c_{\text{out}}} \sum_{m \in \mathbb{Z}} |\hat{f}(m)_i|^2.$$

Assumption 1 implies $\phi(x)$'s energy is in $m = -1, 0, 1$, which corresponds to isotropic $(m = 0)$ and gradient-like $(|m| = 1)$ features under rotation.

Fourier Coefficients of $\tilde{x}$: for $\tilde{x}_i(\alpha) = \int x(y) \psi^i(h_\alpha^{-1} y) dy$, the coefficients depend on ψ^i. If ψ^i is isotropic (e.g., $\psi^i(y) = e^{-|y|^2}$), $\tilde{x}_i(\alpha)$ is nearly constant, with energy at $m = 0$. If ψ^i is directional (e.g., a Gabor filter), $\tilde{x}_i(\alpha)$ varies with α, spreading energy across multiple m, depending on x's content.

Fourier Coefficients of $\tilde{R}(x)$:

$$\tilde{R}(x)_i(\alpha) = \int_{\mathbb{R}^2} \left[\mathcal{R}_1(x)(y) \cos\alpha + \mathcal{R}_2(x)(y) \sin\alpha\right] \psi^i(y) dy,$$

is rewritten using $\cos\alpha = \frac{e^{i\alpha} + e^{-i\alpha}}{2}$, $\sin\alpha = \frac{e^{i\alpha} - e^{-i\alpha}}{2i}$:

$$\tilde{R}(x)_i(\alpha) = a_i e^{i\alpha} + b_i e^{-i\alpha},$$

where:

$$a_i = \frac{1}{2} \int (\mathcal{R}_1(x)(y) - i\mathcal{R}_2(x)(y)) \psi^i(y) dy, \quad b_i = \frac{1}{2} \int (\mathcal{R}_1(x)(y) + i\mathcal{R}_2(x)(y)) \psi^i(y) dy.$$

Thus:

$$\hat{\tilde{\mathcal{R}}}(x)(m)_i = a_i\delta_{m,1} + b_i\delta_{m,-1},$$

concentrating energy at $|m| = 1$, reflecting the Riesz transform's gradient-like nature.

We need to show:

$$\|\tilde{\mathcal{R}}(x) - \phi(x)\|_2^2 < \|\tilde{x} - \phi(x)\|_2^2.$$

Using Parseval's identity:

$$\|\tilde{\mathcal{R}}(x) - \phi(x)\|_2^2 = 2\pi \sum_{i=1}^{c_{\text{out}}} \sum_{m\in\mathbb{Z}} |\hat{\tilde{\mathcal{R}}}(x)(m)_i - \hat{\phi}(x)(m)_i|^2,$$

$$\|\tilde{x} - \phi(x)\|_2^2 = 2\pi \sum_{i=1}^{c_{\text{out}}} \sum_{m\in\mathbb{Z}} |\hat{\tilde{x}}(m)_i - \hat{\phi}(x)(m)_i|^2.$$

By Assumption 1, for $|m| > 1$, $\hat{\phi}(x)(m)_i \approx 0$, so:

$$|\hat{\tilde{\mathcal{R}}}(x)(m)_i - \hat{\phi}(x)(m)_i|^2 \approx |\hat{\tilde{\mathcal{R}}}(x)(m)_i|^2, \quad |\hat{\tilde{x}}(m)_i - \hat{\phi}(x)(m)_i|^2 \approx |\hat{\tilde{x}}(m)_i|^2.$$

Since $\hat{\tilde{\mathcal{R}}}(x)(m)_i \approx 0$ for $|m| > 1$, but $\hat{\tilde{x}}(m)_i$ may be non-zero (e.g., for complex x), the contribution for $|m| > 1$ is smaller for $\tilde{\mathcal{R}}(x)$.

For $|m| \leq 1$, $\tilde{\mathcal{R}}(x)$ aligns with $\phi(x)$ at $|m| = 1$, while $\tilde{x}$ may have energy misaligned with $\phi(x)$ (e.g., at $m = 0$). Thus, $|\hat{\tilde{\mathcal{R}}}(x)(m)_i - \hat{\phi}(x)(m)_i|^2$ is typically smaller than $|\hat{\tilde{x}}(m)_i - \hat{\phi}(x)(m)_i|^2$.

