Supplementary Material

Precip-UNetGAN: An Attention-Guided Multi-Scale GAN for Radar-Based Precipitation Nowcasting with Metric-Oriented Optimization

*Juncong Shao and Yanhong Liu**

School of Software, Henan University, Kaifeng 475004, China

*Corresponding author: yanhongliu@vip.henu.edu.cn

Contents

- Supplementary Method S1. Detailed architecture of Precip-UNetGAN
- Supplementary Method S2. dBZ-weighted reconstruction loss
- Supplementary Method S3. Time-weighted Soft-CSI formulation
- Supplementary Method S4. Training and optimization procedure
- Supplementary Tables S1-S3
- Supplementary Figures S1-S3

Supplementary Method S1. Detailed architecture of Precip-UNetGAN

Precip-UNetGAN receives 12 consecutive radar-derived precipitation fields sampled at 5-min intervals and predicts six future fields corresponding to lead times of 5, 10, 15, 20, 25, and 30 min. The generator is based on an encoder-decoder architecture with skip connections. Multi-scale spatial attention is applied to intermediate feature maps, and a convolutional discriminator supplies an adversarial constraint on the generated precipitation morphology.

S1.1 Input-output representation

Let X denote the historical sequence and Y the corresponding future observations:

$$X \in \mathbb{R}^{(T_in \times H \times W)},\ Y \in \mathbb{R}^{(T_out \times H \times W)},\ T_in = 12,\ T_out = 6. \quad \text{(S1)}$$

The generator G maps the historical sequence to the predicted sequence $\hat{Y}$:

$$\hat{Y} = G(X). \quad \text{(S2)}$$

H and W denote the grid height and width. In the reported experiments, all fields were cropped or resampled to 288 × 288 pixels before model input.

S1.2 Generator

The generator uses a UNet-style encoder-decoder. The encoder successively reduces spatial resolution and increases the feature-channel dimension, allowing broad precipitation organization to be represented. The decoder restores spatial resolution and combines decoder features with encoder features through skip connections, preserving local gradients and compact precipitation structures. The final output layer produces six continuous rain-rate fields.Each encoder or decoder block consists of convolution, normalization, and nonlinear activation operations. The channel numbers, convolutional settings, normalization layers, activation functions, and upsampling operations are summarized in Supplementary Table S3.

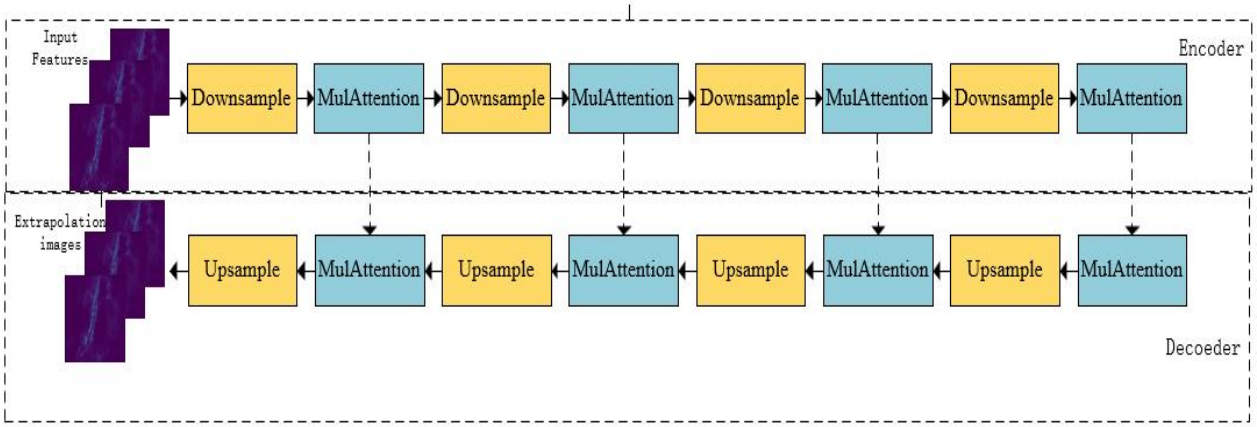


Fig. S1. Detailed architecture of the Precip-UNetGAN generator. The encoder extracts precipitation features at successively coarser spatial scales, while the decoder restores spatial resolution and merges encoder features through skip connections. The generator consists of convolutional blocks with the channel numbers, kernel sizes, strides, normalization layers, and activation functions specified in Supplementary Table S3.

