Key factors for quantitative precipitation nowcasting using ground weather radar data based on deep learning

Daehyeon Han[1], Jungho Im[1], Yeji Shin[2,3], Juhyun Lee[1]

[1]Department of Urban and Environmental Engineering, Ulsan National Institute of Science and Technology, Ulsan, 44919, South Korea

[2]National Institute of Meteorological Sciences, Korea Meteorological Administration, Jeju-do, South Korea

[3]Market Intelligence Team, CJ CheilJedang Corporation, Seoul, South Korea

Correspondence to: Jungho Im (ersgis@unist.ac.kr)

Abstract

Quantitative precipitation nowcasting (QPN) can help to reduce the enormous socioeconomic damage caused by extreme weather. The QPN has been a challenging topic due to rapid atmospheric variability. Recent QPN studies have proposed data-driven models using deep learning (DL) and ground weather radar. Previous studies have primarily focused on developing DL models, but other factors for DL-QPN have not been thoroughly investigated. This study examined four critical factors in DL-QPN, focusing on their impact on forecasting performance. These factors are the deep learning model (U-Net, and convolutional long short-term memory, or ConvLSTM), input past sequence length (1 h, 2 h, or 3 h), loss function (mean squared error (MSE) or balanced MSE (BMSE)), and ensemble aggregation. Total of 24 schemes were designed to measure the effects of each factor using weather radar data from South Korea with a maximum lead time of 2 h. A long-term evaluation was conducted for the summers of 2020-2022, from an operational perspective, and a heavy rainfall event was analyzed to examine an extreme case. In both evaluations, U-Net outperformed ConvLSTM in overall accuracy metrics. For the critical success index (CSI), MSE loss yielded better results for both models in the weak intensity range (≤ 5 mm/h), whereas BMSE loss was more effective for heavier precipitation. There was a small trend where a longer input time (3 h) gave better results in terms of MSE and BMSE, but this effect was less significant than other factors. The ensemble by averaging results of using MSE and BMSE losses provided balanced performance across all aspects, suggesting a potential strategy to improve skill scores when implemented with optimal weights for each member. All DL-QPN schemes exhibited problems with underestimation and overestimation when trained by MSE and BMSE losses, respectively. All DL models produced blurry results as the lead time increased, while the non-DL model retained detail in prediction. With a comprehensive comparison of these crucial factors, this study offers a modeling strategy for future DL-QPN work using weather radar data.

1 Introduction

Short-term precipitation forecasting is an essential topic in weather forecasting, providing crucial information related to socioeconomic effects in daily life. Short-term precipitation forecasting within two hours is generally called quantitative precipitation nowcasting (QPN), which can be of great assistance in preventing damage from severe precipitation over a short period (Prudden et al., 2020). Despite the critical importance of QPN, it has been a challenging issue for a long time because of the complexity and dynamic characteristics of the atmosphere (Ravuri et al., 2021). Two major approaches to QPN exist numerical weather prediction (NWP) and statistical extrapolation (Prudden et al., 2020). The NWP simulates future atmospheric conditions, such as precipitation, pressure, temperature, and wind vectors, based on physical governing equations and global data assimilation. Even though NWP has been improved over decades with higher prediction skills and denser spatiotemporal resolution, NWP for QPN still has limitations due to its high computational cost, synoptic scale prediction, and spin-up issues (Yano et al., 2018; Bowler et al., 2006). For short term forecasting of precipitation, QPN has generally adopted extrapolation of the sequence of weather radar to focus on local rainfall with relatively high estimation accuracy (Wang et al., 2009; Ravuri et al., 2021; Prudden et al., 2020; Ayzel et al., 2020). Generally, QPN extrapolates the precipitation pattern using only radar sequences (Shi et al., 2015; Ravuri et al., 2021), but it can integrate other data sources, such as weather stations, NWP, and satellite data (Bowler et al., 2006; Haiden et al., 2011; Chung and Yao, 2020).

Weather radar provides real-time distribution of precipitation with high spatial (approximately 0.5-1 km) and temporal (about 5-10 minutes) resolutions. Various extrapolation approaches have been used for QPN from time-series weather radar data. Temporal extrapolation of the radar sequence demonstrated high prediction accuracy for 1-2 hours lead times, but the performance degraded as lead times increased. Several radar extrapolation methods have been developed, including thunderstorm identification tracking analysis and nowcasting (TITAN), tracking radar echo by correlation (TREC), and the McGill Algorithm for Precipitation Nowcasting by Lagrangian Extrapolation (MAPLE) (Dixon and Wiener, 1993; Mecklenburg et al., 2000; Germann and Zawadzki, 2002; Turner et al., 2004; Germann and Zawadzki, 2004). Despite their superior performance within a few hours compared with NWP, there have been limitations in predicting the onset of precipitation (Kim et al., 2021).

Recent advances in deep learning (DL) have altered conventional weather forecasting methods, especially for short-term predictions like QPN. The radar-based QPN can be viewed as a spatiotemporal video prediction that simulates upcoming frames based on past sequences (Han et al., 2023). Some studies used multiple input sources, such as meteorological variables, ground measurements, and NWP data (Adewoyin et al., 2021; Chen and Wang, 2022; Zhang et al., 2021a; Kim et al., 2021), but the majority of studies used only radar precipitation without any additional input sources. Among DL approaches, convolutional neural networks (CNNs) are widely used for spatial modeling in computer vision and geoscientific fields. Recurrent neural networks (RNNs) are expected to perform well on time-series datasets owing to their architecture, which recursively feeds the output as the following input and handles successive sequence data. Basic RNNs are not designed to consider spatial information, so there have been attempts to use RNNs for precipitation forecasting for

each gauge station (Kang et al., 2020). Shi et al. (2015) suggested convolutional long-short term memory (ConvLSTM) to combine the benefits of CNN and RNN to improve QPN performance in Hong Kong. ConvLSTM was designed to model spatiotemporal prediction by applying long-short-term memory (LSTM), one of the most popular RNN models, to convolutional CNN operations. Other RNN-based models of the trajectory gated recurrent unit (TrajGRU) and convolutional gated recurrent unit (ConvGRU) were proposed by (Shi et al., 2017). They reported that the deep learning models outperformed the operational models based on real-time optical flow by variational methods for echoes of radar (ROVER) by the Hong Kong observatory. Several studies using ConvLSTM, TrajGRU, and ConvGRU have demonstrated the superiority of CNN-RNN models over traditional approaches (Franch et al., 2020; Chen et al., 2020; Zhang et al., 2021b; Ravuri et al., 2021). However, some studies have only used CNNs for DL-QPN. The most widely used model is the U-Net (Ronneberger et al., 2015), which has a U-shaped structure with cascaded encoders, decoders, and skip connections. As U-Net can predict upcoming radar precipitation frames with a more straightforward form than RNN-fused models, it has been widely adopted in recent QPN studies employing deep learning (Ayzel et al., 2020; Agrawal et al., 2019; Samsi et al., 2019; Ko et al., 2022; Kim and Hong, 2021). In several application domains, CNNs have demonstrated their numerical robustness during training and made more accurate predictions than RNNs (Bai et al., 2018; Gehring et al., 2017). Recent studies have indicated that deep learning has become the predominant method for QPN owing to its superior performance compared to traditional approaches. However, there is still a dearth of exploration of the various considerations for the DL-QPN besides the DL model itself. As most studies have primarily focused on developing DL models across multiple study areas and datasets, it is difficult to determine how other factors can affect the skill score, even if they are crucial.