Since $\tilde{\mathcal{R}}(x)$'s spectral content matches $\phi(x)$'s (concentrated at $|m| = 1$), while $\tilde{x}$ has additional energy at other m, the total squared difference is smaller for $\tilde{\mathcal{R}}(x)$, proving:

$$\|\tilde{\mathcal{R}}(x) - \phi(x)\|_2 < \|\tilde{x} - \phi(x)\|_2.$$

This highlights the Riesz transform's role in aligning representations with equivariant features, enhancing network performance. The scaled features, as demonstrated in the heatmaps in Figure. 7, play a critical role in this process by accentuating the Riesz Transform outputs, ensuring that subtle frequency variations are amplified and made comparable across different components like $\mathcal{R}_1$, $\mathcal{R}_2$, $\mathcal{R}_{xx}$, $\mathcal{R}_{yy}$, and $\mathcal{R}_{xy}$. This scaling allows for better visualization and analysis of fine-grained details, which is essential for capturing object features where objects are defined by strong edges. Moreover, scaled features improve numerical stability when these representations are used in downstream algorithms, such as neural networks, by preventing large magnitude differences from skewing results. Ultimately, scaling enhances the network's ability to leverage equivariant features, leading to more robust and effective performance in frequency-domain tasks.

Here, we have finally established that enhancing the convolution layers, with our learnable steerable Riesz network has significant advantages: (a) It prevents aliasing (b) It encapsulates object features better (c) It makes existing equivariant networks generalze better.

D.1 Group-Equivariant ConvNet with Standard Riesz Transform for Scale Equivariant Networks

The baseline model integrates a standard Riesz transform within a group-equivariant CNN (G-CNN) to capture rotation-invariant features. The Riesz transform, a multi-dimensional generalization of the Hilbert transform, operates in the frequency domain to extract directional derivatives, enhancing the model's ability to detect oriented structures.

Table 5: Model classification performance

Model	Acc.%
Group CNN	82.76%
StandardCNN	70.41%
Riesz+CNN	89.78%
Riesz+GCNN	93.45%

The spatial-domain outputs are obtained via the inverse Fourier transform:

$$\mathcal{R}_1 = \mathcal{F}^{-1}(\mathcal{F}(x)\cdot\mathcal{R}_1), \quad \mathcal{R}_2 = \mathcal{F}^{-1}(\mathcal{F}(x)\cdot\mathcal{R}_2),$$

$$\mathcal{R}_{xx} = \mathcal{F}^{-1}(\mathcal{F}(x)\cdot\mathcal{R}_{xx}), \quad \mathcal{R}_{yy} = \mathcal{F}^{-1}(\mathcal{F}(x)\cdot\mathcal{R}_{yy}), \quad \mathcal{R}_{xy} = \mathcal{F}^{-1}(\mathcal{F}(x)\cdot\mathcal{R}_{xy}).$$

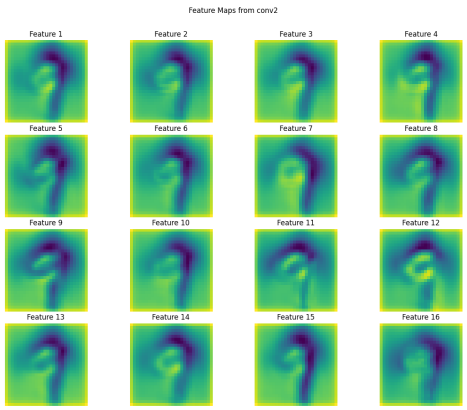

Figure 8: *
(a) Conv2 feature maps after Riesz

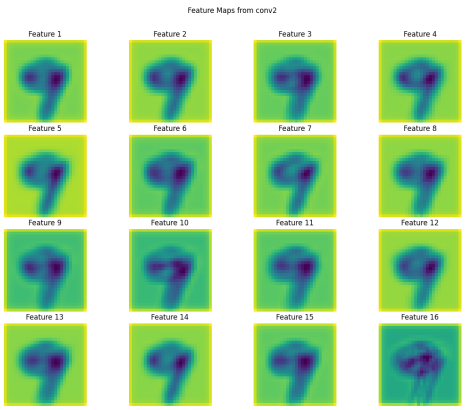

Figure 9: *
(b) Conv2 feature maps without Riesz

Figure 10: Feature maps of digit 9 with and without Riesz.

