# Distribution Modeling and GenAI-Assisted Projection for SAR Incremental Learning

Heqing Huang[1, 2]
huangheqing@buaa.edu.cn
Fei Gao[1]
feigao2000@163.com
Vahid Akbari[2, *]
vahid.akbari@stir.ac.uk

[1] The School of Electronic and Information Engineering, Beihang University, Beijing, China

[2] Departmen of Computing Science and Mathematics, University of Stirling, Stirling, UK

**Abstract**

In class incremental learning for synthetic aperture radar (SAR) imagery, models must acquire new categories while retaining knowledge of previous ones. Generative replay can mitigate forgetting by synthesizing old class samples. However, vanilla generative networks, such as variational autoencoder (VAE), prioritize pixel level reconstruction and do not inherently enforce class separability, which may not be optimal for incremental recognition. To address this issue, we analyze the distribution of the dataset used. The class-wise latent distributions are modeled via flow-based density estimation, enabling the generation of representative, in-distribution exemplars. Then we combine with current-task data, the exemplars support a feature projection between old and new latent spaces, from which a numerically optimized closed-form classifier is reconstructed. This dual use of learned distributions both constrains generative replay to in-distribution regions and calibrates decision boundaries to reduce drift. Experiments on SAR benchmarks demonstrate that our approach achieves state-of-the-art accuracy while maintaining a superior stability and plasticity trade-off.

## 1 Introduction

Synthetic aperture radar (SAR) imagery is vital for earth observation, environmental monitoring, and disaster management [3, 4, 17, 28]. Among them, the SAR Automatic Target Recognition (ATR) system is the key to realize the above applications, which is designed to automate the precise identification of various types of targets from SAR images. In operational SAR, the steady expansion of mission scope and the introduction of novel target types pose a core challenge: the tracking system must accommodate new classes while retaining performance on known ones. Retraining the entire model for new classes is costly and often impossible without historical data. Fine tuning only on new classes causes catastrophic forgetting, sharply degrading performance on previous ones. Class incremental learning (CIL) [19, 25] meets this need by enabling models to acquire knowledge from disjoint class

sets while retaining prior information. In such SAR applications, exemplars from previous classes may be absent or limited, as privacy regulations and proprietary restrictions often prevent their storage or sharing, necessitating alternative strategies to preserve prior knowledge.

Generative Artificial Intelligence (GenAI) models, such as variational autoencoders (VAEs) [15] and generative adversarial networks (GANs) [6], have been applied to CIL to generate pseudo-samples of previous classes [16, 18, 29, 34], reducing reliance on large exemplar sets. However, in the SAR domain, effective generative replay is challenged by high intra-class variability, speckle noise, and scarce labeled data. Furthermore, generative models typically optimize their latent spaces for high-fidelity reconstruction, an objective that is misaligned with the demands of discriminative ATR tasks. This focus on reconstruction is further hampered by the standard Gaussian prior, which encourages smooth, continuous structures that are suboptimal for forming distinct decision boundaries [27]. While such frameworks [18, 23] excel at generating in-distribution samples by concentrating on high-density regions, their effectiveness hinges on the assumption of a Gaussian data distribution. This reliance, however, is often invalid in complex SAR domains, where data characteristics differ significantly from natural optical imagery.

A more fundamental challenge is semantic drift, which occurs when the latent spaces of old and new models become misaligned. This issue is especially acute in SAR ATR, where class separability is inherently limited by complex scattering behaviors and diverse imaging geometries. Recent analytic continual learning methods (ACL) [8, 42] attempt to mitigate this drift by learning a linear projection matrix that maps features from the old model's space to the new one. However, a key limitation of this approach is its dependence on new task data to indirectly estimate the feature drift of old classes. The reliability of this estimation diminishes as the semantic gap between old and new tasks widens, leading to increased bias and further compromising the stability of the learned model.

In this study, we tackle catastrophic forgetting in SAR CIL while preserving stable decision boundaries across incremental steps. To better align generative replay with SAR data characteristics, we replace the conventional unimodal Gaussian prior in VAEs with a more expressive prior based on normalizing flows (NF) [20, 26]. Normalizing flows learn a complex, multi-modal probability distribution by transforming a simple base distribution through a series of invertible mappings, which allows them to capture the heavy-tailed statistics and complex dependencies inherent in SAR backscatter data. Consequently, this enables the generation of high-fidelity old class samples that preserve the fine-grained scattering patterns essential for discrimination.

Leveraging the generated old-class samples from this improved prior, we perform feature space alignment by learning a projection matrix that maps features using a mixture of generated old and real new data. This aligned feature space then enables the closed-form reconstruction of a ridge regression classifier (RRC) [42] based on calibrated statistical summaries (covariances and prototypes) from both data sources. The resulting classifier is less biased, more robust to semantic drift, and achieves a superior stability-plasticity trade-off.

Our main contributions are: 1) Our analysis of SAR image statistics reveals that its heavy-tailed distribution is poorly modeled by the standard Gaussian prior in VAEs. Therefore, we correct the learned latent distribution by introducing a flexible, distribution-aware prior based on normalizing flows, which is explicitly trained to fit the characteristic statistics of SAR data. 2) Our approach skillfully combines data-driven generative modeling with model-driven ACL. The generative model produces calibration references for old classes. And the RRC leverages these references to achieve an efficient and unbiased model update.

