We thank the reviewers for their valuable comments on the manuscript. In the following paragraphs, the reviewers' comments are in black font, and our point-by-point responses are in blue.

Major comments

Presentation quality

The quality of the figures is unacceptable, mostly due a lack of resolution. Some figures are just impossible to interpret (e.g. Fig. 8). I pointed out this issue in my first report, and I have to say I am quite annoyed that it has not been addressed. Instead, it has become worse. The recommendation to use colormaps that take into account colour-blindness was ignored, too.

Furthermore, the manuscript needs considerable language editing, and the line of arguments is often difficult to follow. I recommend putting much more emphasis into the readability and conciseness of the main text.

➔ As suggested by the reviewer, we have improved the quality of the all figures. We have also modified the colors of the figures to be color-blind friendly based on www.ColorBrewer.org

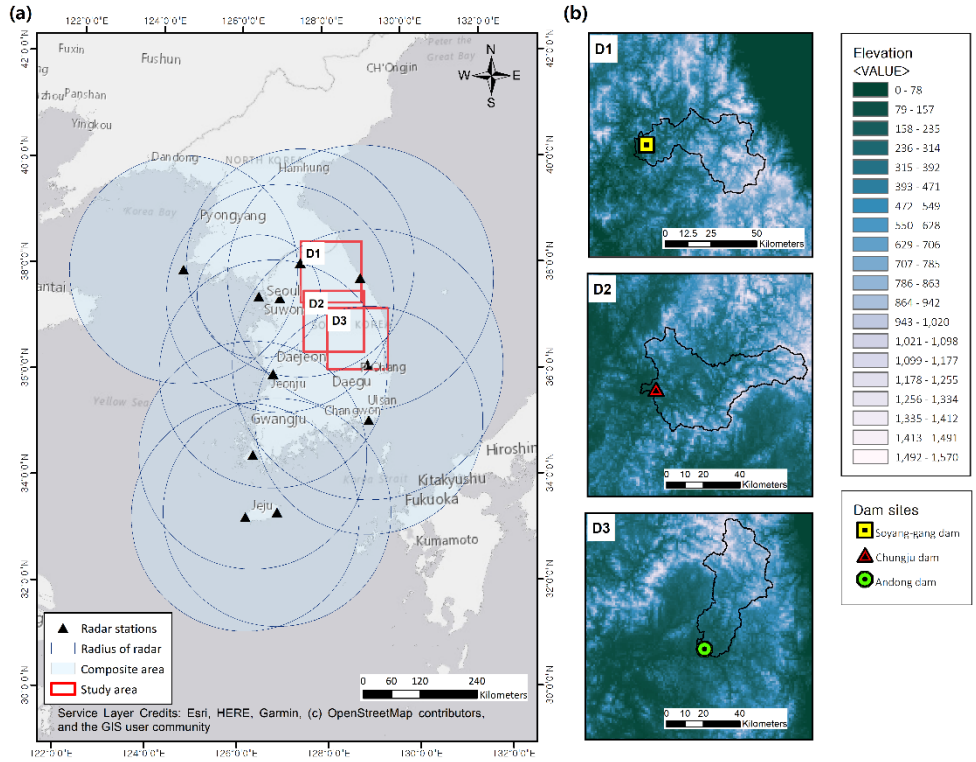

Figure 1: (a) Composite map of radar reflectivity and location of the dam basins; (b) selected areas over the dam basin. D1, D2, and D3 represent the areas of Soyang-gang, Chungju, and Andong Dam Basins, respectively. Maps were created using ArcGIS software by Esri; Base-map source: Esri, HERE, Garmin, © OpenStreetMap contributors, and the GIS User Community.