The output tensor concatenates the original input with these components: $[x, \mathcal{R}_1, \mathcal{R}_2, \mathcal{R}_{xx}, \mathcal{R}_{yy}, \mathcal{R}_{xy}]$, increasing the channel dimension to $6C$.

This is integrated into a G-CNN, where group convolutions operate over a rotation group $G = SO(2)$ discretized into R orientations. For an input $x \in \mathbb{R}^{B\times C\times R\times H\times W}$, the group convolution applies a kernel $\psi \in \mathbb{R}^{C_{\text{out}}\times C_{\text{in}}\times \mathcal{R}_{\text{in}}\times K\times K}$, transformed across rotations:

$$\psi_g = T_g\psi, \quad g \in G,$$

where T_g is the rotation operator. The output is:

$$(\psi \star x)(g, h, w) = \sum_{g'\in G} \int x(g', h', w')\psi_g(g'^{-1}g, h - h', w - w')\, dh'\, dw'.$$

The Riesz transform enhances the input representation, followed by group convolutions, max-pooling over the rotation dimension, and fully connected layers.

This baseline achieves robustness to rotations by combining the Riesz transform's directional sensitivity with group-equivariant convolutions. However, the fixed Riesz operators may not adapt to varying frequency content across images, potentially limiting performance on datasets with diverse scales and orientations.

We demonstrate in Table 5 that having a standard Riesz transform upstream to a standard convolutional neural network significantly improves scale equivariance significantly. We test on multi scaled MNIST, where we sample scale from a uniform distribution between (0.5,2.5).

Figure. 10 show the feature maps after the second group convolution layer. The Riesz transform as explained above is applied upstream to the first group convolution layer. We can clearly observe that the Riesz-enhanced convolution layers have better feature gradients and edge representation.

E From variational autoencoding to unsupervised object discovery

We use a Gaussian Mixture VAE (GMVAE) Kingma et al.; Dilokthanakul et al. (2016); Yang et al. (2019) as the base model for our VAE, which learns object representations and preserves class-conditioned glimpses. We structure the latent space with three components: z_{what} for object appearance, z_{cls} for class probabilities, and z_α for the rotation angle.

The joint distribution is factorized as:

$$p(\mathbf{z}_{\text{what}}, \mathbf{z}_{\text{cls}}, \mathbf{z}_\alpha) = p(\mathbf{z}_{\text{what}}|\mathbf{z}_{\text{cls}}, \mathbf{z}_\alpha)p(\mathbf{z}_{\text{cls}})p(\mathbf{z}_\alpha) \tag{3}$$

Appearance and Class Features: Appearance is modeled with a Gaussian Mixture Model, conditioned on class:

$$p(\mathbf{z}_{\text{what}}|\mathbf{z}_{\text{cls}} = k) = \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{4}$$

Rotational Features: Rotations are modeled using a Von Mises distribution as in Hyper-Spherical VAE Davidson et al. (2018), but here we assume a standard Normal prior:

$$p(\mathbf{z}_\alpha) = \text{VM}(\mu_\alpha, \kappa) \tag{5}$$

The rotation angle α is computed from $\mathbf{z}_\alpha$ using:

$$\alpha = \tanh(\mathbf{z}_\alpha)\pi \tag{6}$$

This angle is passed through a rotation-equivariant Spatial Transformer Network (STN).

Variational Inference: The variational posterior is factorized as:

$$q_\phi(\mathbf{z}_{\text{what}}, \mathbf{z}_{\text{cls}}, \mathbf{z}_\alpha|\mathbf{x}) = q_\phi(\mathbf{z}_{\text{what}}|\mathbf{x})q_\phi(\mathbf{z}_{\text{cls}}|\mathbf{x})q_\phi(\mathbf{z}_\alpha|\mathbf{x}) \tag{7}$$

First there is a feature extraction layer followed by a class encoder predicts z_{cls}, invariant to rotation, and z_α is predicted through an equivariant head, and then z_{what}decoded using z_{cls} and z_α with a glimpse decoder.

Handling objects at varying angles and scales pose a significant challenge for conventional models in unsupervised object discovery. While methods like data augmentation and rotational pooling address this to some extent, they fall short. Data augmentation adds computational overhead and lacks generalization to unseen rotations, while rotational pooling often sacrifices fine-grained orientation details.