# 2 Related work

CIL addresses the challenge of acquiring knowledge from a sequence of disjoint class sets while maintaining performance on previously learned classes. Existing approaches can be broadly grouped into rehearsal-based [1, 12, 40], regularization-based [31, 32, 39], and parameter-isolation-based strategies [30, 33]. Generative replay [13, 24, 41] mitigates forgetting by replacing or augmenting stored exemplars with pseudo-samples of previous classes. These samples are synthesized by generative models trained to capture the data distribution of old classes. Recent methods often use diffusion models [5, 22] or advanced GANs [7, 9]. However, such models are not suitable for SAR imagery, where labeled data is scarce and large transformer-based generators cannot be trained effectively.

Another line of work closely related to ours is ACL [8, 36, 38], which updates the classifier in a closed-form manner, thus avoiding regularization-based optimization for old classes. These methods typically estimate class prototypes, covariances, and projection matrices to align the feature spaces of old and new models. This alignment enables stable classifier reconstruction without overwriting historical decision boundaries. When combined with our generative replay, ACL further exploits synthetic old class features for distribution-aware calibration. This effectively mitigates semantic drift while maintaining a favorable stability–plasticity trade-off.

# 3 Method

## 3.1 SAR Category Incremental Learning Foundation Framework

For our SAR image CIL experiments, we adopt the incremental classifier and representation learning (iCaRL) framework [25] as the baseline. iCaRL alleviates catastrophic forgetting by transferring knowledge from the previous model to the current one for old classes through a memory-constrained exemplar set. This is achieved via a soft-label knowledge distillation loss with temperature scaling, which aligns the current model's logits for old classes to those of the previous model using the Kullback–Leibler (KL) divergence. The distillation loss is formulated as follows:

$$L_{\text{distill}} = \tau^2 \cdot \text{KL}\big(\text{Softmax}(z^{\text{teacher}}/\tau) \parallel \text{Softmax}(z^{\text{student}}/\tau)\big) \tag{1}$$

where $z^{\text{teacher}}$ and $z^{\text{student}}$ denote the logits from the previous and current models for old classes, and $\tau$ represents the temperature parameter controlling the softness of the probability distribution.

In iCaRL, the exemplar memory is typically constrained to a fixed budget, which may be insufficient in practice. To address this, we integrate VAE [23] to enable generative replay, synthesizing pseudo-samples of previously learned classes to augment limited exemplar sets without incurring additional memory overhead. Finally, the baseline employs a shared encoder, decoder, and linear classifier to jointly optimize reconstruction, classification, and KL divergence terms. For variational inference with this model, the sum over all elements in the dataset $n \in \mathcal{D}$ in the following lower bound is optimized.

$$\mathcal{L}(x,y) = \mathbb{E}_{q_\theta(z|x)}\left[\log p_\phi(x|z) + \log p_\xi(y|z)\right] - \beta\,\text{KL}\left(q_\theta(z|x) \parallel p(z)\right) \tag{2}$$

where $x$ and $y$ are the image inputs and their labels. $z$ represents the latent variable, and $\beta$ balances the reconstruction classification terms and latent regularization. Here, $p_\phi(x|z)$

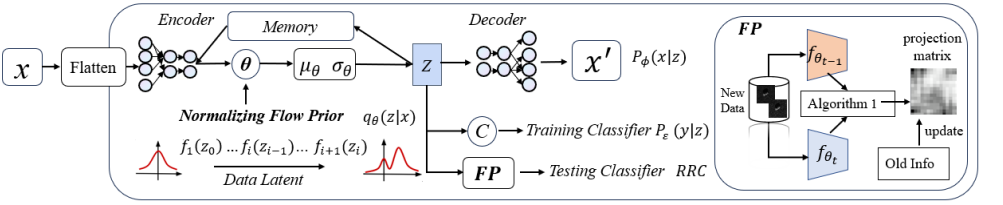


Figure 1: Our framework. An encoder maps the input to distribution parameters $(\mu, \sigma)$, from which a latent code $z$ is sampled. A decoder then reconstructs the input as $\hat{x}$. The feature projection (FP) head, which stabilizes the encoder's features for downstream tasks and loss computation. The fully connected classifier (C) is only used to update the model parameters. Memory buffer stores exemplars for rehearsal and distillation; it is not used during inference.

denotes the reconstruction likelihood parameterized by the decoder with parameters $\phi$, capturing the probability of reconstructing the input $x$ given the latent variable $z$. The term $p_\xi(y|z)$ represents the classification likelihood parameterized by the linear classifier with parameters $\xi$, modeling the probability of predicting the label $y$ conditioned on $z$. The encoder, parameterized by $\theta$, approximates the variational posterior $q_\theta(z|x)$, while $p(z)$ is the prior distribution over the latent space. In the subsequent work, we use distributed modeling and feature alignment strategies. Our overall framework diagram is shown in Figure 1.