Considering this context, this study investigates critical factors that affect a DL-QPN model. Categorizing key factors in the DL-QPN is challenging, as there is a lack of standard agreement or explicit considerations in the literature. After analyzing various experimental designs used in previous studies, we summarized the following four critical factors in the DL-QPN: (1) the DL model, (2) the input sequence length, (3) the loss function, and (4) ensemble aggregation. The DL model has been the most highlighted factor in previous DL-QPN studies. The two representative types are fully convolutional networks (FCN) and a combination of CNN and RNN. U-Net (Agrawal et al., 2019; Ayzel et al., 2020; Trebing et al., 2021; Kim and Hong, 2021) and ConvLSTM (Shi et al., 2015; Jeong et al., 2021; Xiong et al., 2021) are the most popular models for FCN and CNN-RNN in DL-QPN, respectively. As the DL-QPN is data-driven, the sequence length will likely determine the model performance. However, the data sequence length has rarely been investigated in previous studies. Various time steps from 25 to 180 min were used as input data to predict future radar precipitation for up to 360 min, with little explicit comparison in the literature. Hence, the input sequence lengths were included as critical factors for further in-depth analysis. The loss function is a crucial element as it guides the training process of the model. In this context, we examined two widely employed loss functions in DL-QPN: the mean squared error (MSE) and the balanced MSE (BMSE). Lastly, we viewed ensemble of schemes as a potential key factor in DL-QPN. Although the ensemble method has not yet been a major topic of discussion in DL-QPN, we believe that this approach holds significant potential for enhancing accurate nowcasting in the future. A detailed explanation of these four key factors is provided in Section 2.

In this study, we compared diverse DL-QPN schemes considering four factors along with a non-DL model. Experiments were conducted in South Korea using weather radar data from 2012-2022 over June to August. As the DL-QPN is highly anticipated to mitigate damage from severe weather, heavy rainfall events in the Korean Peninsula were examined in detail. The remainder of this paper is organized as follows. A detailed explanation of the four key factors and comparison schemes is provided in section 2. Section 3 describes the data and methods used. Section 4 presents the results, and Section 5 discusses the results. Finally, Section 6 concludes the paper.

2 Key factors in DL-QPN

2.1 Deep learning model

The four critical factors identified from previous DL-QPN studies using weather radar are summarized in Table 1. Two basic models are CNN and RNN. CNNs have been widely adopted in various DL-QPN studies because of their outstanding performance in spatial modeling in various remote sensing and environmental studies including atmosphere (Lee et al., 2021; Gardoll and Boucher, 2022; Geiss et al., 2022; Chattopadhyay et al., 2022), ocean (Chinita et al., 2022; Barth et al., 2022), urban and land (Wu et al., 2022; Sato and Ise, 2022), and cryosphere (Lu et al., 2022; Kim et al., 2020b). The crucial part of CNNs is finding optimal convolutional filters to predict the target value with input data. The set of convolutional filters with specific window sizes (e.g., 3 × 3 or 5 × 5) was initialized with dummy values. The initial prediction is generated by conducting the dot product between the filters and the input over the cascading layers. The total error is calculated using the loss function between the model prediction and the actual value. The L1 (mean absolute error (MAE) or L2 (mean squared error (MSE) loss function is commonly used in supervised learning for regression; however, other types of loss functions can be used in DL-QPN (Ayzel et al., 2020; Ravuri et al., 2021). Following the loss calculation, the weights of the convolutional layers are updated through backpropagation. With an increasing number of iterations, the model was progressively fitted to the given dataset.

U-Net is one of the most representative image-to-image models among CNNs. Because DL-QPN can be viewed as image-to-image modeling, U-Net has been widely adopted in recent DL-QPN studies (Ayzel et al., 2020; Bouget et al., 2021; Ko et al., 2022; Kim and Hong, 2021). U-Net consists solely of convolutional layers that can preserve spatial information from the input to the output. The input for the DL-QPN with weather radar is the previous sequence of the radar images. This model is expected to generate a series of future precipitation scenes. The input and output are image sequences with the dimensions of [*ny, nx, M*] and [*ny, nx, N*], where *M* and *N* are the lengths of the sequence in the past and future, respectively. Its skip connections distinguish U-Net between the same level of encoding and decoding layers, which can mitigate the loss of original information as the network deepens (Ronneberger et al., 2015). In this study, RainNet v1.0 (https://github.com/hydrogo/rainnet) by Ayzel et al. (2020) was adopted for the U-Net model.

RNNs are expected to yield good performance in time-series forecasting. An RNN is distinguished by its recurrent layers, which feed the output of a specific layer back to its input. As the vanilla RNN structure suffers from the vanishing gradient problem with an increasing number of recurrent hidden layers, revised RNNs, such as long short term memory (LSTM) and gated recurrent units (GRU), have gained widespread acceptance (Cho et al., 2014; Hochreiter and Schmidhuber, 1997). They added additional gates to control the information transmitted or dropped, resulting in improved performance compared with vanilla RNNs. Several studies have been conducted on using RNNs for short-term rainfall forecasting (Ni et al., 2020; Srinivas et al., 2019; Aswin et al., 2018). However, these were station-based rainfall predictions that did not account for 2-D information because RNNs were not designed to take spatial information into account. Shi et al. (2015) suggested the ConvLSTM, which combines CNN and LSTM into a single model. Because ConvLSTM has been adopted in recent DL-QPN studies (Chen et al., 2020), it was compared with U-Net in this study. A detailed explanation of the ConvLSTM can be found in Shi et al. (2015).

Table 1. Summary of key factors used in previous image-to-image QPN only using radar sequence based on deep learning. 'B' in the loss column stands for 'Balanced,' which blends different weights for rainfall intensity or each input time.

Reference	Study area Data Source	Key factors			Others		
		DL model	input sequence	loss function	Model design	Prediction time	GAN
Shi et al. (2015)	HongKong Hong Kong Observatory 7 (HKO7)	ConvLSTM	5 (30 min.)	Cross entropy	Multiple	15 (90 min.)	-
Shi et al. (2017)	HongKong HKO-7	TrajGRU	5 (30 min.)	BMSE, BMAE	Multiple	20 (120 min.)	-
Agrawal et al. (2019)	USA Multi-Radar/Multi-Sensor System (MRMS)	U-Net	Unknown	Cross entropy	Unknown	Unknown	-
Ayzel et al. (2020)	Germany Deutscher Wetterdienst (DWD)	U-Net	6 (30 min)	LogCosh	Recursive	1 (5 min)	-
Franch et al. (2020)	Italy Trentino-Alto Adige/Südtirol	TrajGRU	5 (25 min.)	BMSE, BMAE	Multiple	20 (100 min.)	-

	Radar 2019 (TAASRAD19)						
Ravuri et al. (2021)	UK RadarNet4	ConvGRU	22 (110 min.)	Custom spatial and temporal losses	Multiple	18 (90 min.)	Conditional GAN
Xiong et al. (2021)	Hong Kong HKO-7	ConvLSTM	5 (30 min.)	MAE, MSE, BMAE, BMSE	Multiple	20 (120 min.)	-
Cuomo and Chandrasekar (2021)	USA Next Generation Weather Radar (NEXRAD)	CNN ConvGRU	16 (80 min.)	LogCosh	Multiple	16 (80 min.)	-
Trebing et al. (2021)	Netherlands Koninklijk Nederlands Meteorologisch Instituut (KNMI)	U-Net	12 (60 min.)	MSE	Single	1 (30 min.)	-
Jeong et al. (2021)	Korea KMA	CNN ConvLSTM	18 (180 min.)	MSE	Multiple	18 (180 min.)	-
(Kim and Hong, 2021; Jeong et al., 2021)	Korea KMA	U-Net	60 min.	MAE, MSE, BMSE	Multiple	240 min.	Conditional GAN
Ko et al. (2022)	Korea KMA	U-Net	7 (70 min.)	BCSI	Single	6 (360 min.)	-
(Liu et al., 2022)	China China Meteorological Data website (CMD)	LSTM	3 (3 hours)	MAE+MSE	Multiple	3 (3 hours)	-
Han et al., 2023	Korea KMA	Simpler yet better video prediction	12 (120 min.)	MSE	Multiple	12 (120 min)	-

140

2.2 Input sequence length

The DL-QPN predicts upcoming precipitation based on past sequences. Thus, the composition of the past series directly affects the model performance. A longer past sequence may provide more information than a shorter one, but it could also contain unnecessary information for model training. The optimal length of the past sequence can vary depending on other factors, such as the forecasting design, DL model, radar time interval, and maximum lead-time. Thus, it is difficult to determine the direct effect of past sequence length on DL-QPN. To examine the impact of the input sequence length, we compared radar sequences of 1 h, 2 h, and 3 h against 12 future radar scenes for 2 h. This sets up a ratio of input-output sequences at 1:2, 1:1, and 3:2.