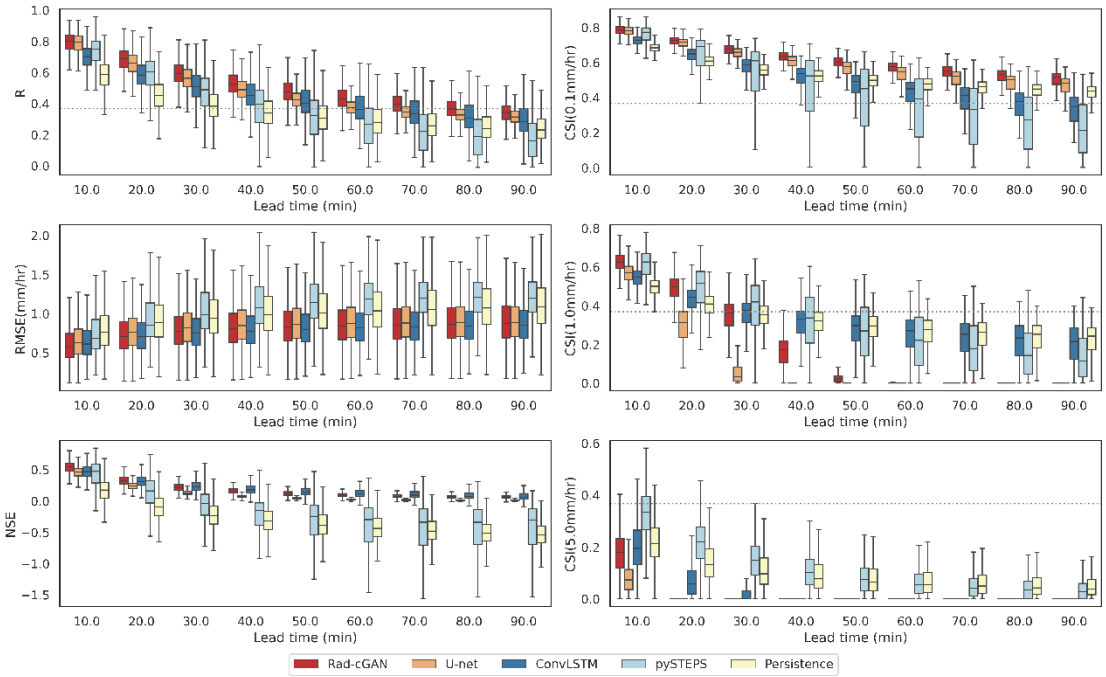

Figure 3: Box plot of the verification metrics of model predictions at the lead time up to 90 min over all grid cells from the Soyang-gang Dam region. From top to bottom, left panels represent R, RMSE, and NSE, and right panels represent the CSI at intensity thresholds of 0.1, 1.0, and 5.0 mm h^{-1}. Grey dotted line represents the predictability threshold (1/e≒0.37).