## 3.2 Distribution-Aware Latent Modeling with Normalizing Flow Prior

### 3.2.1 Normalizing Flow Prior for SAR Generative Replay

Traditional VAEs assume latent representations follow a multivariate Gaussian distribution, enabling efficient reparameterization and sampling. This assumption, however, conflicts with the intrinsic statistics of SAR imagery. SAR's coherent imaging mechanism produces heavy-tailed and skewed amplitude statistics due to multiplicative speckle. Prior studies confirm this mismatch: Mahapatra et al. [21] showed that Weibull distributions better fit low-heterogeneity regions, while log-normal distributions are more suited for heterogeneous clutter. Xie et al. [35] found that MSTAR clutter regions are better modeled by K or Weibull distributions, whereas target regions exhibit log-normal-like characteristics. Our statistical analysis (Fig. 2) fits log-normal, gamma, and normal distributions to pixel intensity histograms of MSTAR dataset. The results show that log-normal and gamma provide a much better fit to the heavy-tailed SAR amplitude statistics than Gaussian, underscoring the need for more expressive priors in variational frameworks.

In order to solve the distribution mismatch problem, we add the planar NF after the standard Gaussian prior in the VAE [20, 26]. NF transforms samples from a simple Gaussian distribution into a more expressive one via a sequence of invertible, differentiable mappings. This design enables the latent space to more accurately model the heavy-tailed and potentially multi-modal statistics of SAR backscatter. As a result, the generative module can capture fine-grained scattering variations. This capability ultimately allows for the synthesis of more representative pseudo-samples for old classes. Formally, given encoder outputs $\mu \in \mathbb{R}^{B\times d}$ and $\sigma \in \mathbb{R}^{B\times d}$ for batch size $B$ and latent dimension $d$, we first sample from the approximate posterior: $z = \mu + \varepsilon \odot \sigma, \quad \varepsilon \sim \mathcal{N}(0, I)$, where $\odot$ denotes element-wise

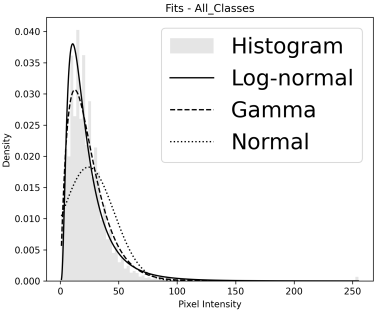


Figure 2: MSTAR pixel intensity distribution fitting. The log-normal distribution align better with the heavy-tailed nature of SAR clutter compared to the Gaussian distribution.

multiplication. The base latent variable $z_0 = z$ is then transformed through $K$ planar flows:

$$z_k = f_k \circ \cdots \circ f_1(z_0), \quad f_k(z) = z + u_k\, h(w_k^T z + b_k) \tag{3}$$

with learnable parameters $u_k, w_k, b_k$ and a non-linear activation $h(\cdot)$. The log-determinant of the Jacobian for each flow, $\log \left| \det \frac{\partial f_k}{\partial z} \right|$, is computed and accumulated to account for volume changes under transformation. Under this flow transformation, the density of $z_K$ is given by:

$$\log p(z_K) = \log p(u) - \sum_{k=1}^{K} \log \left| \det \frac{\partial f_k}{\partial z} \right| \tag{4}$$

where $p(u)$ is the base Gaussian prior before flows. Using Eq.(2), this term modifies the Kullback-Leibler divergence (KLD) in the VAE objective to:

$$\mathcal{L}_{\text{KLD}} = \text{KL}(q(z|x) \,\|\, p(z)) = \mathbb{E}_{q(z|x)} \left[ \log q(z|x) - \log p(u) - \sum_{k=1}^{K} \log \left| \det \frac{\partial f_k}{\partial z} \right| \right] \tag{5}$$

The NF prior enhances our VAE to generate high-fidelity synthetic SAR samples for generative replay in CIL. At each incremental step $t$, real SAR samples are encoded into distribution parameters $\mu_{\text{real}}$ and $\sigma_{\text{real}}$. A latent code $z$ is sampled using the reparameterization trick: $z = \mu_{\text{real}} + \varepsilon \cdot \sigma_{\text{real}}, \varepsilon \sim \mathcal{N}(0, I)$. The NF learns a flexible transformation that effectively captures the heavy-tailed and multi-modal statistics of SAR data, overcoming the limitations of a standard Gaussian prior. The model is trained on both current task data and the generated pseudo-samples, which enhances knowledge preservation from previous tasks.

### 3.2.2 Decoupled Encoder-Decoder Training

Current VAE models often exhibit unstable training dynamics on complex SAR data. This instability stems largely from conflicting objectives during joint encoder-decoder optimization. The encoder aims to map inputs into a well-structured latent distribution, while the decoder must reconstruct detailed imagery from this distribution. Such coupled optimization can lead to oscillatory behavior and convergence issues. This challenge is particularly pronounced with non-Gaussian, heavy-tailed SAR statistics. To enhance training stability and model performance, we introduce a decoupled training strategy [□]. This approach separates the learning of the encoder and decoder into distinct phases, thereby reducing gradient

competition and promoting more robust latent space formation. In this scheme, the overall loss $\mathcal{L}_t$ is split into encoder and decoder objectives that guide each component's update, as described below.