Our EqSTN architecture, combined with a glimpse-based approach, resolves these issues. By focusing only on regions of interest identified by localization parameters, the model eliminates the need to process entire images or rely on extensive datasets with diverse object orientations.

The latent space in our model is disentangled to separate equivariant features, ensuring smooth handling of transformations while preserving critical object properties. A custom loss function enhances detection accuracy by enforcing structural priors, reducing false positives, and improving spatial consistency.

By embedding roto-scale equivariance at its core, our model surpasses traditional approaches, achieving efficient and reliable object detection regardless of orientation.

Object-like Latent Representation We implement an encoding model inspired by SPAIR Crawford & Pineau (2019), GMAIR Zhu et al. (2022), and SPACE Lin et al. (2020). The input image is divided into an $H \times W$ grid. For each grid cell (i, j), the encoder learns six latent variables: $z^{ij}_{\text{what}} \in \mathbb{R}^A$, $z^{ij}_{\text{depth}} \in \mathbb{R}$, $z^{ij}_{\text{pres}} \in [0, 1]$, $z^{ij}_{\text{where}} \in \mathbb{R}^4$, $z^{ij}_{\text{cls}} \in [0, 1]^C$, and $z^{ij}_\alpha \in [-\pi, \pi]$.

Here, z_{what}, z_{cls}, and z_α follow the VAE framework. z_{depth} encodes depth, z_{pres} represents object presence, and z_{where} captures spatial coordinates. The z_{pres}, z_{where}, z_{depth} and z_{cls} are rotation-invariant variables that allow stable object discovery, while z_α is rotation-equivariant, capturing angular information.

All encoder heads for predicting the latents are convolutional variational encoders, predicting μ and σ, with latent variables sampled from a normal distribution, as in VAE.

Inference and Generation Model The model processes input data x through a feature extractor of size $H \times W \times D$, from which it infers latent variables: z_{pres}, z_{where}, z_{depth}, and z_α. The input image is segmented into $H \times W$ grids, each processed in parallel to predict localization z_{where}, via z_{pres}. The model then computes α by localizing on the extracted glimpse. Further, it infers z_{cls} to class condition the object appearance encoding z_{what}, improving semantic representation learning through Gaussian mixture priors. Objects are decoded from z_{what} via a glimpse decoder and composited as in Crawford & Pineau (2019).

Background Model The background model in our model SPAGMACE is implemented as a distinct module that processes the input image to generate a set of background components, denoted as $\mathbf{z}^{\text{bg}} = (\mathbf{z}^{\text{bg}}_{1:K})$, where each component $\mathbf{z}^{\text{bg}}_k = (\mathbf{z}^{\text{m}}_k, \mathbf{z}^{\text{c}}_k)$ consists of a mask latent $\mathbf{z}^{\text{m}}_k$ and a content latent $\mathbf{z}^{\text{c}}_k$. Following GENESIS-V2Engelcke et al. (2021), the model uses a variational autoencoder (VAE)-like structure with the following steps:

An input image $\mathbf{x}$ is processed by a convolutional background image encoder to produce a feature map. This map is fed into an LSTM-based module to model sequential dependencies among background components, capturing the autoregressive relationships inspired by GENESIS-V2's stick-breaking process (SBP) for mask generation.

For each component k, the mask latent $\mathbf{z}^{\text{m}}_k$ is sampled from a Gaussian distribution $\mathcal{N}(\mu^{\text{m}}_k, \sigma^{\text{m},2}_k)$, decoded into a pixel-wise mask $\hat{\pi}_k$ using a mask decoder with sub-pixel convolution layers. The masks are normalized via the SBP to produce mixing probabilities π_k, ensuring $\sum_k \pi_k = 1$.

The content latent $\mathbf{z}^{\text{c}}_k$ is sampled from $\mathcal{N}(\mu^{\text{c}}_k, \sigma^{\text{c},2}_k)$ and decoded into an RGB appearance μ^{bg}_k using a spatial broadcast decoder. The background is reconstructed as a weighted sum of components, combined with the foreground via a pixel-wise mixture model: $p(\mathbf{x} \mid \mathbf{z}^{\text{fg}}, \mathbf{z}^{\text{bg}}) = \alpha p(\mathbf{x} \mid \mathbf{z}^{\text{fg}}) + (1 - \alpha) \sum k = 1^K \pi_k p(\mathbf{x} \mid \mathbf{z}^{\text{bg}}_k)$, where α is the foreground mixing weight.