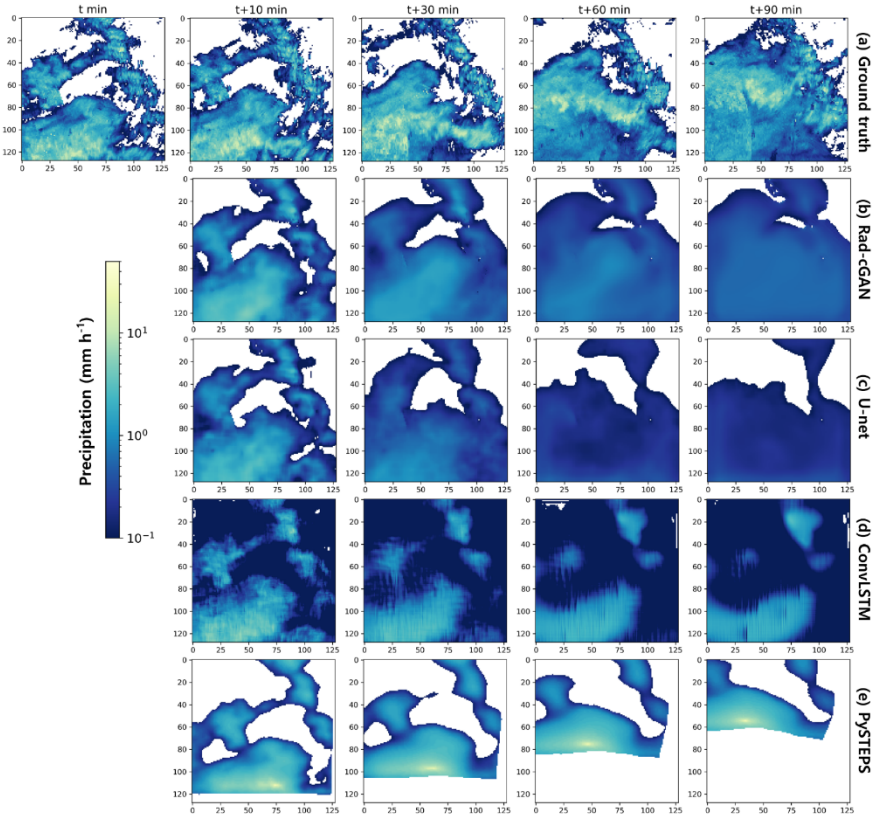

Figure 5: Example of precipitation at forecasting time t = 23 August 2018, 17:50 UTC, for model predictions and (a) ground truth (OBS). Panels from top to bottom express ground truth: (b) prediction of Rad-cGAN model, (c) prediction of U-net based model, (d) prediction of ConvLSTM, and (e) prediction of pySTEPS.

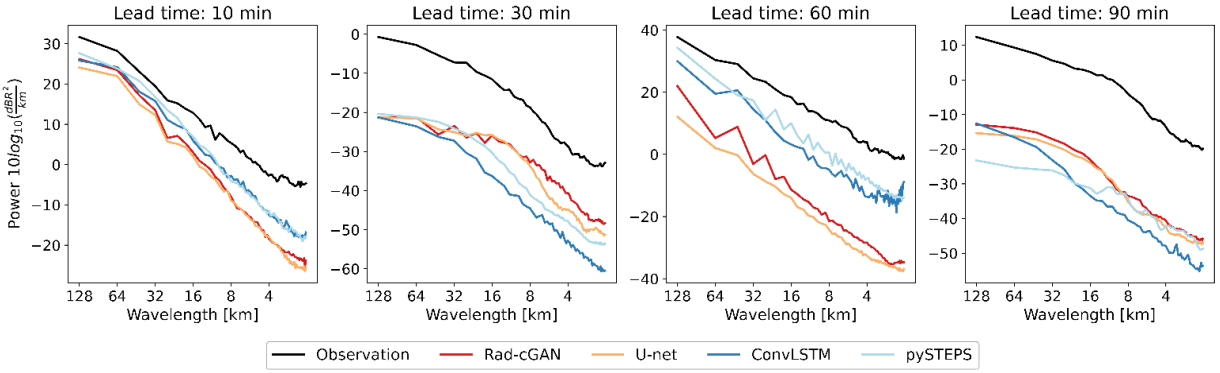

Figure 6: Radially averaged power spectral density (PSD) at forecasting time t = 23 August 2018, 17:50 UTC, for model predictions and observation.