2.3 Balanced loss function

The loss function guides the direct optimization of DL models. The basic loss function in DL-QPN is MSE. By summing up the error of each pixel, it produces a single value for a given prediction image. As most valid precipitation pixels are severely skewed in weak rainfall intensity (about ≤ 5 mm/h), calculating MSE (Equation 1) with a uniform weight for all pixels might result in an underestimation problem. Shi et al. (2017) suggested the BMSE to mitigate the sample imbalance by using different weights for precipitation intensity (Equation 2).

$$MSE = \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{N} \quad (1)$$

$$BMSE = \frac{\sum_{i=1}^{N} w(y_i)(y_i - \hat{y}_i)^2}{N}, w(y_i) = \begin{cases} 1, & y_i < 2 \\ 2, & 2 < y_i < 5 \\ 5, & 5 < y_i < 10 \\ 10, & 10 < y_i < 30 \\ 30, & y_i > 30 \end{cases}, \quad (2)$$

where y is the reference value, and $\hat{y}$ represents the predicted value and N is the number of all valid pixels within the radar area. Figure 1 shows the distribution of rainfall intensity and weights for BMSE.

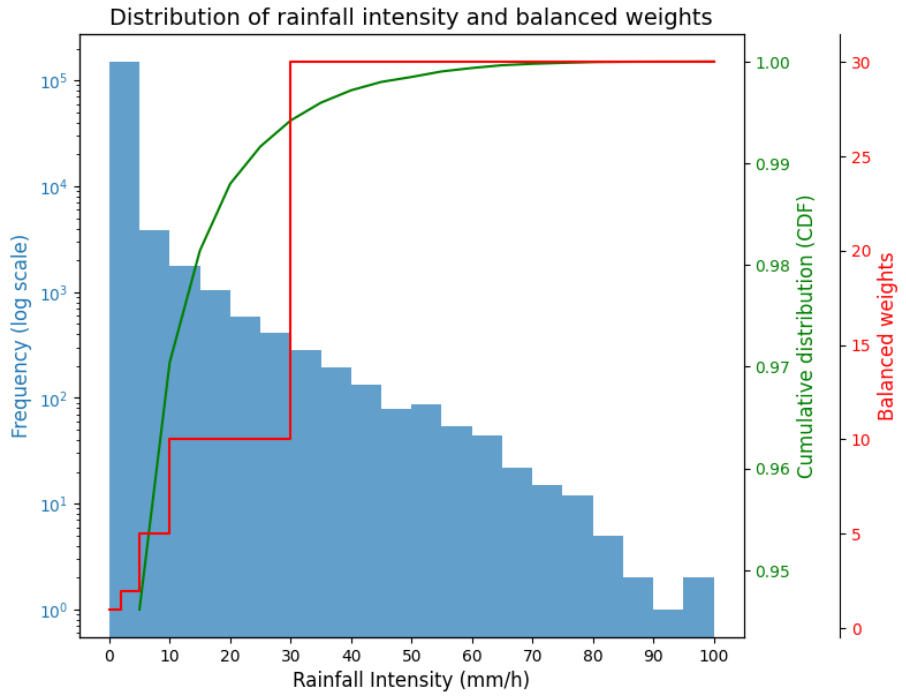

Figure 1. Mean distribution of rainfall intensity for the summers of 2020-2022 in a pixel window of 400×400. The blue bar represents the histogram of rainfall intensity. The green line shows the cumulative distribution function. The red line represents the balanced weights for mitigating data imbalances, as suggested by Shi et al. (2017).

2.4 Ensemble approach

Ensemble approaches have been adopted as the standard in NWP by combining multiple different members to produce robust results. Despite their potential, ensemble approaches have not been actively adopted in DL-QPN. The ensembling of multiple schemes can bring enhanced and more stable performance than a single model in various evaluation aspects. Ensemble can be executed with simple aggregation with equal weights, or it can employ another machine learning model to learn the optimal way of combining each prediction, such as a stacking ensemble (Cho et al., 2020; Franch et al., 2020). The primary interest of this study is how the results can be improved through ensemble. Hence, only a simple ensemble was done by averaging results from multiple schemes with same weight.

3. Data and Methods

3.1 Data and preprocessing

South Korea, located in Northeast Asia, has a population of approximately 50 million and hosts numerous industrial facilities. Every year, particularly during the summer monsoon season, the country experiences flooding, inundation, and landslides. Sudden heavy rainfall can occasionally disrupt urban transportation systems, and continuous heavy rain during the rainy season (Jangma in Korean) can result in dam failures and significant flooding in river basins. Eleven ground weather radars have been operated by the Korea Meteorological Administration (KMA) to monitor precipitation over South Korea. Multiple radars were combined to produce a composite radar reflectance image with a spatial resolution of 1 km (Figure 2). We used the constant altitude plan position indicator (CAPPI), widely used to study precipitation, with an altitude of 1.5 km provided by the KMA (Shi et al., 2017; Han et al., 2019; Kim et al., 2021). To exclude the area outside the radar coverage, we cropped the data from longitude 124E to 131.15E and latitude 33N to 39N. This encompasses most of the national territory of South Korea and a portion of North Korea. For the long-term test, the CAPPI dataset was collected every 10 minutes from 2012 to 2022 during the months of June, July, and August (JJA). A large radar sequence dataset was created using 144 radar scenes daily, with more than 142,560 radar scenes in the study period. Data from 2012-2018 were used for model training, data from 2019 were used for model validation (i.e., hyperparameter optimization). To evaluate the long-term performance from an operational standpoint, we used the summer data of 2020–2022 as the test dataset, which corresponds to approximately 38,880 radar scenes.

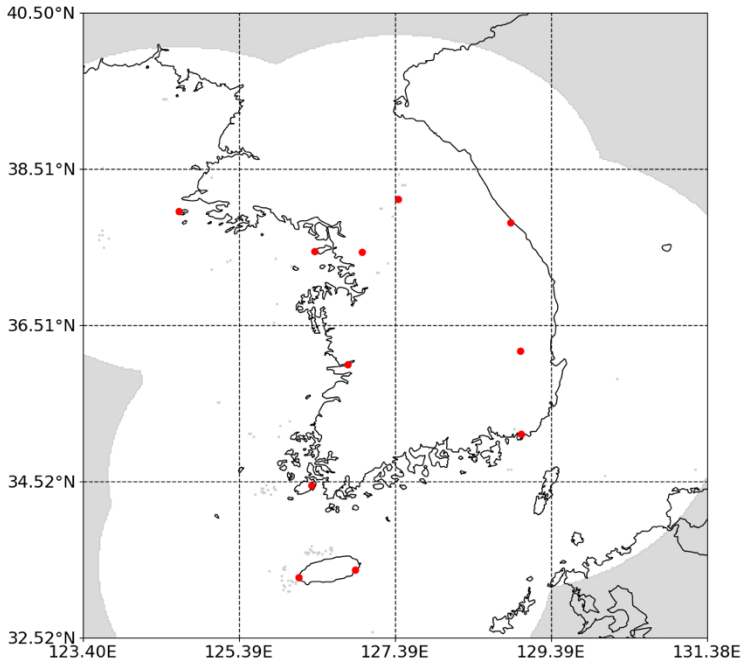

Figure 2 Weather radar over the Korean Peninsula used in this study. The grey shadow at the boundary indicates the area outside of valid radar coverage. The locations of the eleven weather radars are represented by red dots.