Loss Functions The loss function consists of two main components: the reconstruction loss and the KL divergence. The reconstruction loss is computed as the average binary cross-entropy across all pixels and channels. For the KL divergence, we sum the divergences of all latent variables, following a similar approach to models like SPAIR and GMAIR. However, for z_α, which is sampled from a von Mises distribution, the KL divergence is computed as described in the previously mentioned VAE framework, ensuring proper handling of this angular latent variable.

Fake bounding-box loss In glimpse-based models, we observed a tendency to predict spurious fully-transparent bounding boxes, particularly in regions without objects. While this has no impact on reconstruction as the boxes are transparent, it is detrimental to interpretability and to evaluation measures that take into account detection accuracy. To address this issue, we introduce a penalty term $\mathcal{L}_{fake-bbox}$ that, for each cell ij, discourages predictions with both low total opacity (i.e., $a^{ij} \approx 0$) and non-zero object probability ($z^{ij}_{pres} > 0$):

$$\mathcal{L}_{fakebbox} = \sum_i \sum_j (1 - \gamma^{ij})^2 \cdot z^{ij}_{pres} \tag{8}$$

that is added to the GMAIR loss.

As described above we compare against other glimpse-based models that have an STN-based encoding architecture, as in Figure. 6(a), and in order to create variants of our model, we replace this encoding with our LeaRN-EqSTN, as in Figure. 6(b). Our architecture given sequentially predicting the transformations is very robust, and generalizes much better to unseen transformations.

E.1 Learning composite transformations

We aim at providing an accessible presentation of the impact of transformation invariance and equivariance on (training) data-efficiency. As such, we illustrate the elementary concepts with rotations (e.g. ρ_α for a rotation of angle α) but the concepts apply to other symmetry groups. To extend toward our case of interest, composite transformations, we further consider translations (e.g., $T_{\mathbf{t}}$ for a translation by a vector $\mathbf{t}$) and their composition with rotations.

Setting: We consider an arbitrary transformable input x, typically an image or feature maps. A *downstream task* is a function F learned from data that takes x as input. Such tasks are often complex and can benefit from equivariance or invariance properties with respect to x. While we discuss a fixed x, the learned model must capture canonical variations of x, encompassing variations not covered by equivariance or invariance.

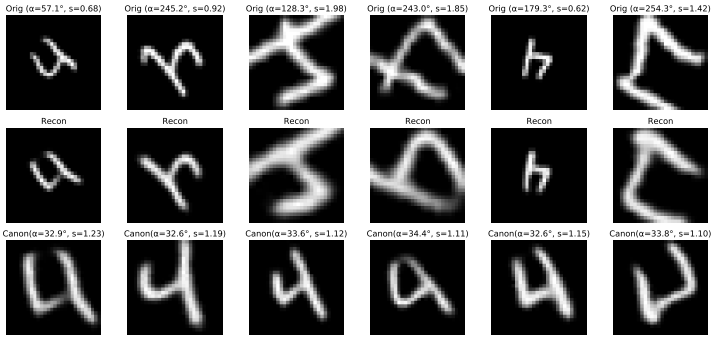

Figure 11: The canonicalisation of the digit 4. The first row is the original rotated and scaled digit, the second row is the reconstructed image, and the final row is the reconstruction canonicalised.

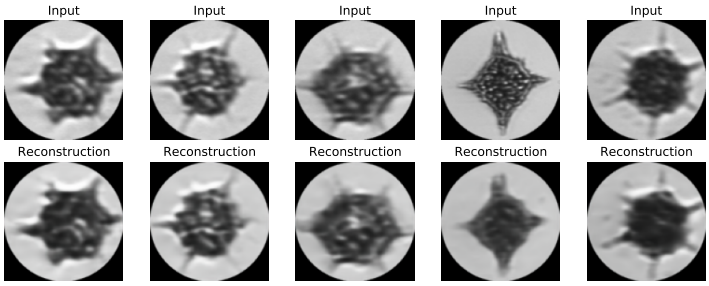

Figure 12: Qualitative visualisation of the WHO-Plankton dataset. The first row are the original samples, and the second row corresponds to the reconstructions by our LeaRN-EqSTN+GMVAE