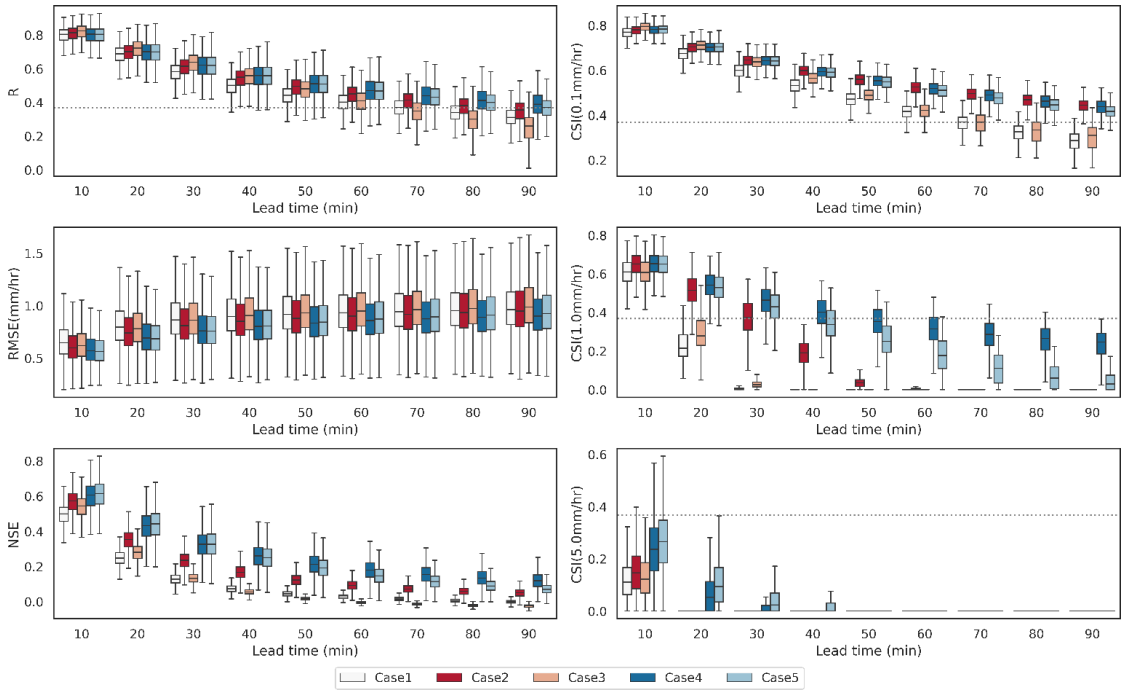

Figure 8: Box plot of the verification metrics of model predictions at lead time up to 90 min over all grid cells from the Andong Dam Basin. (a) R, (b) NSE, (c) RMSE, and (d, e, f) CSI at intensity thresholds of 0.1, 1.0, and 5.0 mm h^{-1}, respectively. Grey dotted line represents the predictability threshold (1/e ≒0.37).

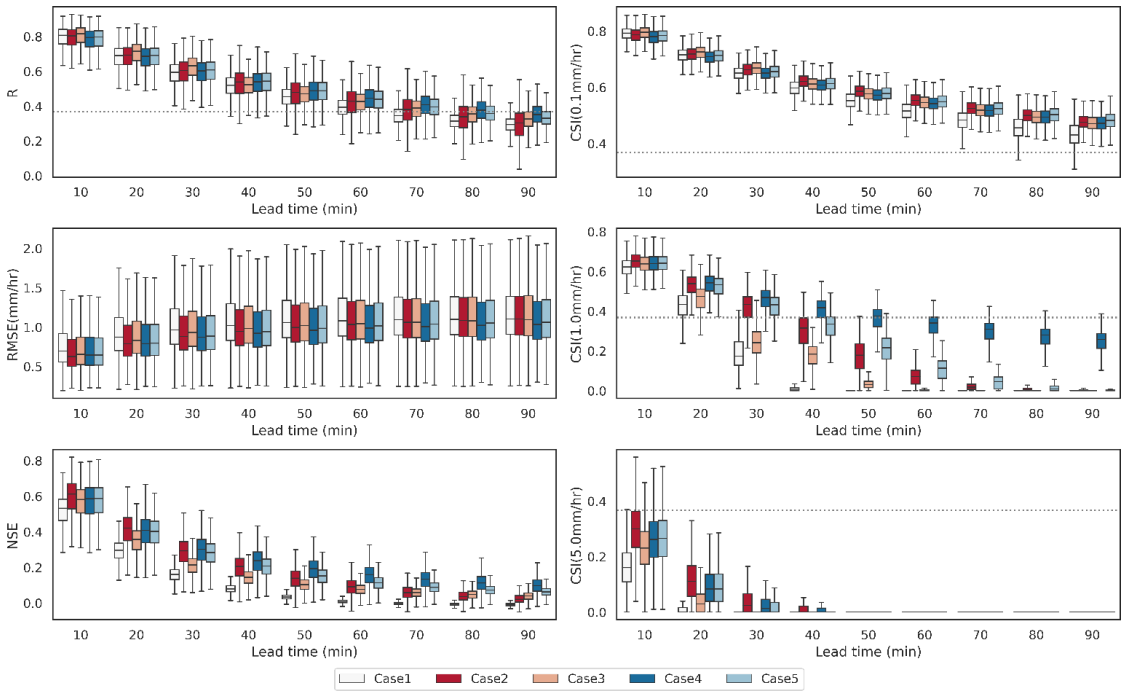

Figure 9: Box plot of the verification metrics of model predictions at the lead time up to 90 min over all grid cells from the Chungju Dam Basin. (a) R, (b) NSE, (c) RMSE, and (d, e, f) CSI at intensity thresholds of 0.1, 1.0, and 5.0 mm h^{-1}, respectively. Grey dotted line represents the predictability threshold (1/e ≒0.37).