The KMA weather radar data were provided as reflectance values in decibels relative to Z (dBZ). Marshall–Palmer's Z-R equation (Marshall and Palmer, 1948) converts radar reflectance into precipitation intensity (Equation 1).

$$R = \frac{1}{200} Z^{0.625}, \quad (3)$$

where Z represents reflectance, and R is the precipitation intensity in mm/h. While the original CAPPI radar data had a spatial resolution of 1 km, they were resampled to 2 km, resulting in a 400 × 400 grid to reduce the model training time. As most pixels in the radar data have no precipitation value, we only used radar scenes when pixels with precipitation intensity higher than 10 mm/h existed in more than 3% of the study area for training.

3.2 Comparison with non-DL model

To compare the DL-QPN results with a non-DL model, we also included pySTEPS (Pulkkinen et al., 2019) in our comparison. PySTEPS is a Python implementation of the Short-Term Ensemble Prediction System (STEPS) proposed by Bowler et al. (2006). It has been widely used as a control non-DL model in previous studies (Ravuri et al., 2021; Choi and Kim, 2022; Han et al., 2023; Zhang et al., 2023). By calculating the mean wind vector using the input radar sequence,

pySTEPS simulates future radar sequences. To examine the impact of input sequence length, we also tested 1-3 h of input sequence in pySTEPS to predict a maximum of 2 h, the same as the other DL models. More detailed information and usage of pySTEPS can be found in its documentation (https://pysteps.github.io) and repository (https://github.com/pySTEPS/pysteps).

3.3 Scheme configuration

Considering these four key factors, total of 24 experimental schemes for DL-QPN and pySTEPS were designed (Table 2). The selection of the DL model and its tuning are crucial for maximizing forecasting accuracy. U-Net and ConvLSTM, two representative models in the DL-QPN, were compared in this study. The structures of the U-Net and ConvLSTM are summarized in Figure 3. The U-Net model uses five levels of spatial filters to utilize the different scales of hidden features (Figure 3a). At each level, two convolutional layers with a 3×3 kernel were used. The number of convolutional filters varied depending on the depth of the layer. Skip connections concatenated the equal-sized shallow and deep layers. Several tests were conducted for ConvLSTM to determine the optimal number of layers and hidden states for the ConvLSTM layer, considering the model performance and GPU's memory capacity. Four ConvLSTM layers were used in this study with 16, 32, 32, and 16 hidden states per layer (Figure 3b). All models were trained with the MSE and BMSE loss functions and adaptive momentum optimizer (ADAM) with a learning rate of 0.001, widely adopted in deep learning regression models (Kingma and Ba, 2014). Considering the GPU memory, the batch sizes for U-Net and ConvLSTM were set to 16 and 1, respectively. After the convolutional and ConvLSTM layers, the rectified linear unit (ReLU) and leaky ReLU with 0.1 threshold were used respectively as activation functions to model the nonlinearity of the data. Each model was trained with a maximum of 100 epochs, and the model training was terminated when the validation performance did not improve over three iterations.

It is notable that other DL models and hyperparameters are also to be considered in addition to these four factors. The common hyperparameters for deep learning include the batch size, activation function, learning rate and optimizer. Because finding the best set of hyperparameters is time-consuming, other combinations of hyperparameters were not considered at this time. Some recent studies have adopted generative adversarial networks (GAN) in DL-QPN (Ravuri et al., 2021; Kim and Hong, 2021). Although GAN can be expected to be a promising approach in DL-QPN, we did not compare it in this study because it is far beyond our scope owing to its complexity and diversity.

In DL models, there are various aspects that contain randomness, such as random weight initialization or a random mini batch from the entire samples. To evaluate the model performance more reliably, we ran each scheme 5 times with different random seeds of 0, 999, 2023, 44919, and 2022276. Starting from combining DL models (U-Net or ConvLSTM), loss functions (MSE or BMSE), and input sequence length (1 h, 2 h, or 3 h), total of 12 initial schemes were generated before ensemble averaging. As each initial scheme had 5 runs with different random scenes, a total of 60 runs for training DL models were conducted in this study (Table 2 and Figure 4). By aggregating 5 members for each scheme, each scheme can assure more stability than a single run. For example, UM1 stands for the average of U-Net trained by MSE loss with an input

sequence of 1 h for 5 different random seeds. The average of each scheme with different input sequence lengths was again aggregated; for example, UM is the ensemble mean of UM1, UM2, and UM3. Consequently, UM is the mean of the total 15 runs, same is true for UB, CM, and CB. The remaining ensemble processes are similar to those in Table 2.

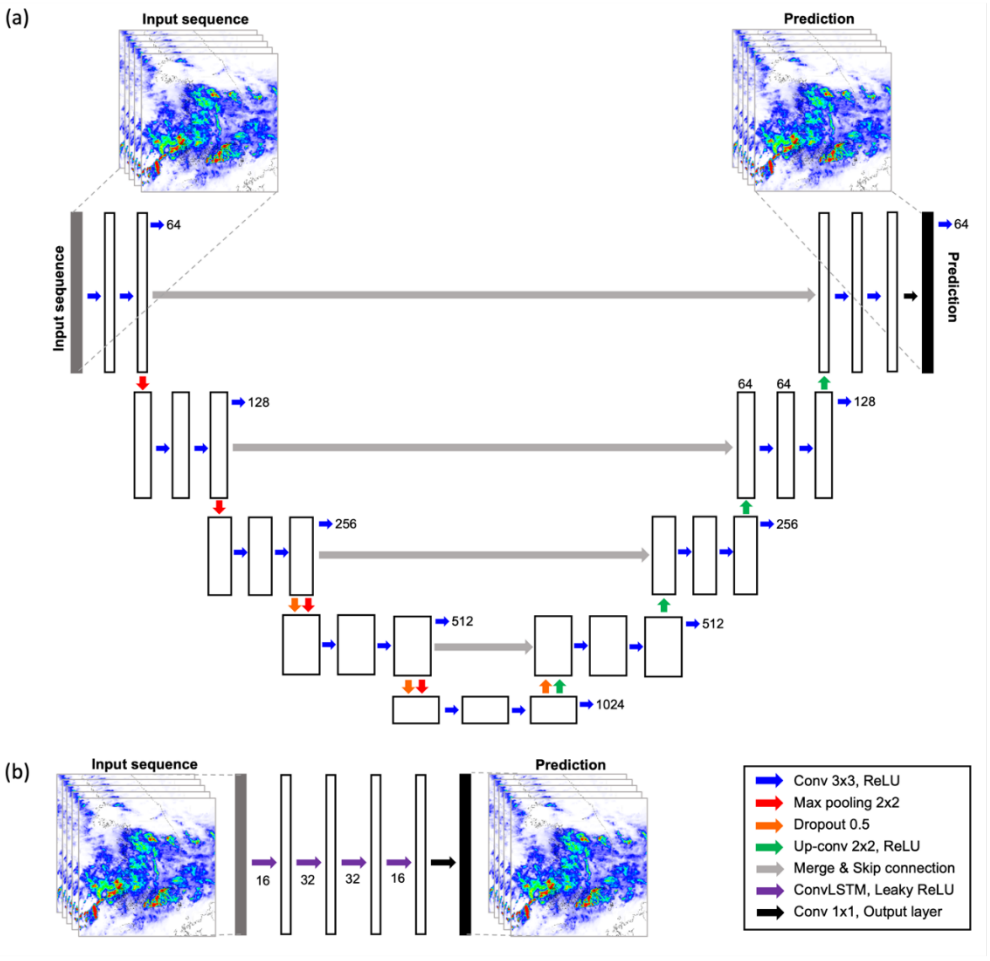

Figure 3. The structure of (a) U-Net and (b) convolutional long-short term memory (ConvLSTM) used in this study.