Encoder Update: The encoder is updated by jointly optimizing classification, reconstruction, and NF-based KL divergence terms while incorporating distillation to enhance feature discriminability and knowledge retention. Specifically, the encoder loss is defined as:

$$\mathcal{L}_{\text{enc}} = \mathcal{L}_{\text{cls}} + \alpha\,\mathcal{L}_{\text{rec}} + \beta\left(\mathcal{L}_{\text{KLD}}^{\text{real}} + \frac{\gamma}{2}\left(\mathcal{L}_{\text{KLD}}^{\text{rec}} + \mathcal{L}_{\text{KLD}}^{\text{fake}}\right)\right) + \lambda_d\,\mathcal{L}_{\text{distill}}. \tag{6}$$

where $\alpha, \beta, \gamma, \lambda_d$ are hyperparameters balancing each term. The KLD terms in Eq. (6) collectively enforce a structured latent space: $\mathcal{L}_{\text{KLD}}^{\text{real}}$ ensures the latent distribution of real inputs conforms to the prio. $\mathcal{L}_{\text{KLD}}^{\text{rec}}$ regularizes the latent codes of reconstructed samples toward the same prior to maintain generation stability. Furthermore, $\mathcal{L}_{\text{KLD}}^{\text{fake}}$ explicitly aligns the latent distribution of synthetic pseudo-samples with the prior, which is critical for mitigating catastrophic forgetting by preventing the representation drift of old-class data during incremental learning. The classification loss $\mathcal{L}_{\text{cls}} = -\sum_i \log p_{y_i}$, where $p_{y_i}$ denotes the predicted probability of the ground-truth class $y_i$. $L_{\text{rec}}$ denotes the reconstruction loss, which is implemented using the binary cross-entropy loss to measure the similarity between the input image and the reconstructed output generated by the decoder. It is defined as:

$$\mathcal{L}_{\text{rec}} = -\sum_i \Big[x_i \log \sigma(\hat{x}_i) + (1 - x_i)\log\big(1 - \sigma(\hat{x}_i)\big)\Big] \tag{7}$$

Decoder Update: After updating the encoder, the decoder is updated to ensure that reconstructed samples and pseudo samples align with the NF prior while maintaining high-fidelity reconstruction required for effective generative replay. During this phase, latent codes sampled from real samples ($z$) and from the prior ($z_p$) are decoded to generate $\hat{x}$ and $\hat{x}_p$. The reconstructed samples are re-encoded to compute the NF-based KL divergences. The decoder parameters are then updated by minimizing the following objective:

$$\mathcal{L}_{\text{dec}} = 0.5\gamma\big(\mathcal{L}_{\text{KLD}}^{\text{rec}} + \mathcal{L}_{\text{KLD}}^{\text{fake}}\big) + \alpha\mathcal{L}_{\text{rec}} \tag{8}$$

## 3.3 Generative Sample Augmented ACL for Classifier Reconstruction

To mitigate semantic drift between consecutive tasks, we estimate a mapping from the old model's feature space to the current one via a cross-covariance transformation, following the dual projection idea in ACL. Unlike the original ACL, which uses only real current task samples, we augment the estimation set with generated old class samples from the VAE backbone. The combined set $\mathcal{D}_{\text{gen+real}}$ thus contains real new class data and generated old class data, ensuring the projection captures relationships across both old and new classes. The specific process of our method is shown in Algorithm 1.

Let $\mathbf{F}_{\text{old}}, \mathbf{F}_{\text{new}} \in \mathbb{R}^{N\times d}$ be features from the old encoder $f_{\theta_{t-1}}$ and the new encoder $f_{\theta_t}$ on the same inputs from $\mathcal{D}_{\text{gen+real}}$. The projection matrix is computed as $\Delta = \mathbf{C}_{\text{old}}^{-1}\mathbf{Q}_{\text{old}\to\text{new}}$, where $\mathbf{C}_{\text{old}} = \mathbf{F}_{\text{old}}^{\top}\mathbf{F}_{\text{old}}$ is the old feature covariance. $\mathbf{Q}_{\text{old}\to\text{new}} = \mathbf{F}_{\text{old}}^{\top}\mathbf{F}_{\text{new}}$ is the cross-covariance, and the inversion of $\mathbf{C}_{\text{old}}$ is regularized for numerical stability. This $\Delta$ maps old features into the new latent space for category-specific calibration.

After projection, we reconstruct the classifier in closed form via RRC, enabling regularized weight estimation for all classes simultaneously. Class prototypes $\mu_{i,c}$ and uncentered

**Algorithm 1** GenAI-Assisted Projection Classifier Reconstruction

**Require:** Old model $f_{\text{old}}$, current model $f$, dataset $\mathcal{D}_{\text{gen+real}}$, ridge factor $\gamma$
**Ensure:** Updated projector $\mathcal{P}$, classifier $W$

1: Extract old/new features for all samples in $\mathcal{D}_{\text{gen+real}}$
2: Accumulate covariances $\Sigma_{\text{old}}, \Sigma_{\text{new}}$ and cross-correlation $Q_{\text{old}\to\text{new}}$
3: Compute projection $\Delta = \Sigma_{\text{old}}^{-1} Q_{\text{old}\to\text{new}}$; set $\mathcal{P}$ weight $\leftarrow \Delta^{\top}$
4: Calibrate old class prototypes and covariances via $\Delta$
5: Compute new class statistics directly from $\mathcal{D}_{\text{gen+real}}$
6: Reconstruct classifier via ridge regression; apply $L_2$ normalization

covariances $\Phi_{i,c}$ for all old and new classes are computed from $\mathcal{D}_{\text{gen+real}}$, with old class statistics first calibrated into the new space via $\Delta$. The reconstructed classifier weights are given by:

$$\hat{W}_t = \left( \sum_{i=1}^{t} \sum_{c \in C_i} \Phi_{i,c} + \gamma I \right)^{-1} \left( \sum_{i=1}^{t} \sum_{c \in C_i} H_{i,c} \right) \tag{9}$$

The reconstructed classifier weights at task $t$ are computed to consolidate knowledge from all observed categories. $C_i$ denote the set of class labels introduced in the $i$-th task, and $\Phi_{i,c}$ be the uncentered covariance matrix of the deep features for class $c$ in task $i$, where $\phi(x_{c,n})$ is the feature vector of the $n$-th sample from class $c$. Similarly, $H_{i,c}$ aggregates the feature-label correlations, with $y_c$ being the one-hot vector of class $c$.