Spatial verification set-up

I am not convinced by the author's arguments; from a dam management perspective, it is almost irrelevant how much precipitation falls at the location of the dam. The relevant parameter is the water level in the reservoir, which depends on the inflow, which in turn depends on the rainfall over the reservoir catchment. To be more explicit, I suggest dropping the entire part of the study that is related to this issue.

➔ As per the reviewer's suggestion, we have revised our manuscript to focus on the dam basin. We modified the values in Tables 4 and 5, which demonstrate the verification metrics for the dam sites to average values for the entire dam basin, and modified the line plots to box plots representing the values of the entire area in pixels.

Table 4: Comparison of the average values of the verification metrics for a 10-min precipitation prediction of different models at the Soyang-gang Dam Basin during summer (June–August), 2018.

	R	RMSE (mm h^{-1})	NSE	CSI (0.1 mm h^{-1})	CSI (1.0 mm h^{-1})	CSI (5.0 mm h^{-1})

Rad-cGAN	**0.7891**	**0.6138**	**0.5367**	**0.7827**	**0.6262**	0.1772
U-Net	0.7822	0.6626	0.4582	0.7783	0.5688	0.0793
ConvLSTM	0.6976	0.6508	0.4694	0.7247	0.5462	0.2019
pySTEPS (baseline)	0.7076	0.7826	0.4100	0.7181	0.5803	**0.3214**
Persistence (baseline)	0.5839	0.8117	0.1678	0.6821	0.4987	0.2197

Table 5: Comparison of the average values of the verification metrics for a 10-min precipitation prediction of the five different models using different transfer learning strategies for the (a) Andong and (b) Chungju Dam Basins in the summer (June–August) of 2018.

(a) Andong Dam domain						
	R	RMSE (mm h^{-1})	NSE	CSI (0.1 mm h^{-1})	CSI (1.0 mm h^{-1})	CSI (5.0 mm h^{-1})
Case 1	0.7945	0.8169	0.4926	0.7662	0.6073	0.1193
Case 2	0.8037	0.7673	0.5624	0.7756	0.6482	0.1523
Case 3	0.8146	0.7858	0.5351	0.7916	0.6067	0.1317
Case 4	0.7952	0.7407	0.5948	0.7782	0.6497	0.2399
Case 5	0.7952	0.7319	0.6051	0.7794	0.6472	0.2682
(b) Chungju Dam domain						
	R	RMSE (mm h^{-1})	NSE	CSI (0.1 mm h^{-1})	CSI (1.0 mm h^{-1})	CSI (5.0 mm h^{-1})
Case 1	0.7909	0.9221	0.5161	0.7893	0.6169	0.1639
Case 2	0.7863	0.8609	0.5876	0.7831	0.6492	0.2981
Case 3	0.7995	0.8849	0.5623	0.7920	0.6351	0.2324
Case 4	0.7776	0.8808	0.5661	0.7761	0.6380	0.2614
Case 5	0.7809	0.8783	0.5685	0.7803	0.6386	0.2657