Table 2. Specifications of the schemes designed in this study with their abbreviations. 'U' stands for U-Net, 'C' stands for ConvLSTM, and 'P' is for pySTEPS. 'M' and 'B' indicate MSE and BMSE loss functions, respectively. The following number is the length of the input past sequence in hours. The number of runs indicates that there are *n* total members to yield an average for each scheme with different random seeds to ensure stability and representativeness.

Abbreviation	Model	Loss	Input sequence	# of run
UM1	U-Net	MSE	1h	5
UM2	U-Net	MSE	2h	5

UM3	U-Net	MSE	3h	5
UM	Ensemble averaging (UM1, UM2, UM3)			
UB1	U-Net	BMSE	1h	5
UB2	U-Net	BMSE	2h	5
UB3	U-Net	BMSE	3h	5
UB	Ensemble averaging (UB1, UB2, UB3)			
U	Ensemble averaging (UM, UC)			
CM1	ConvLSTM	MSE	1h	5
CM2	ConvLSTM	MSE	2h	5
CM3	ConvLSTM	MSE	3h	5
CM	Ensemble averaging (CM1, CM2, CM3)			
CB1	ConvLSTM	BMSE	1h	5
CB2	ConvLSTM	BMSE	2h	5
CB3	ConvLSTM	BMSE	3h	5
CB	Ensemble averaging (CB1, CB2, CB3)			
C	Ensemble averaging (CM, CB)			
P1	pySTEPS	-	1h	1
P2	pySTEPS	-	2h	1
P3	pySTEPS	-	3h	1
P	Ensemble averaging (P1, P2, P3)			
UC	Ensemble averaging (U, C)			
UCP	Ensemble averaging (U, C, P)			

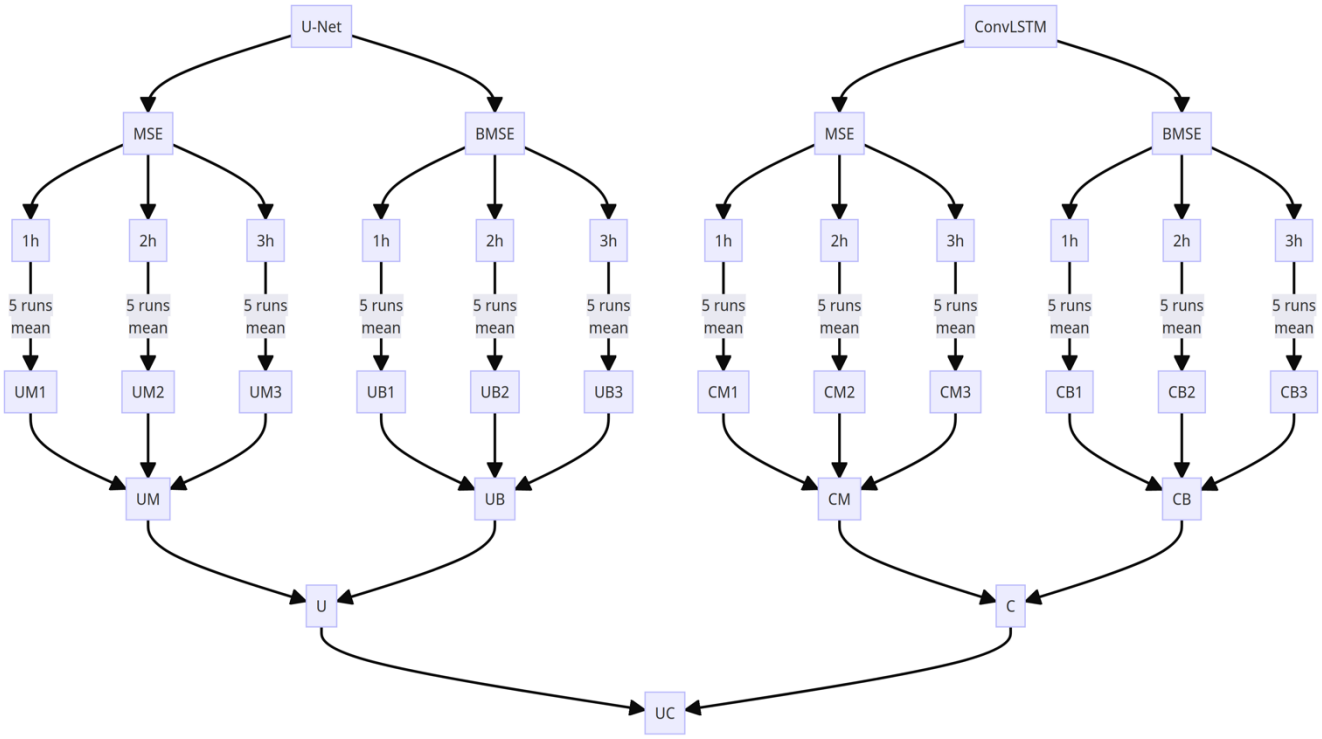

Figure 4. The infographic for overall flowchart for DL models in this study. 'U' stands for U-Net and 'C' stands for ConvLSTM. 'M' and 'B' indicate MSE and BMSE loss functions, respectively. Refer to Table 2 for each scheme.

3.3 Evaluation

Three metrics were used to evaluate model performance: MSE, BMSE, mean bias, and the critical score index (CSI) (Table 3 and Equations 2-5). To exclude clear scenes from the evaluation, we only used scenes in which the number of pixels with precipitation greater than 1 mm/h exceeded 3% of each scene's total number of pixels. A total of 3,801 scenes met this criterion during the evaluation period from 2020 to 2022, and the percentage of precipitation days was 12.73%. We excluded precipitation of less than 1 mm/h to avoid the effects of clear skies and radar noise. Additional thresholds of 5 and 10 mm/h were used to account for moderate and heavy precipitation events.

Moreover, temporal analysis was conducted using summer monsoon rainfall events across the Korean Peninsula in August 2020 to examine the model performance for heavy rainfall phenomena. The summer monsoon rainfall in Korea in 2020 lasted 54 days, from 24 June to 16 August (Lee et al., 2020; Mun et al., 2020). More than 66% of the annual average precipitation during this period fell, with significant regional variation. In particular, 400–600 mm of rainfall fell over the southern part of the Korean Peninsula between 7 and 8 August (Lee et al., 2020; Kim et al., 2020a). Therefore, the specific

evaluation period was set up from 08:00 KST on August 7 to 20:00 KST on August 8, including record-breaking rainfall in the southwestern region.

Table 3. Confusion matrix for given threshold precipitation p.

	Radar	Prediction
Hit	$\geq p$	$\geq p$
Miss	$\geq p$	$< p$
False alarm	$< p$	$\geq p$
Correct negative	$< p$	$< p$

$$mean\ bias = \frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)}{N} \quad (4)$$

where y is the reference value, and $\hat{y}$ represents the predicted value.

$$CSI = \frac{hit}{hit + miss + false\ alarm} \quad (5)$$

4 Results

4.1 Model evaluation using precipitation events for 2020-2022

Tables 4 and 5 present the evaluation results for 1h and 2h predictions respectively, over June, July, and August (JJA) from 2020 to 2022. The *n*-hour persistence model represents a straightforward approach in which the current precipitation is assumed to persist without any change for the next *n* hours. A longer input sequence length generally yields lower MSE and BMSE for all schemes, with specific improvements seen for UM and CM in both prediction lead times. This trend can also be observed with UB and CB and, interestingly, in the pySTEPS non-DL model. P3 demonstrated the lowest MSE and BMSE across all P1-P3 schemes in both prediction periods, suggesting that a longer input sequence can reduce both general capitation (MSE) and high-intensity weighted metrics (BMSE). However, no significant patterns were discernible in other metrics, such as mean bias or CSI, across all schemes with varying input sequence lengths. As the ensemble mean revealed similar results to individual members with different input sequence lengths, subsequent analyses solely focused on aggregated results (i.e., UM, UB, CM, and CB), as illustrated in Figure 5.