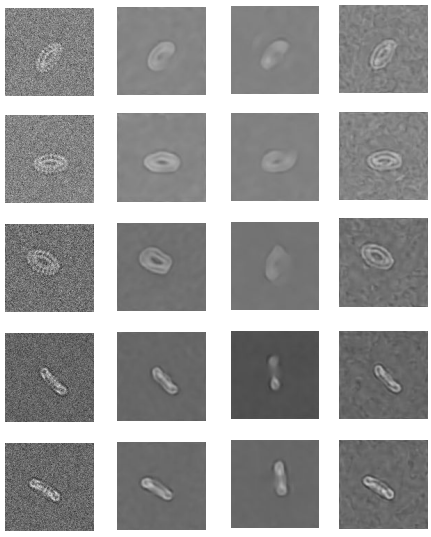

Figure 13: * Figure 14: * Figure 15: * Figure 16: * Original (a) (b) (c)

Figure 17: The reconstruction results from the benchmark models on Tomotwin Cryo-EM dataset- (a)IRL-INR (b) TARGET-VAE (c) LeaRN-EqSTN GMVAE

A function Φ is *rotation-invariant*, if for all x and α, $\Phi(\rho_\alpha x) = \Phi(x)$. It is *rotation-equivariant*, if for all x and α, $\Phi(\rho_\alpha x) = \rho'_\alpha \Phi(x)$. The transformation $\mathcal{R}'_\alpha$ is not necessarily a rotation; for instance, if $\Phi(x)$ estimates an angle, then we can have $\rho'_\alpha \Phi(x) = \alpha + \Phi(x)$.

Built-in equivariance: An ideal approach to learning a downstream task F is to make it equivariant, ensuring $F(\rho_\alpha x) = \rho'_\alpha F(x)$ for all α. This enables generalization from a single instance of x. However, achieving full rotation equivariance is computationally costly and practically challenging. Most models rely on discrete rotation groups (e.g., P_4, corresponding to four discrete rotations), limiting equivariance to a finite set of transformations. To learn F, the model must be exposed to rotations covering the quotient set ρ/P_4, meaning it should observe rotations

Learning equivariance: An alternative approach is to learn equivariance rather than enforcing it explicitly. Here, the downstream task solver F lacks built-in equivariance, but a learned transformer corrects its input. Formally, this is expressed as $G(x)F(\rho_{f(x)}x)$, where both F and the transformation predictor f are learned jointly. This approach is beneficial when the gains from correcting x outweigh the difficulty of learning f, assuming the optimization process finds an optimal f that aligns inputs correctly.

For any x and $\alpha \in [0, 2\pi]$, the transformed output follows $G(\rho_\alpha x) = F(\rho_{f(\mathcal{R}_\alpha x)}$

Decomposing composite transformations: We now focus on the case of compositions of a translation with a rotation, i.e. $T_t\mathcal{R}_\alpha$. We consider a two-step prediction and correction of the transformation. More precisely, the full process computes $H(x)F(\rho_{f^\rho(y)}y)$ with $y(x) = T_{f^T(x)}x$, where f^ρ, f^T and F are to be learned.

We rewrite, for any x, any translation t and rotation $\alpha \in [0, 2\pi]$ the output of H when applied to the transformed version of x, $T_t\rho_\alpha x$. We have $y(T_t\rho_\alpha x) = T_{f^T(T_t\rho_\alpha x)}T_t\rho_\alpha x$. In case of an optimal f^T, the translation component gets removed, i.e. $y(T_t\rho_\alpha x) = \rho_\alpha x$ The treatment of f^R in $H(T_t\rho_\alpha x)$ then boils down to the previous case of G, concluding on the fact that all factors of variations have been removed for F.

We can achieve maximal data efficiency by leveraging this decomposition. By using a f^T that is translation equivariant we get $T_{f^T(T_t\rho_\alpha x)} = T_{-t+f^T(\rho_\alpha x)}$. By using a f^T that is further rotation *invariant* we get $T_{-t+f^T(\rho_\alpha x)} = T_{-t+f^T(x)}$. Finally, using a rotation equivariant f^R for the second level of transformation. The only remaining variations necessary to learn are the fine translation for f^T and the continuous rotations for f^ρ, independently.