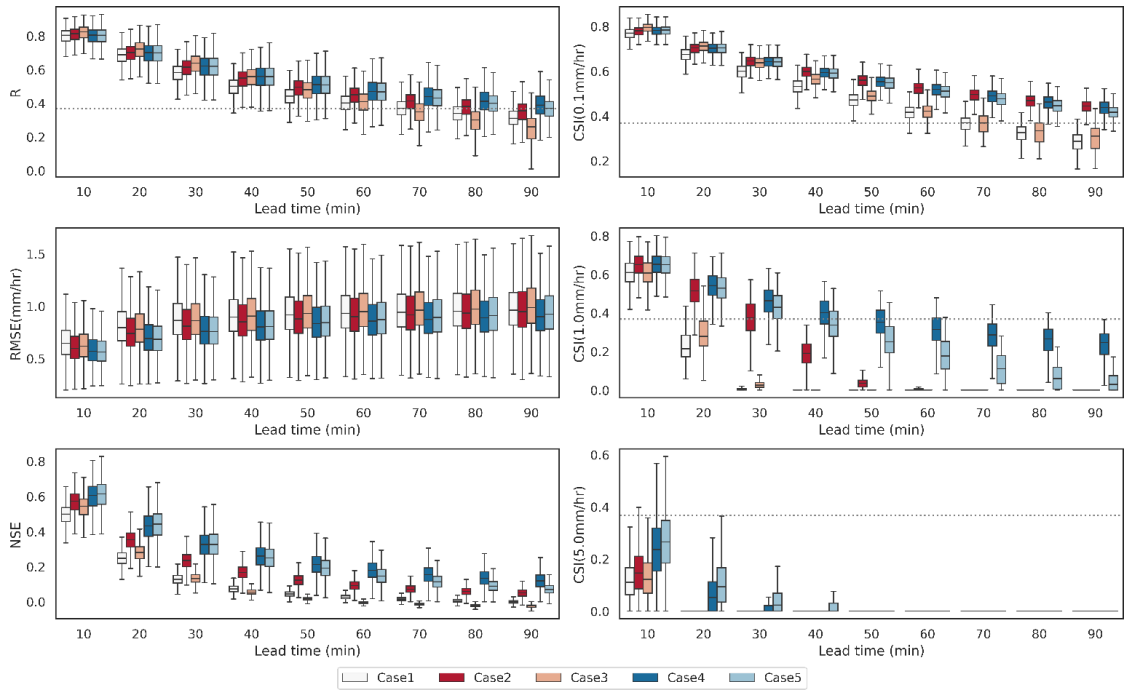

Figure 8. Box plot of the verification metrics of model predictions at the lead time up to 90 min over all grid cells from the Andong Dam Basin. (a) R, (b) NSE, (c) RMSE, and (d, e, f) CSI at intensity thresholds of 0.1, 1.0, and 5.0 mm h^{-1}, respectively. Grey dotted line represents the predictability threshold (1/e ≒0.37).

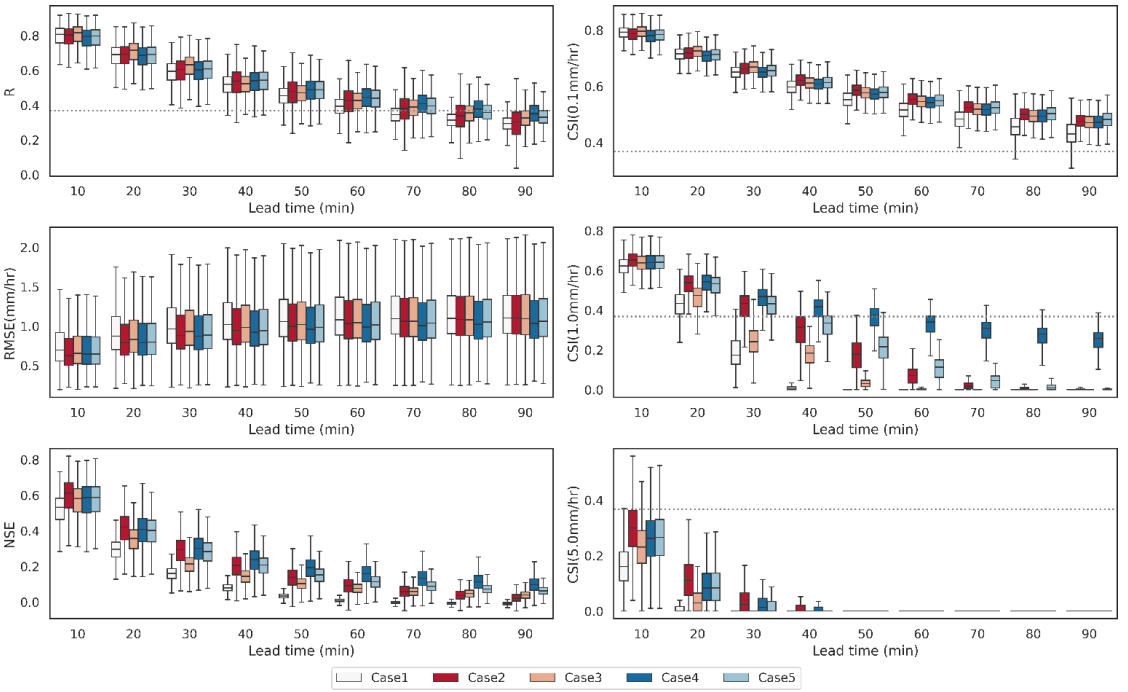

Figure 9. Box plot of the verification metrics of model predictions at the lead time up to 90 min over all grid cells from the Chungju Dam Basin. (a) R, (b) NSE, (c) RMSE, and (d, e, f) CSI at intensity thresholds of 0.1, 1.0, and 5.0 mm h^{-1}, respectively. Grey dotted line represents the predictability thresholds (1/e ≒0.37).