Distinct loss functions exert a substantial influence on every metric. Schemes with MSE loss (UM and CM) revealed a lower MSE and higher BMSE compared to BMSE scenarios (UB and CB) at both lead times, an expected outcome given they were optimized for each respective loss function (refer to Tables 4 and 5, and Figure 5). In terms of bias, UM and CM

consistently had negative values at thresholds of 1, 5, and 10 mm/h. While the magnitude of negative BIAS1 in CM was less than UM, it increased at higher thresholds as lead time extended, reaching approximately -15 mm/h for BIAS10 in a 2 h lead time. Conversely, UB and CB generally presented positive biases except for UM with a 2 h lead time in BIAS10 (Figure 5). This suggests that utilizing BMSE bolsters overall intensities by focusing on heavy precipitation, potentially leading to model overestimation. Equally notable is the exacerbation of underestimation in UM and CM at higher thresholds and longer lead times. Consequently, a fusion of both losses could help alleviate issues of underestimation and overestimation, as demonstrated in U and C. Figure 5 reveals that the biases of the U and C approaches are close to zero across all thresholds and lead times after aggregating MSE and BMSE. Zero bias does not inherently signify superior performance. But when looked at with other metrics like MSE or CSI, it seems that the combination of MSE and BMSE successfully reduced severe underestimation and overestimation problems while keeping overall performance the same. For instance, even though U does not exhibit higher CSI across all thresholds and lead times than UM or UB, it provides a more balanced performance. There were insignificant improvements resulting from the ensemble over different input sequence lengths. Additionally, UC or UCP did not show more meaningful improvement than U. This implies that having more ensemble members with different schemes does not guarantee model improvement when the individual members do not create meaningful synergy.

Both MSE and BMSE loss functions yielded better overall metrics with U-Net than ConvLSTM. Comparisons of ensemble means of input sequence length in both 1 and 2 h predictions showed that UM and UB had lower MSE and BMSE and higher CSIs for all thresholds than CM and CB, respectively (Tables 4 and 5, Figure 5). The performance gap between U-Net and ConvLSTM widened with increasing lead time and precipitation intensity. The disparity in bias was particularly larger in ConvLSTM than in U-Net. Moreover, when aggregating all ensemble members for each model, U consistently outperformed C. However, given the variety of possible model configurations, it should be noted that our findings do not suggest that all CNN-RNN models are inherently inferior to solely CNN models for QPN.

Table 4. Quantitative performance over summers of 2020-2022 for the 1 h prediction. Please refer to Table 2 for each scheme. The numbers after metrics indicate the thresholds of precipitation for evaluation. Upward and downward arrows indicate that the metric is better when larger and smaller values are better, respectively.

Scheme	MSE↓	BMSE↓	Bias1	Bias5	Bias10	CSI1↑	CSI5↑	CSI10↑
UM1	10.22	206.43	-0.80	-3.00	-7.93	0.60	0.41	0.22
UM2	9.98	200.15	-0.80	-3.02	-7.96	0.59	0.40	0.22
UM3	**9.73**	192.80	-0.66	-2.75	-7.58	0.59	0.40	0.22
UM	9.97	203.99	-0.80	-3.11	-8.21	**0.61**	**0.42**	0.22
UB1	15.58	151.96	2.62	3.75	3.78	0.51	0.36	0.25
UB2	15.27	148.74	2.52	3.62	3.50	0.49	0.35	0.25
UB3	14.56	150.27	2.22	3.15	2.73	0.48	0.35	0.24
UB	12.87	**144.76**	2.39	3.25	2.44	0.48	0.37	**0.28**
U	10.27	168.64	1.01	0.50	-2.62	0.53	**0.42**	**0.28**
CM1	12.28	251.25	-0.72	-3.87	-11.33	0.51	0.31	0.10
CM2	11.87	242.57	-0.79	-4.09	-11.24	0.51	0.31	0.11
CM3	11.60	235.31	-0.77	-4.14	-11.15	0.51	0.30	0.11
CM	12.09	249.31	-0.80	-4.12	-11.49	0.52	0.32	0.11
CB1	25.52	171.82	4.08	6.15	7.33	0.32	0.23	0.18
CB2	24.67	165.02	3.58	5.99	7.12	0.29	0.23	0.18
CB3	25.57	159.65	3.88	6.25	7.59	0.31	0.23	0.17
CB	24.79	168.68	3.89	6.03	7.11	0.31	0.24	0.18
C	14.79	196.83	2.06	2.42	-0.36	0.37	0.30	0.20
P1	19.18	291.18	-0.19	-0.33	-0.62	0.52	0.35	0.22
P2	18.90	284.38	-0.16	-0.29	-0.55	0.52	0.34	0.22
P3	18.73	279.51	-0.14	-0.26	-0.51	0.52	0.34	0.21
P	16.18	252.12	-0.34	-0.76	-1.41	0.47	0.31	0.19
UC	11.72	180.08	1.52	1.21	-2.34	0.43	0.37	0.25
UCP	11.04	191.04	0.87	-0.08	-4.22	0.44	0.38	0.26
Persistence	21.61	267.60	0.15	0.34	0.55	0.47	0.27	0.16

Table 5. Quantitative performance over summers of 2020-2022 for the 2 h prediction. Please refer to Table 2 for each scheme. The numbers after metrics indicate the thresholds of precipitation for evaluation. Upward and downward arrows indicate that the metric is better when larger and smaller values are better, respectively.

Scheme	MSE↓	BMSE↓	Bias1	Bias5	Bias10	CSI1↑	CSI5↑	CSI10↑
UM1	12.62	257.45	-1.75	-5.63	-13.00	0.49	0.23	0.07
UM2	12.29	249.03	-1.75	-5.59	-12.91	0.48	0.23	0.07
UM3	**12.03**	242.30	-1.63	-5.35	-12.52	0.48	0.23	0.07
UM	12.27	254.84	-1.78	-6.09	-13.69	**0.51**	0.24	0.06
UB1	18.04	208.84	1.80	2.21	0.61	0.38	0.26	0.14
UB2	19.07	196.56	2.31	3.25	2.66	0.39	0.26	0.15
UB3	18.52	197.14	2.12	2.90	1.95	0.39	0.25	0.14
UB	14.83	192.23	1.93	2.30	-0.36	0.37	0.29	**0.17**
U	12.20	218.85	0.50	-1.25	-7.61	0.44	**0.30**	0.13
C1M	13.90	281.72	-1.00	-7.06	-16.73	0.40	0.12	0.00
C2M	13.50	273.02	-1.05	-7.06	-16.53	0.41	0.12	0.00
C3M	13.23	266.75	-1.04	-7.06	-15.95	0.41	0.12	0.01
CM	13.82	281.01	-1.09	-7.22	-16.55	0.41	0.12	0.01
C1B	29.79	198.82	3.05	6.04	6.82	0.16	0.17	0.12
C2B	29.04	190.89	3.01	6.00	6.91	0.16	0.18	0.12
C3B	29.74	**185.57**	3.19	6.23	7.34	0.17	0.17	0.12
CB	29.23	196.31	3.01	5.98	6.78	0.16	0.18	0.13
C	16.99	227.21	1.92	2.16	-3.46	0.27	0.22	0.10
P1	24.33	337.08	-0.31	-0.46	-0.71	0.39	0.21	0.11
P2	23.87	327.89	-0.27	-0.40	-0.62	0.38	0.21	0.11
P3	23.58	321.85	-0.24	-0.37	-0.59	0.38	0.20	0.11
P	19.41	288.13	-0.52	-1.10	-2.03	0.33	0.18	0.10
UC	13.64	220.93	1.30	0.18	-7.40	0.33	0.28	0.11
UCP	13.26	240.00	0.49	-1.69	-9.33	0.35	0.27	0.11
Persistence	25.39	288.91	0.25	0.60	1.03	0.37	0.19	0.10