S1.3 Multi-scale spatial attention

For an intermediate feature map U ∈ ℝ^(C×H_s×W_s), channel-wise average pooling and maximum pooling are used to generate two spatial descriptors. These descriptors are concatenated to form S:

$$\mathrm{U_avg} = \mathrm{AvgPool_c(U)},\ \ \mathrm{U_max} = \mathrm{MaxPool_c(U)},\ \ \mathrm{S} = [\mathrm{U_avg}\ ;\ \mathrm{U_max}]. \quad \text{(S3)}$$

The descriptor S is processed by three parallel convolutional branches with kernel sizes 3 × 3, 5 × 5, and 7 × 7. The branch outputs are

$$\mathrm{A_k} = \sigma(\mathrm{BN}(\mathrm{Conv_k(S)})),\ \ \mathrm{k} \in \{1,2,3\}, \quad \text{(S4)}$$

where BN denotes batch normalization and σ(·) denotes the sigmoid function. A trainable scalar q_k is assigned to each branch and converted into a normalized coefficient by a softmax function:

$$\beta_k = \frac{exp(q_k)}{\sum_{i=1}^{3} exp\,(q_i)}, \quad \sum_{k=1}^{3} \beta_k = 1. \qquad (S5)$$

The fused spatial attention map and the attention-enhanced feature map are

$$\mathrm{A} = \sum_{k=1}^{3} \beta_k\, A_k,\ \ \tilde{\mathrm{U}} = \mathrm{U} \odot \mathrm{A}, \quad \text{(S6)}$$

where ⊙ denotes element-wise multiplication. The three receptive-field sizes are intended to represent local echo boundaries, intermediate-scale precipitation structures, and broader precipitation organization. This interpretation concerns spatial representation in radar fields and does not imply explicit modeling of atmospheric governing equations.

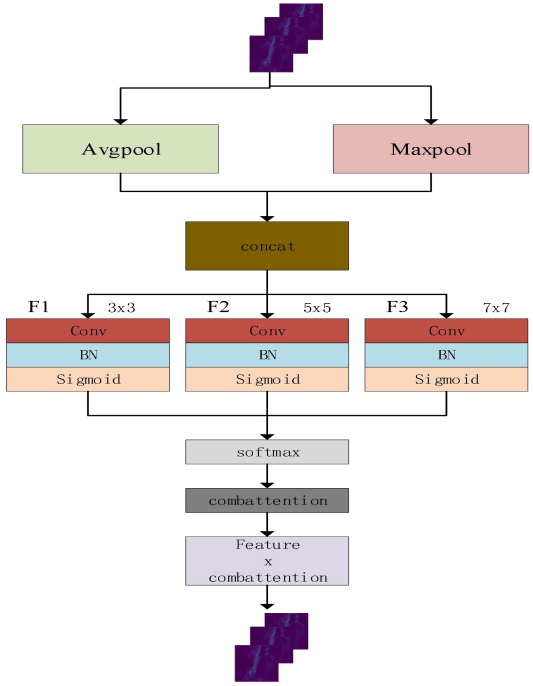


Fig. S2. Multi-scale spatial attention module. Channel-wise average and maximum descriptors are processed by 3 × 3, 5 × 5, and 7 × 7 convolutional branches. Trainable softmax-normalized coefficients combine the branch outputs into a spatial attention map, which is multiplied element-wise with the input feature map.

S1.4 Discriminator and adversarial objective

The convolutional discriminator D receives either an observed future field Y or a generated field $\hat{Y}$ and returns a score indicating whether the input is observational or generated. The discriminator is optimized to distinguish the two sources, whereas the generator is optimized to produce fields that the discriminator classifies as observational. This constraint is used to reduce diffuse boundaries and excessive smoothing in regression-only forecasts.