Lack of a competitive benchmark

I welcome the use of PySTEPS as a benchmark model. However, I think that the used verification metrics cannot be simply applied to a probabilistic ensemble forecast. Can you just compute the ensemble mean, and then treat it the same way you would treat a deterministic forecast? I don't think so. Please refer e.g. to Imhof et al. (2020) how to treat probabilistic PySTEPS forecasts in a verification context, or replace the ensemble forecast by a deterministic PySTEPS model. Otherwise, I suspect that the performance of the benchmark model might not be assessed fairly.

Looking at Fig 7, I am also very concerned about the role of edge effects in the verification of PySTEPS. Why not use a larger spatial window for the prediction to avoid this?

Overall, I am quite surprised by the poor performance of PySTEPS in comparison to other studies, specifically with regard to the CSI and FSS. What could be the reason?

➔ We have revised the benchmark model to S-PROG (Spectral Prognosis) from the pySTEPS library, which is a deterministic nowcasting model.

L218: *"2.3.1 PySTEPS*

PySTEPS (Pulkkinen et al., 2019) is an open-source and community-driven Python framework for radar-based deterministic and probabilistic precipitation nowcasting, and is considered a strong baseline model (Imhoff et al., 2019; Ravuri et al., 2021). In this study, deterministic S-PROG (Seed, 2003) nowcast from the pySTEPS library was used as the benchmark model.

To predict precipitation, we input the precipitation images (unit: dBR) transformed from four consecutive radar reflectivity images (from t-30 to t), which were the same as the input of the Rad-cGAN model, based on the Z-R relationship (Eq. (1)). Additionally, the transformed precipitation was used to estimate the motion field, which was used together with precipitation as input data in the model. Future precipitation at a lead time of up to 90 min for the test period (JJA, 2018) was generated from the results of the S-PROG nowcasts. The source code of pySTEPS is available at GitHub repository (https://pysteps.github.io, last accessed: 23 May 2022)."

We discussed the poor prediction performance of pySTEPS to be the reason for not predicting the rainfall area at the edge of the domain, as observed from the results of the typhoon event. As per the reviewer's suggestion, to reduce this edge effect, we have added the results of pySTEPS using 384×384 pixels input data, which was extended by 128 pixels on each side of the original input data. However, owing to limited data availability in the real-world, it was inappropriate to compare its performance with that of Rad-cGAN using extended data only for pySTEPS; hence, the edge-effect-

eliminated pySTEPS results were evaluated against Rad-cGAN that was trained using the same extended data (384×384 pixels).

L111: *"Furthermore, to reduce the edge effect caused by the fast Fourier transform (FFT), which is used for scale decomposition of pySTEPS (S-PROG) nowcast (Pulkkinen et al., 2019; Foresti and Seed, 2014), we derived the pySTEPS results using 384 × 384 km2 input data extended by 128 pixels on each side of the original input data (128 × 128 pixels)."*