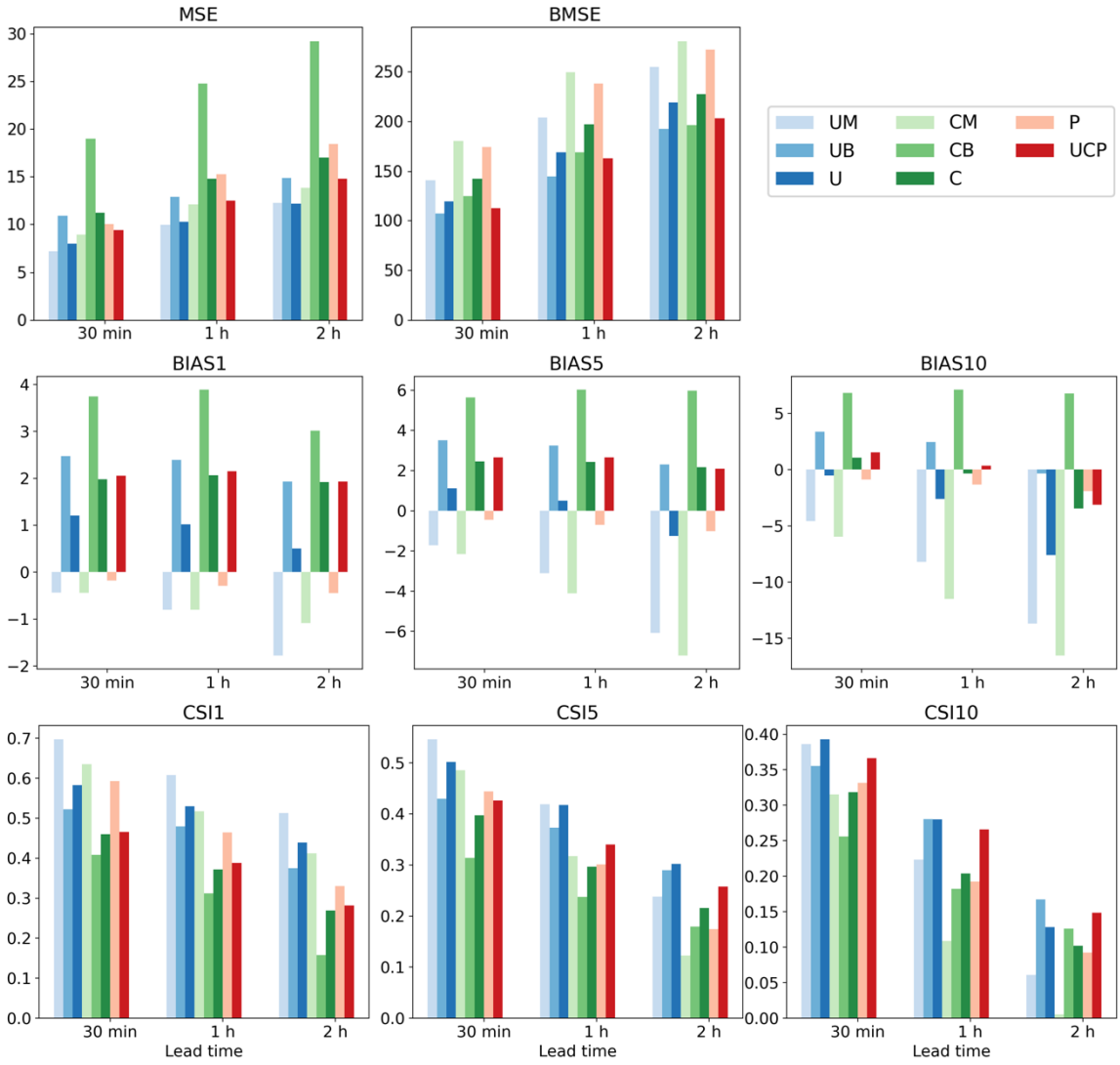

Figure 5. Quantitative performance over summers of 2020-2022 of lead times of 30 min, 1 h, and 2 h. Please refer to Table 2 for each scheme. The numbers after metrics indicate the thresholds of precipitation for evaluation.

4.2 Evaluation on heavy rainfall events

Figures 6 and 7 present a time-series analysis of CSI1 and CSI10, respectively, for a heavy rainfall event occurring from August 7 to 9, 2020. It is important to note that the performance of all models, including persistence, was influenced by the rate of rain pixels in each scene (Figure 6). Consequently, the performance of QPN should be interpreted in terms of the precipitation rate because the likelihood of achieving a correct prediction (i.e., CSI) increases as the area of precipitation expands (Han et al., 2023). As illustrated in Figure 6, UM achieved the highest CSI1, whereas CB registered the poorest performance across all lead times. Echoing the long-term evaluation results from the summers of 2020-2022, UM and CM

significantly outperformed UB and CB for the 1 mm/h threshold, yielding higher CSI values. U and C, the ensembles of MSE and BMSE for each deep learning model, demonstrated CSI1 performance roughly equivalent to the average of UM and UB and CM and CB, respectively. At the 10 mm/h threshold, however, UM and CM's performance declined. Although UM retained competitive performance in the 30 min forecast, its skill score plummeted to the lowest level as CM in the 2 h prediction (Figure 7). UB and CB achieved higher CSI10 than UM and CM in both 1 and 2 h predictions, implying that the benefits of using BMSE increased with longer lead times, a pattern consistent with the long-term evaluation in Figure 5. As the CSI10 performances of UB and U were nearly identical in 30 min and 1 h predictions, U's performance in heavy precipitation seemed to be largely dictated by UB. Despite the overestimation caused by the use of BMSE, U's CSI10 did not degrade, even with the average of weaker precipitation from UM whose loss was MSE. However, due to the severe underestimation issue with UM, U's CSI10 ultimately degraded relative to UB in the 2 h prediction (Figure 7). This pattern also aligns with Figure 5.

Figure 8 provides a comparative map for each scheme's 2 h prediction for 05:00 on August 8, 2020, in KST. To demonstrate the sensitivity to random seeds, predictions with each seed were included. Stability over random seeds increased more significantly with U-Net than with Conv-LSTM and with MSE than BMSE, in terms of both quantitative metrics (i.e., CSI1 and CSI10) and qualitative visual interpretation. Generally, UM and CM tended to underestimate both the area and intensity of precipitation, while UB and CB generated higher intensity over a larger area. UM exhibited the best performance in CSI1, suggesting its proficiency at forecasting the overall precipitation area. In terms of CSI10, UB and CB outperformed the others, indicating the effectiveness of BMSE in forecasting areas of heavy precipitation. Although pySTEPS appeared to closely mimic actual radar observations, the location of the forecast was simulated faster (more eastward) than the actual position. Moreover, no forecast results were available for the western part of the map because its prediction was based on a calculated wind vector. In contrast, some deep learning models simulated rainfall over the West Sea of Korea (black circle in the Radar observation map), indicating the superior capabilities of these models in forecasting areas of precipitation. In terms of the 99th percentile (P99%) precipitation intensity for each scene in Figure 8, the results from the ensemble generally show a decreased P99%. Even U showed a lower P99% than UM, despite being ensembled with UB. As Franch et al. (2020) pointed out, the ensemble might attenuate peak intensity. All deep learning models produced blurred predictions compared to pySTEPS or radar observations, showing the common limitation which is discussed in Section 5.2.