$$D(Y) \in [0,1], \quad D(\hat{Y}) \in [0,1] \quad \text{(S7)}$$

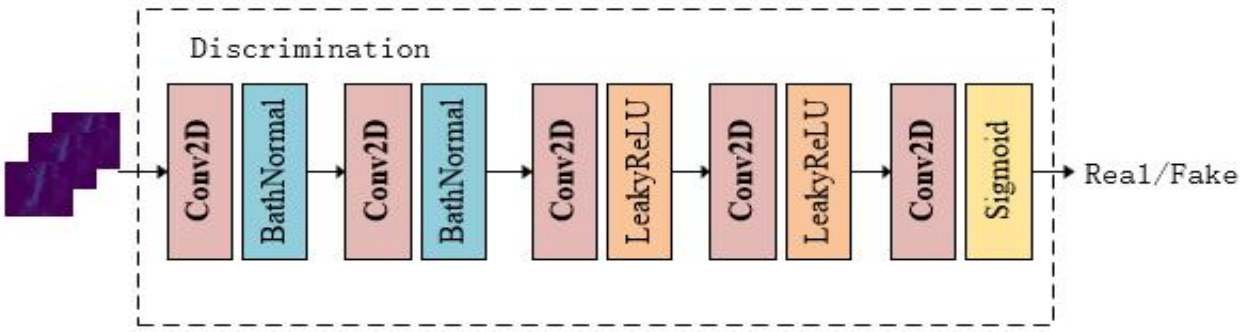


Fig. S3. Detailed architecture of the convolutional discriminator. The figure shows the number of channels, kernel sizes, strides, normalization layers, activation functions, and the final output operation.

Supplementary Method S2. dBZ-weighted reconstruction loss

The KNMI input consists of radar-derived, gauge-adjusted precipitation accumulation rather than raw radar reflectivity. Accordingly, the dBZ values used in the loss are diagnostically derived from rain rate through a fixed Z-R relation and should not be interpreted as direct observations of reflectivity.

S2.1 Rain-rate-to-reflectivity transformation

Let R denote rain rate in mm/h. The Marshall-Palmer relation is used to derive a reflectivity factor Z:

$$Z = aR^b, \quad a = 200, \quad b = 1.6 \quad \text{(S8)}$$

where Z is expressed in $mm^6\ m^{-3}$. The logarithmic reflectivity is

$$z = 10\log_{10}(Z) = 10\log_{10}(200R^{1.6}), \quad \text{(S9)}$$

where z is expressed in dBZ. The fixed relation provides a monotonic intensity transformation for constructing the weighted loss. Because Z-R coefficients vary with precipitation type and drop-size distribution, the transformed values are used only as an optimization representation and not as a retrieval of raw reflectivity.

S2.2 Piecewise weighting

Let z and $\hat{z}$ denote the transformed observed and predicted values. The weighted reconstruction loss is

$$\mathscr{L}_{DW-MSE} = E\left[W(Z)\cdot\ (\hat{Z}-Z)^2\right] \quad \text{(S10)}$$

The weighting function is separated at 25, 40, and 60 dBZ:

$$W(Z) = \begin{cases} W1, & Z < 25\ \text{dBZ}, \\ W2, & 25\ \text{dBZ} \le Z < 40\ \text{dBZ}, \\ W3, & 40\ \text{dBZ} \le Z < 60\ \text{dBZ}, \\ W4, & Z \ge 60\ \text{dBZ}. \end{cases} \quad \text{(S11)}$$

Larger weights assign a greater contribution to rarer, stronger transformed echoes.

Supplementary Method S3. Time-weighted Soft-CSI formulation

The conventional critical success index is calculated from hard threshold exceedances and is therefore non-differentiable. A sigmoid approximation is used to construct differentiable event indicators for multiple rain-rate thresholds and forecast lead times.

S3.1 Soft event indicators

For forecast time t and rain-rate threshold θ, define

$$p_{t,\theta}=\sigma\left(\frac{\hat{Y}_t-\theta}{\tau}\right), g_{t,\theta}=\sigma\left(\frac{Y_t-\theta}{\tau}\right) \quad \text{(S12)}$$

where $p_{t,\theta}$ and $g_{t,\theta}$ take values in [0,1], $\sigma(x)=1/(1+\exp(-x))$, and $\tau>0$ is a temperature parameter with the same unit as rain rate. A smaller τ approaches hard thresholding, while a larger τ gives a smoother transition. The reported experiments use $\tau=0.5$.