Unlike methods that infer semantic drift only indirectly from new task data, our feature projection process achieves direct alignment. We form a hybrid dataset by combining the generated old class samples with the real new class samples. By solving a uniform linear projection matrix on this hybrid set, we are able to simultaneously and directly capture the drift of the old class features themselves as well as the changes introduced by the new class features. This approach effectively transforms the drift estimation problem from an ill-posed inference problem to a well-posed regression problem. Ultimately, a more robust and accurate projection transformation is obtained.

# 4 Experiments

## 4.1 Experiment Setting

**Implementation Details.** We use Adam with a learning rate of $1 \times 10^{-3}$ and train each incremental step for 100 epochs (batch size 32). We finally set $\alpha = 1.0$, $\beta = 0.5$, $\gamma = 1.0$, $\lambda_d = 0.1$, and $\lambda_c = 0.05$. We selected these values using a small held-out validation split around each coefficient, observing moderate sensitivity—especially in the trade-off between reconstruction and distillation—followed by fixing the same configuration for all reported runs. A memory buffer of 200 exemplars is maintained. Performance is evaluated by classification accuracy $\text{Acc} = (N_{\text{correct}}/N_{\text{total}}) \times 100\%$.

**Model structure.** Our model is an MLP-based conditional VAE with a normalizing-flow prior. The encoder processes the flattened SAR image concatenated with a one-hot class vector through two fully connected layers and two residual blocks to produce latent parameters $(\mu, \sigma)$. Latent codes are sampled via reparameterization and refined by three planar flows. The decoder takes the latent code and class vector to reconstruct the SAR

image. We also instantiate a CNN-based conditional VAE. A ResNet-10 encoder maps each SAR chip to a global feature, from which two linear heads produce $(\mu, \sigma)$. The latent code is first projected by a linear layer. Then upsampled through a stack of transposed-convolution and convolution blocks, yielding the reconstructed image.

**Datasets and protocols.** We evaluate on two SAR benchmarks. MSTAR [14] is an X-band ATR dataset of military ground targets acquired at 17° and 15° depression angles with notable intra-class variability and heavy-tailed speckle noise. SHIP dataset (FUSAR-SHIP [10]) provides high-resolution GF-3 ship chips over diverse marine backgrounds. We adopt two evaluation settings, "S1–S5" and "S1–S2" denote the evaluation stage indices: (A) MSTAR (5-stage), each stage introduces two new classes in order: S1={ZIL131, D7}, S2={ZSU_23_4, BTR70}, S3={T72, BMP2}, S4={BRDM_2, T62}, S5={BTR60, 2S1}. And (B) MSTAR→FUSAR-SHIP (2-stage), S1 uses all ten MSTAR classes and S2 introduces four ship classes from SHIP: {BulkCarrier, CargoShip, Fishing, Tanker}.

## 4.2 Quantitative Comparison with Existing Methods

Table 1 reports the classification accuracy under different experiment settings. Comparing our method with representative rehearsal [25], generative replay [23], and projection-based approaches [8]. In the 5-stage setting, where catastrophic forgetting is more pronounced, our method achieves consistently higher accuracy across all incremental steps. Notably, it retains 91.09% accuracy at the final step (S5), outperforming iCaRL by +28.2%, VAE by +19.6%, and DPCR by +13%, demonstrating superior resistance to knowledge degradation. Compared with recent non-generative methods such as DER [37] and PODNet [4], our approach still delivers clear gains of +4.6% and +15.3% at S5, respectively.

In the challenging MSTAR→SHIP cross-domain CIL setting, our method demonstrates a significant advantage in mitigating catastrophic forgetting under domain shift. It achieves a final accuracy of 92.5% at Stage 2. This result surpasses all compared state-of-the-art methods by a considerable margin: it outperforms iCaRL by +19.3%, the generative VAE baseline by +25.8%, DPCR by +8.4%, DER by +2.6%, and PODNet by +10.1%. The pronounced performance gap, particularly over other generative and replay-based techniques, highlights the effectiveness of our flow-enhanced generative replay and distribution-aware classifier reconstruction in preserving knowledge across domains.

## 4.3 Ablation and Component Analysis

To assess the contribution of each component in our framework, we conduct an ablation study under the 5-stage CIL setting, reporting the final-step accuracy in Table 2. Starting from the full model (memory + NF + RRC) with 91.1% accuracy, removing the NF module and reverting to a Gaussian prior leads to a significant drop of -21.5%. Introducing a small exemplar memory partially mitigates this degradation +(10.1%), while adding NF further improves performance to 83.4% by producing distribution-consistent generative samples.