L412: *"PySTEPS shows poor performance (Fig. 5) compared to previous studies (e.g., Imhoff et al., 2020) in the verification metrics (Table. 4 and Figs. 3-4). The overall prediction performance degrads particularly because the precipitation area near the edge of the basin is not predicted. To better understand this side effect, we reran pySTEPS and Rad-cGAN with the extended data of 384×384 pixels. Compared to the predictions in Fig. 5, the typhoon event (Fig. 7) shows that using the extended area reduces the edge effect of pySTEPS and properly maintains high rainfall intensity, thereby improving the performance. Moreover, the average R and CSI (at the highest rainfall intensity of 5.0 mm h-1) for the 10-min precipitation prediction during the entire test period is calculated as 0.77 and 0.38, respectively, indicating that the performance improves quantitatively compared to the previous results (R=0.70 and CSI=0.32). Additionally, the prediction performance of typhoon event improves using the extended area in Rad-cGAN (Fig. 7), and the average R and CSI (at the rainfall intensity of 5.0 mm h-1) in the 10-minute rainfall forecast for the entire test period improves from 0.79 to 0.80 and from 0.18 to 0.37, respectively. Both models show improved performance using extended area, but considering the applicability of the model to real-world problems with limited data availability, we conclude that, unlike pySTEPS, Rad-cGAN is more efficient in rainfall prediction without considering the edge effects of the spatial size of the input domains."*

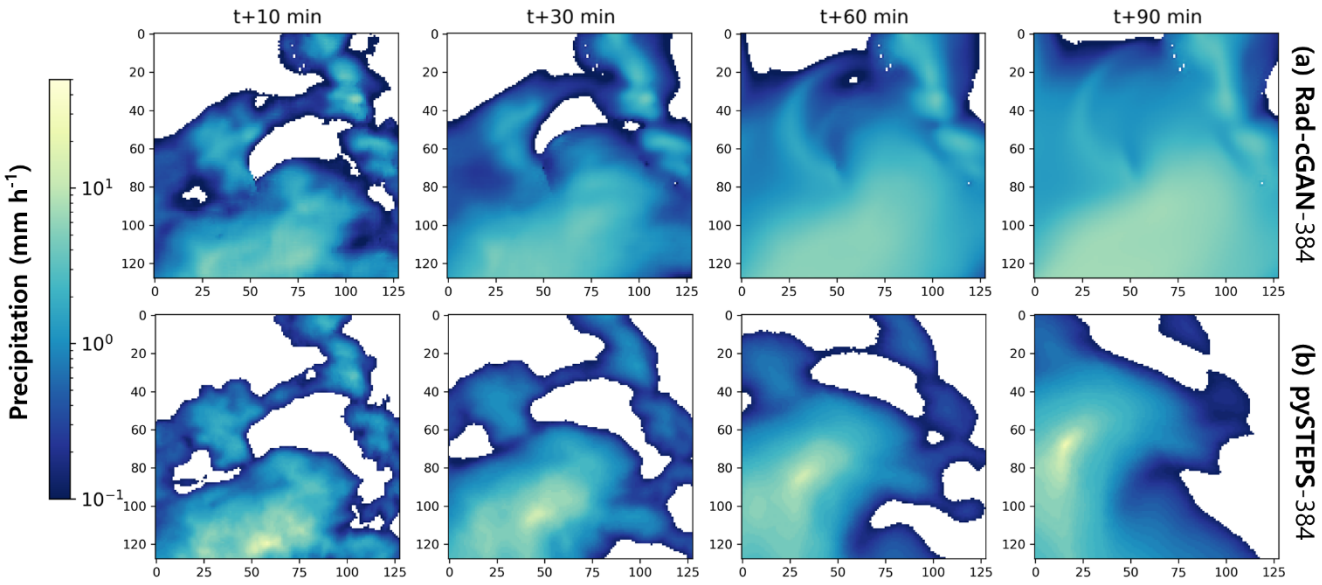

Figure 7. Example of precipitation at forecasting time t = 23 August 2018, 17:50 UTC, for model prediction using increased input area (384 × 384). Panels from top to bottom express (a) prediction of Rad-cGAN model, and (e) prediction of pySTEPS.

Model software and reproducibility

Improvements were made, but the repository does not yet contain the PySTEPS model implementation.

➔ We added the PySTEPS implementation to our repository.

Code and data availability: "*Source code of the model architecture is available at the GitHub repository https://github.com/SuyeonC/Rad-cGAN (last access: 13 June 2022). The pre-trained model for Soyang-gang Dam Basin and example data are available at https://doi.org/10.5281/zenodo.6460012. The radar reflectivity composite data samples provided by the Korea Meteorological Administration (KMA) are available at the public service: https://data.kma.go.kr/resources/html/en/ncdci.html (last access: 11 November 2021). The dataset for the entire period can be obtained through a separate request to the KMA.*"