S3.2 Soft contingency quantities

The soft true-positive, false-positive, and false-negative amounts are

$$TP_{t,\theta}=E\left[p_{t,\theta}\cdot g_{t,\theta}\right] \quad \text{(S13)}$$

$$FP_{t,\theta}=E\left[p_{t,\theta}\cdot (1-g_{t,\theta})\right] \quad \text{(S14)}$$

$$FN_{t,\theta}=E\left[(1-p_{t,\theta})\cdot g_{t,\theta}\right] \quad \text{(S15)}$$

For a batch of B fields with grid size H × W, the averaging operator is

$$E[f]=\frac{1}{BHW}\sum_{b=1}^{B}\sum_{i=1}^{H}\sum_{j=1}^{W} f_{b,i,j} \quad \text{(S16)}$$

S3.3 Soft-CSI and temporal weighting

The differentiable CSI at time t and threshold θ is

$$CSI_{t,\theta}=\frac{TP_{t,\theta}}{TP_{t,\theta}+FP_{t,\theta}+FN_{t,\theta}+\varepsilon} \quad \text{(S17)}$$

where $\varepsilon = 0.01$ is a small numerical constant used to prevent division by zero.

Later forecast times are assigned normalized weights

$$w_t=\frac{t^{\alpha}}{\frac{1}{T}\sum_{k=1}^{T} k^{\alpha}} \quad \text{(S18)}$$

so that the mean temporal weight equals 1. In the reported experiments, $T=6$ and $\alpha=0.5$.

For the threshold set $\Theta=\{0.5, 2, 5, 10\}$ mm/h, the loss is

$$\mathcal{L}_{\mathrm{TW-SoftCSI}}=\frac{1}{T|\Theta|}\sum_{t=1}^{T}\sum_{\theta\in\Theta}w_t\,(1-\mathrm{CSI}_{t,\theta}) \quad \text{(S19)}$$

This objective aligns part of the training signal with the categorical thresholds used for forecast verification. It does not remove all differences between differentiable training scores and hard-threshold evaluation, because the sigmoid indicators remain continuous and depend on τ.

Supplementary Method S4. Training and optimization procedure

S4.1 Joint objective

The complete generator objective is

$$\mathcal{L}_G=\lambda_{\mathrm{DW}}\mathcal{L}_{\mathrm{DW-MSE}}+\lambda_{\mathrm{CSI}}\mathcal{L}_{\mathrm{TW-SoftCSI}}+\lambda_{\mathrm{adv}}\mathcal{L}_{\mathrm{GAN}} \quad \text{(S20)}$$

The three components respectively constrain intensity-sensitive reconstruction, threshold-exceedance overlap, and adversarial structural realism. The coefficients control the relative contributions of intensity-sensitive reconstruction, threshold-exceedance overlap, and adversarial learning.

S4.2 Alternating optimization

Training proceeds by alternating updates of the discriminator and generator. For each minibatch, the generator first produces six forecast fields from the 12 input fields. The discriminator is then updated using observed and generated future fields. With the discriminator parameters fixed, the generator is updated using the weighted sum in Eq. (S20). The procedure is repeated for a maximum of 200 epochs with a batch size of 8.

Algorithm S1. Alternating training procedure

1. Sample a minibatch of historical sequences X and observed future sequences Y.
2. Generate forecasts $\hat{Y} = G(X)$.
3. Update D using observed Y and detached generated $\hat{Y}$.
4. Recompute or reuse $\hat{Y}$ and evaluate $\mathcal{L}_{\mathrm{DW-MSE}}$, $\mathcal{L}_{\mathrm{TW-SoftCSI}}$, and $\mathcal{L}_{\mathrm{GAN}}$.
5. Update G by minimizing $\mathcal{L}_G$.
6. Evaluate the selected validation criterion and save the best checkpoint.
7. Repeat until the stopping criterion or 200 epochs is reached.

S4.3 Reproducibility settings

The manuscript specifies the Adam optimizer, a maximum of 200 epochs, batch size 8, α=0.5, τ=0.5, and thresholds of 0.5, 2, 5, and 10 mm/h. Additional architecture and optimization settings are summarized in Supplementary Table S1.

Table S1. Model architecture and training hyperparameters used in the experiments.