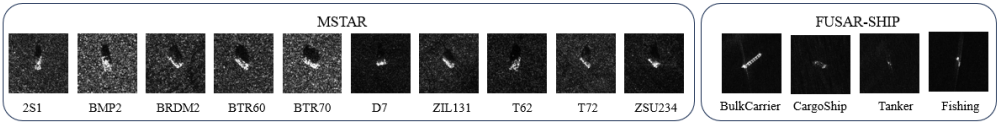


Figure 3: Dataset image examples.

Table 1: Accuracy (%) on MSTAR (5-stage) and MSTAR→SHIP (2-stage) CIL settings. The "Upper bound" is an upper limit on the performance of the classification model, denotes oracle joint training on all classes simultaneously. Bold = best among CIL methods.

| | MSTAR | | | | | MSTAR→SHIP | |
|---|---|---|---|---|---|---|---|
| Method | S1 | S2 | S3 | S4 | S5 | S1 | S2 |
| Upper Bound | 99.2 | 98.2 | 98.0 | 97.6 | 97.1 | 98.4 | 95.9 |
| iCaRL [25] | 99.7 | 94.4 | 88.3 | 72.4 | 62.9 | 95.8 | 73.2 |
| VAE [23] | 98.7 | 95.8 | 91.7 | 81.2 | 71.5 | 96.2 | 66.7 |
| DPCR [8] | 99.0 | 96.7 | 92.3 | 85.4 | 78.1 | 96.9 | 84.1 |
| DER [37] | 98.9 | 96.1 | 93.8 | 90.7 | 86.5 | 97.1 | 89.9 |
| PODNet [2] | 98.8 | 95.9 | 91.6 | 84.1 | 75.8 | 96.5 | 82.4 |
| Ours | **99.1** | **98.5** | **98.0** | **94.8** | **91.1** | **98.2** | **92.5** |

Replacing NF with RRC yields an even larger gain (87.1%), as the closed-form reconstruction better preserves decision boundaries. The best performance is achieved when memory, NF, and RRC are combined, confirming their complementary effects.

The analysis of planar flow depth $K$ indicates diminishing returns beyond an optimal point. Performance peaks at $K = 3$ (91.1%). A shallower flow ($K = 1$) causes a significant drop of $-3.7\%$, demonstrating the need for sufficient nonlinearity. Conversely, a deeper flow ($K = 5$) yields no gain (90.4%, $\Delta = -0.7\%$), suggesting $K = 3$ offers the best balance of expressiveness and stability for this task.

We further examine the impact of the classifier type within the same 5-stage CIL setting (Table 3). Compared with the Nearest Mean Classifier (NMC) and Softmax, the RRC consistently achieves higher accuracy across all incremental steps, with a particularly large margin in the final stage (S5: 91.1% vs. 80.2% for NMC and 81.6% for Softmax). This demonstrates that distribution-aware closed-form reconstruction not only reduces bias toward new classes but also enhances long-term retention of old class knowledge.

**Model choice: MLP vs. CNN.** Although CNNs are a natural choice for imagery, on our 64×64 SAR chips with strict rehearsal budgets a lightweight MLP delivers equal or better CIL accuracy at substantially lower training cost. Under identical preprocessing, augmentation, optimizer, schedule, and exemplar budget, the MLP attains a higher mean accuracy

Table 2: Ablation study with different modules and different flow depth $K$ at 5-stage, last step accuracy setting.

| Variant | Final Step Acc. (%) | Δ vs. Full |
|---|---|---|
| Full model (memory + NF + RRC) | **91.1** | – |
| Gaussian prior only | 69.6 | -21.5 |
| + memory | 79.7 | -11.4 |
| + memory + NF | 83.4 | -7.7 |
| + memory + RRC | 87.1 | -4.0 |
| *Flow depth K (planar flows)* | | |
| Full model w/ $K$=1 | 87.4 | -3.7 |
| Full model w/ $K$=3 | **91.1** | – |
| Full model w/ $K$=5 | 90.4 | -0.7 |

Table 3: Impact of classifier choice on accuracy (%) in the 5-stage CIL setting.

| Classifier | S1 | S2 | S3 | S4 | S5 |
|---|---|---|---|---|---|
| NMC | 99.1 | 96.4 | 91.5 | 85.0 | 80.2 |
| Softmax | 99.3 | 96.8 | 92.3 | 85.9 | 81.6 |
| RRC | **99.1** | **98.6** | **98.0** | **94.8** | **91.1** |

Table 4: CIL results on MSTAR (5-stage). "Train time" rows list wall-clock minutes per stage on a single 3070 GPU. "Params (M)" denotes model size in millions of parameters.

| | S1 | S2 | S3 | S4 | S5 | Mean | Params (M) |
|---|---|---|---|---|---|---|---|
| MLP | 99.1 | 98.5 | 98.0 | 94.8 | 91.1 | 96.3 | 16.0 |
| Train time | 6 | 9 | 11 | 14 | 19 | 11.8 | - |
| CNN | 99.1 | 98.7 | 95.5 | 92.6 | 90.1 | 95.6 | 13.2 |
| Train time | 10 | 14 | 19 | 25 | 29 | 19.4 | - |

(96.3% vs. 95.6%) and a stronger last-step score (S5: 91.1% vs. 90.1%), while reducing average per-stage training time from 19.4 to 11.8 minutes (about 1.6× faster), see Table 4. Despite having more parameters (16.0M vs. 13.2M), the MLP trains faster: the CNN decoder stacks several transposed convolution blocks that are compute and memory intensive on small chips. Crucially, parameter count does not equal training time: small kernel convolutions incur higher per pixel FLOPs and memory traffic, fragmented kernel launches, and lower hardware utilization on 64×64 inputs, whereas the MLP path relies mainly on dense matrix operations with higher throughput. This explains the CNN's similar accuracy but consistently longer wall clock time, and motivates our choice of MLP as the default encoder for stability, efficiency, and reproducibility.