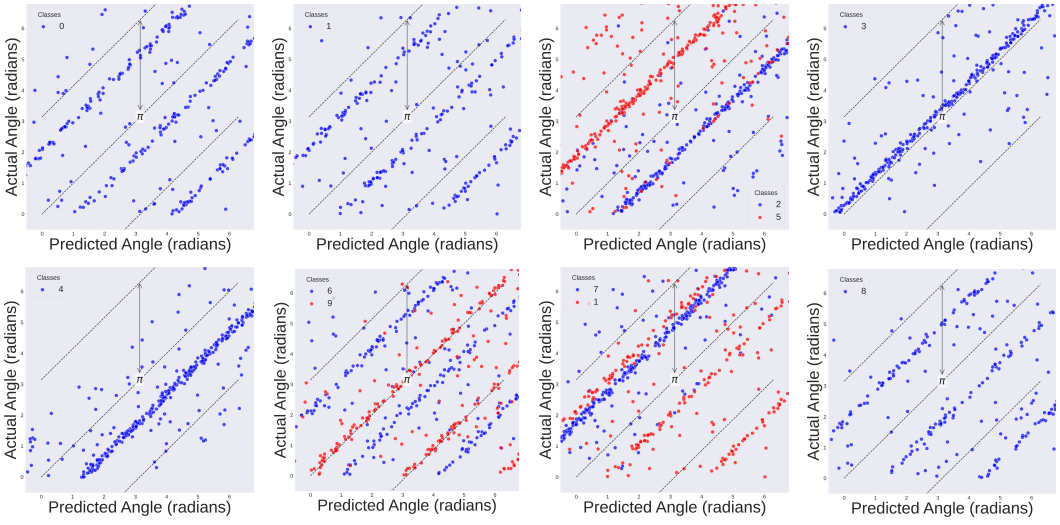

Figure 18: Scatter plots demonstrating the correlation between predicted and actual angles of orientation for different MNIST digit classes.

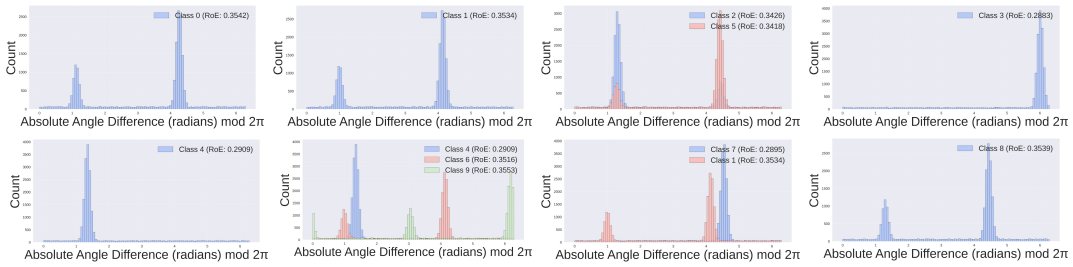

Figure 19: Histograms demonstrating the absolute angle difference modulo 2π between predicted and actual angles of orientation for different MNIST digit classes with the corresponding Rotation-offset Entropy(RoE)

F Qualitative and Quantitative Results

In this section we provide qualitative and quantitative results from our experiments. Figure 17 compares the reconstructions of all the models on Tomotwin Cryo-EM benchmark dataset on low data, as described in our main paper. We demonstrate our results on the WHO-Plankton dataset in Figure 12.

To finely quantify the quality of angle/scale capture, we demonstrate, in Figure 11, the effectiveness of how our model effectively learns scale and rotation, and successfully learns a canonical representation of the object, here we demonstrate it for the digit 4. We provide more angle-recovery results in Figures 18 and 19.