Category	Parameter	Value	Description
Input/output	Historical input frames	12	60 min of history at 5-min intervals
Input/output	Forecast output frames	6	Lead times of 5, 10, 15, 20, 25, and 30 min
Input/output	Model grid size	288 × 288	Fields cropped or resampled before model input
Architecture	Generator	UNet encoder-decoder	Skip connections with multi-scale spatial attention
Architecture	Attention branch kernels	3 × 3, 5 × 5, 7 × 7	Parallel spatial-attention branches
Architecture	Discriminator conditioning	concat(X,Y) or concat(X,$\hat{Y}$)	18 input channels = 12 historical + 6 future fields
Architecture	Discriminator base channels	64	Conditional convolutional discriminator
Optimization	Optimizer	Adam	Used for both generator and discriminator
Optimization	Generator learning rate	1×10^{-4}	Fixed learning rate used in the reported configuration
Optimization	Discriminator learning rate	1×10^{-4}	Fixed learning rate used in the reported configuration
Optimization	Adam β_1, β_2	0.5, 0.999	Used for both generator and discriminator
Optimization	Maximum epochs	200	Maximum training duration
Optimization	Batch size	8	Minibatch size
Optimization	D:G update ratio	1:1	One discriminator update followed by one generator update
Loss	Weighted dBZ-MSE coefficient λ_wMSE	1.0	Intensity-sensitive reconstruction term
Loss	Adversarial coefficient λ_GAN	0.005	Weight of the adversarial generator term
Loss	GAN loss	Binary cross-entropy (BCE)	Applied to probabilities after the final sigmoid
Loss	Soft-CSI coefficient λ_CSI	0.5	Weight of the time-weighted Soft-CSI term
Loss	Soft-CSI temperature τ	0.5	Controls smoothness of the soft threshold
Loss	Soft-CSI thresholds Θ	0.5, 2, 5, 10 mm/h	Rain-rate thresholds used in the differentiable event loss
Loss	Temporal exponent α	0.5	Gives greater weight to later forecast times

Category	Parameter	Value	Description
Loss	Soft-CSI numerical ε	1×10^{-2}	Prevents division by zero
dBZ transform	Z–R relation	$Z = 200R^{1.6}$	Fixed diagnostic transformation from rain rate to reflectivity factor
dBZ transform	Piecewise dBZ weights	See Table S2	Separate high-intensity configuration used for r ≥ 10 mm/h

Two dBZ-weighting configurations were used in the experiments. For the models evaluated at rain-rate thresholds of 0.5, 2, and 5 mm/h, the four intensity weights were set to 1, 2, 5, and 10. Because precipitation exceeding 10 mm/h is substantially rarer, a separately trained configuration was used for the $r \geq 10$ mm/h experiment, with stronger weights of 1, 3, 8, and 15. The same dBZ interval boundaries were used in all experiments, whereas the interval weights differed between the standard configuration and the separately trained 10 mm/h configuration. The corresponding weighting schemes are summarized in Supplementary Table S2.

Table S2. dBZ weighting schemes used for different precipitation-intensity experiments.

Training configuration	Transformed reflectivity interval	Weight
Standard configuration: $r \geq 0.5$, 2, and 5 mm/h	$z < 25$ dBZ	1.0
	$25 \leq z < 40$ dBZ	2.0
	$40 \leq z < 60$ dBZ	5.0
	$z \geq 60$ dBZ	10.0
Separately trained high-intensity configuration: $r \geq 10$ mm/h	$z < 25$ dBZ	1.0
	$25 \leq z < 40$ dBZ	3.0
	$40 \leq z < 60$ dBZ	8.0
	$z \geq 60$ dBZ	15.0

Table S3. Layer configuration of the generator and conditional discriminator.

Network	Stage	Input size	Output size	Channels	Kernel/stride or operation	Normalization	Activation
Generator	Input	[B,12,H,W]	[B,12,H,W]	12	—	—	—
Generator	Initial feature block	[B,12,H,W]	[B,64,H,W]	12→64	Convolutional feature block	Block-defined	Block-defined
Generator	Attention 1	[B,64,H,W]	[B,64,H,W]	64	3×3, 5×5, 7×7 parallel branches	Batch normalization	Sigmoid; softmax scale fusion
Generator	Down 1	[B,64,H,W]	[B,128,H/2,W/2]	64→128	Downsampling block	Block-defined	Block-defined
Generator	Attention 2	[B,128,...]	same spatial size	128	3×3, 5×5, 7×7 parallel branches	Batch normalization	Sigmoid; softmax scale fusion
Generator	Down 2	[B,128,...]	[B,256,H/4,W/4]	128→256	Downsampling block	Block-defined	Block-defined
Generator	Attention 3	[B,256,...]	same spatial size	256	3×3, 5×5, 7×7 parallel branches	Batch normalization	Sigmoid; softmax scale fusion