# 5 Conclusion

We presented an NF prior-enhanced VAE framework with a numerically optimized classifier, integrated into the iCaRL paradigm for CIL of SAR imagery. By incorporating planar flows into the latent prior, our method better models the heavy-tailed statistics of SAR data, enabling the generation of more representative pseudo-samples for generative replay. Coupled with exemplar rehearsal and distribution-aware classifier reconstruction, the proposed approach effectively mitigates catastrophic forgetting while maintaining adaptability to new classes. Across standard CIL evaluations on MSTAR, our method achieves consistent improvements over Gaussian-prior VAE and iCaRL baselines, with the most notable advantages emerging in the later incremental steps. Ablation analyses confirmed the individual contributions of the NF prior, sufficient flow layers, and classifier choice, while t-SNE visualizations illustrated the superior feature separability preserved by our approach.

In future work, we will explore a unified OOD detection framework that combines dataset based generative distribution modeling with Weibull based statistical calibration. The goal is to jointly address incremental learning and reliable open set recognition in SAR imagery.

# References


[1] Matthias De Lange and Tinne Tuytelaars. Continual prototype evolution: Learning online from non-stationary data streams. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8250–8259, 2021.

[2] Arthur Douillard, Matthieu Cord, Charles Ollion, Thomas Robert, and Eduardo Valle. Podnet: Pooled outputs distillation for small-tasks incremental learning. In *European Conference on Computer Vision*, pages 86–102. Springer, 2020.

[3] Fei Gao, Heqing Huang, Jun Wang, Jinping Sun, Amir Hussain, and Huiyu Zhou. A comprehensive framework for out-of-distribution detection and open-set recognition in sar targets. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 18:16095–16109, 2025. doi: 10.1109/JSTARS.2025.3580373.

[4] Fei Gao, Mingyang Li, Jun Wang, Jinping Sun, Amir Hussain, and Huiyu Zhou. General sparse adversarial attack method for sar images based on key points. *IEEE Transactions on Aerospace and Electronic Systems*, pages 1–19, 2025. doi: 10.1109/TAES.2025.3588821.

[5] Rui Gao and Weiwei Liu. Ddgr: Continual learning with deep diffusion-based generative replay. In *International Conference on Machine Learning*, pages 10744–10763. PMLR, 2023.

[6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

[7] Chen He, Ruiping Wang, Shiguang Shan, and Xilin Chen. Introspective gan: Learning to grow a gan for incremental generation and classification. *Pattern Recognition*, 151: 110383, 2024.

[8] Run He, Di Fang, Yicheng Xu, Yawen Cui, Ming Li, Cen Chen, Ziqian Zeng, and Huiping Zhuang. Semantic shift estimation via dual-projection and classifier reconstruction for exemplar-free class-incremental learning. *arXiv preprint arXiv:2503.05423*, 2025.

[9] Xiaorui He, Chen Ding, Fei Qiao, and Jiaxuan Shi. An incremental remaining useful life prediction method based on wasserstein gan and knowledge distillation. In *2024 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 3857–3862. IEEE, 2024.

[10] Xiyue Hou, Wei Ao, Qian Song, Jian Lai, Haipeng Wang, and Feng Xu. Fusar-ship: Building a high-resolution sar-ais matchup dataset of gaofen-3 for ship detection and recognition. *Science China Information Sciences*, 63(4):140303, 2020.

[11] Huaibo Huang, Ran He, Zhenan Sun, Tieniu Tan, et al. Introvae: Introspective variational autoencoders for photographic image synthesis. *Advances in neural information processing systems*, 31, 2018.

[12] Ahmet Iscen, Jeffrey Zhang, Svetlana Lazebnik, and Cordelia Schmid. Memory-efficient incremental learning through feature adaptation. In *European conference on computer vision*, pages 699–715. Springer, 2020.

[13] Quentin Jodelet, Xin Liu, Yin Jun Phua, and Tsuyoshi Murata. Class-incremental learning using diffusion model for distillation and replay. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3425–3433, 2023.

[14] Eric R Keydel, Shung Wu Lee, and John T Moore. Mstar extended operating conditions: A tutorial. *Algorithms for Synthetic Aperture Radar Imagery III*, 2757:228–242, 1996.

[15] Diederik P Kingma, Max Welling, et al. An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, 12(4):307–392, 2019.

[16] Takashi Kitamura, Sachie K Ogawa, Dheeraj S Roy, Teruhiro Okuyama, Mark D Morrissey, Lillian M Smith, Roger L Redondo, and Susumu Tonegawa. Engrams and circuits crucial for systems consolidation of a memory. *Science*, 356(6333):73–78, 2017.

[17] Lingzhe Kong, Fei Gao, Xiaoyu He, Jun Wang, Jinping Sun, Huiyu Zhou, and Amir Hussain. Few-shot class-incremental sar target recognition via orthogonal distributed features. *IEEE Transactions on Aerospace and Electronic Systems*, 61(1):325–341, 2025. doi: 10.1109/TAES.2024.3443014.