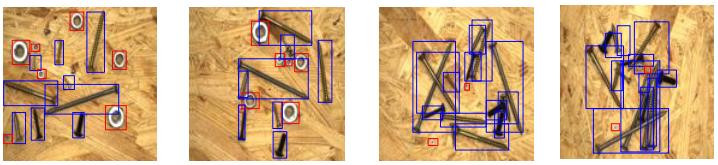

Figure 20: *
(a)

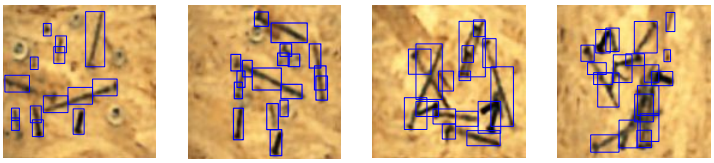

Figure 21: *
(b)

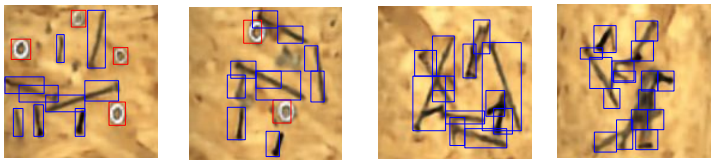

Figure 22: *
(c)

Figure 23: Qualitative visualizations of the object discovery (i.e., no boxes or labels provided) task on Low Data setting for MVTec Screws dataset-(a) Original Images (b) GNM (c) LeaRN-EqSTN+SPAGMACE

For our object discovery task we show the qualitative visualization of our model LeaRN-EqSTN-SPGAMCE and GNM model for the MVTec Screw dataset in Figure 23.

We also provide further ablation studies for our variational autoencoding setting in Table. 6, to demonstrate the effectiveness of each of our component. We provide training time comparison in Table 8. Finally, we give an empirical analysis of the computation complexity of all the models, in Table 7. We use the python package calflops to get the FLOPs analysis.

	More Data			Low Data		
	NMI	ARI	SSIM	NMI	ARI	SSIM
SR-MNIST						
STN-VAE	0.65	0.57	0.78	0.59	0.51	0.75
STN-GMVAE	0.69	0.61	0.79	0.6	0.52	0.75
EqSTN+GMVAE	0.81	0.70	0.92	0.68	0.59	0.81
TARGET-VAE P_8	0.78	0.65	0.86	0.63	0.51	0.77
TARGET-VAE P_{16}	0.78	0.67	0.88	0.65	0.52	0.8
CODAE	0.76	0.66	0.83	0.62	0.52	0.76
IRL-INR	0.83	0.78	0.94	0.65	0.51	0.69
LeaRN-EqSTN+GMVAE	**0.88**	**0.80**	**0.95**	**0.78**	**0.62**	**0.83**
LeaRN+TARGET-VAE P_{16}	0.82	0.72	0.93	0.69	0.60	0.82

Table 6: Ablation Study on clustering and reconstruction metrics across SR-MNIST.

Computational Time: All the experiments were conducted on a system equipped with 48GB of RAM and an NVIDIA RTX A5000 GPU. The runtime for our model LeREqSTN+SPAGMACE and for GNM are as follows

F.1 Results Analysis

From all the empirical evaluation, we observe that our LeaRN-EqSTN enhanced architectures outperform every SOTA model in both the unsupervised tasks. We also demonstrate through our ablation studies how each component effectively improves the models. The complexity analysis in Table. 7 also shows that our model is light in terms of parameter count and very efficient in terms of computation time.

The Rotation-offset entropy metric in Figure. 19 shows how well it encapsulates the rotation offsets for every MNIST class. It is interesting to also observe the peaks in the histogram which correspond to the number of lines of rotation symmetry the object has, which is also mirrored in the Figure. 18 scatter plot.

Finally, we observe that digit classes with similar orientations are clustered together, demonstrating that the latent space effectively captures equivariant features. This clustering is significant, as it highlights the latent space's ability to encode orientation-based similarities in an unsupervised manner, without impacting task performance.

Model	Parameters	FLOPs
LeaRN-EqSTN+GMVAE	1.1M	16 GFLOPs
TARGET-VAE P_8	0.9M	25 GFLOPs
TARGET-VAE P_16	1.4M	29 GFLOPs
IRL-INR	80M	37 GFLOPs

Table 7: Complexity Analysis

Model	MTR-MNIST	MVTec Screws	MVTec D2S
LeaRN-EqSTN+GMAIR	512 mins	794 mins	1257 mins
GNM	715 mins	886mins	1648 mins

Table 8: Training time (in minutes) across different datasets