[18] Dhireesha Kudithipudi, Mario Aguilar-Simon, Jonathan Babb, Maxim Bazhenov, Douglas Blackiston, Josh Bongard, Andrew P Brna, Suraj Chakravarthi Raja, Nick Cheney, Jeff Clune, et al. Biological underpinnings for lifelong learning machines. *Nature Machine Intelligence*, 4(3):196–210, 2022.

[19] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

[20] Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

[21] Dheeren Ku Mahapatra, Kumari Rosy Pradhan, and Lakshi Prosad Roy. An experiment on mstar data for cfar detection in lognormal and weibull distributed sar clutter. In *2015 International Conference on Microwave, Optical and Communication Engineering (ICMOCE)*, pages 377–380. IEEE, 2015.

[22] Zichong Meng, Jie Zhang, Changdi Yang, Zheng Zhan, Pu Zhao, and Yanzhi Wang. Diffclass: Diffusion-based class incremental learning. In *European Conference on Computer Vision*, pages 142–159. Springer, 2024.

[23] Martin Mundt, Iuliia Pliushch, Sagnik Majumder, Yongwon Hong, and Visvanathan Ramesh. Unified probabilistic deep continual learning through generative replay and open set recognition. *Journal of Imaging*, 8(4):93, 2022.

[24] Grégoire Petit, Adrian Popescu, Hugo Schindler, David Picard, and Bertrand Delezoide. Fetril: Feature translation for exemplar-free class-incremental learning. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 3911–3920, 2023.

[25] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017.

[26] Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *International conference on machine learning*, pages 1530–1538. PMLR, 2015.

[27] Johannes Schneider. Explainable generative ai (genxai): a survey, conceptualization, and research agenda. *Artificial Intelligence Review*, 57(11):289, 2024.

[28] Xiping Shang, Nannan Li, Dongjin Li, Jianwei Lv, Wei Zhao, Rufei Zhang, and Jingyu Xu. Ccldet: A cross-modality and cross-domain low-light detector. *IEEE Transactions on Intelligent Transportation Systems*, 26(3):3284–3294, 2025. doi: 10.1109/TITS.2024.3522086.

[29] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. *Advances in neural information processing systems*, 30, 2017.

[30] James Seale Smith, Leonid Karlinsky, Vyshnavi Gutta, Paola Cascante-Bonilla, Donghyun Kim, Assaf Arbelle, Rameswar Panda, Rogerio Feris, and Zsolt Kira. Coda-prompt: Continual decomposed attention-based prompting for rehearsal-free continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11909–11919, 2023.

[31] Shixiang Tang, Dapeng Chen, Jinguo Zhu, Shijie Yu, and Wanli Ouyang. Layerwise optimization by gradient decomposition for continual learning. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 9634–9643, 2021.

[32] Shipeng Wang, Xiaorong Li, Jian Sun, and Zongben Xu. Training networks in null space of feature covariance for continual learning. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 184–193, 2021.

[33] Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, et al. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *European conference on computer vision*, pages 631–648. Springer, 2022.

[34] Chenshen Wu, Luis Herranz, Xialei Liu, Joost Van De Weijer, Bogdan Raducanu, et al. Memory replay gans: Learning to generate new categories without forgetting. *Advances in neural information processing systems*, 31, 2018.

[35] Kun Xie, Xin Zhou, and Pu Yang. Assessment of statistical models for clutter and target in sar images. In *Proceedings of the 29th Chinese Control Conference*, pages 2997–3002. IEEE, 2010.

[36] Yicheng Xu, Yuxin Chen, Jiahao Nie, Yusong Wang, Huiping Zhuang, and Manabu Okumura. Advancing cross-domain discriminability in continual learning of vision-language models. *Advances in Neural Information Processing Systems*, 37:51552–51576, 2024.

[37] Shipeng Yan, Jiangwei Xie, and Xuming He. Der: Dynamically expandable representation for class incremental learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 3014–3023, 2021.

[38] Xianghu Yue, Xueyi Zhang, Yiming Chen, Chengwei Zhang, Mingrui Lao, Huiping Zhuang, Xinyuan Qian, and Haizhou Li. Mmal: Multi-modal analytic learning for exemplar-free audio-visual class incremental tasks. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 2428–2437, 2024.

[39] Guanxiong Zeng, Yang Chen, Bo Cui, and Shan Yu. Continual learning of context-dependent processing in neural networks. *Nature Machine Intelligence*, 1(8):364–372, 2019.

[40] Hanbin Zhao, Hui Wang, Yongjian Fu, Fei Wu, and Xi Li. Memory-efficient class-incremental learning for image classification. *IEEE Transactions on Neural Networks and Learning Systems*, 33(10):5966–5977, 2021.

[41] Fei Zhu, Xu-Yao Zhang, Chuang Wang, Fei Yin, and Cheng-Lin Liu. Prototype augmentation and self-supervision for incremental learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5871–5880, 2021.

[42] Huiping Zhuang, Zhenyu Weng, Hongxin Wei, Renchunzi Xie, Kar-Ann Toh, and Zhiping Lin. Acil: Analytic class-incremental learning with absolute memorization and privacy protection. *Advances in Neural Information Processing Systems*, 35:11602–11614, 2022.