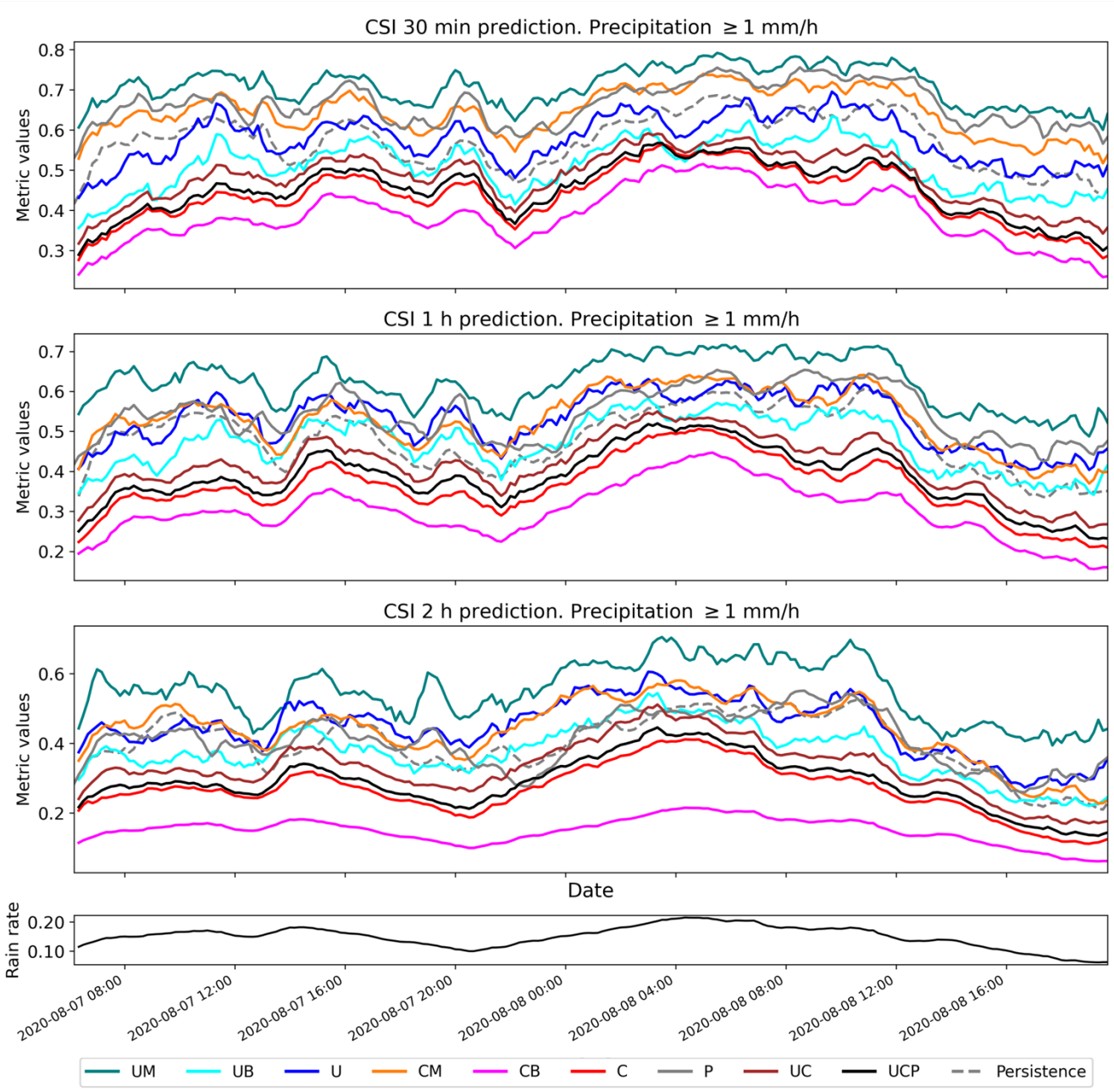

Figure 6. Comparison of CSI performance for the case of heavy rainfall over South Korea from 7th to 8th August 2020 with the 1 mm/h threshold. Refer to Table 2 for scheme names. The bottom black line represents the ratio of precipitation pixels > 1 mm/h for each radar scene.

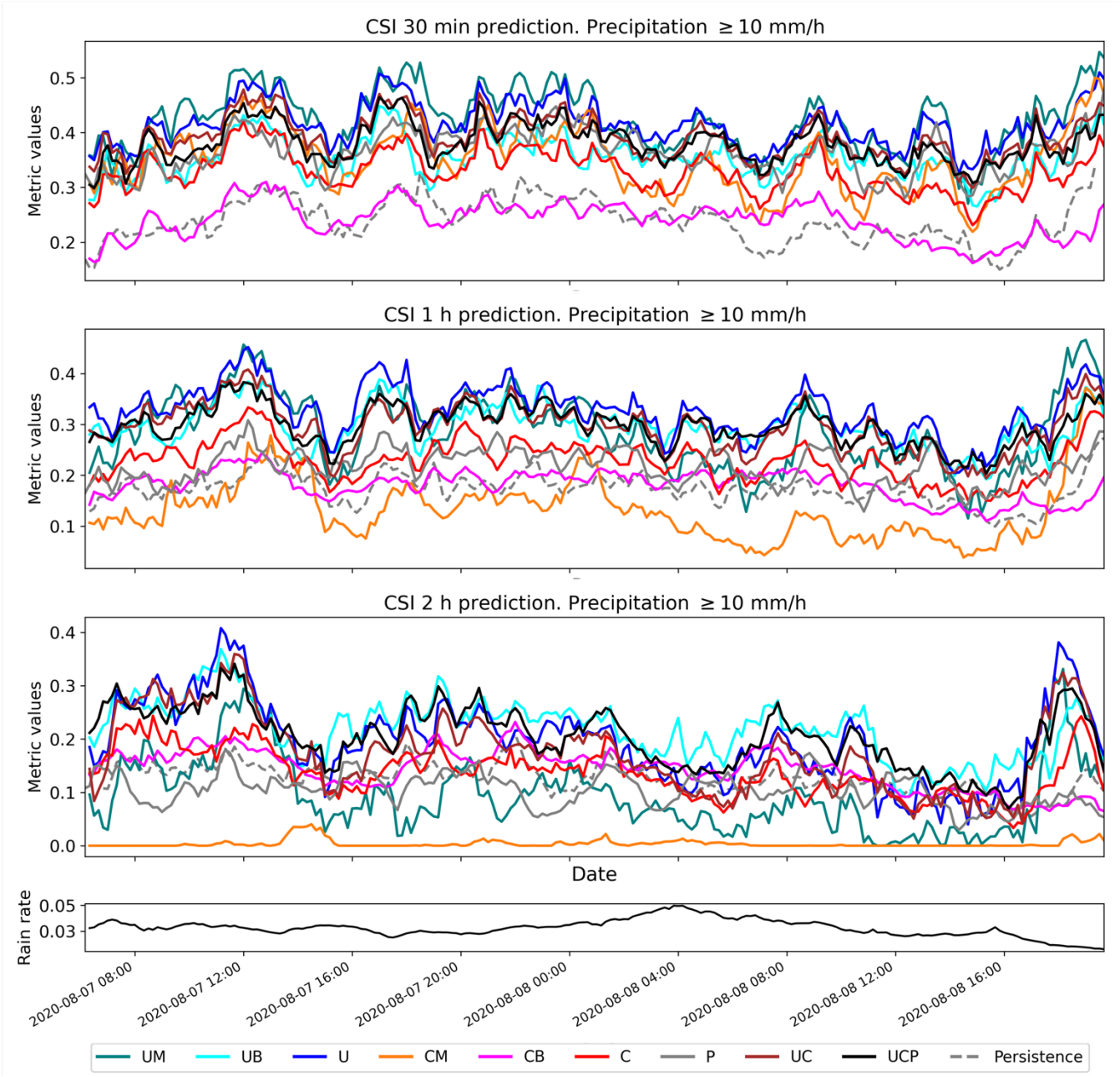

Figure 7. Comparison of CSI performance for the case of heavy rainfall over South Korea from 7th to 8th August 2020 with the 10 mm/h threshold. Refer to Table 2 for scheme names. The bottom black line represents the ratio of precipitation pixels > 10 mm/h for each radar scene.

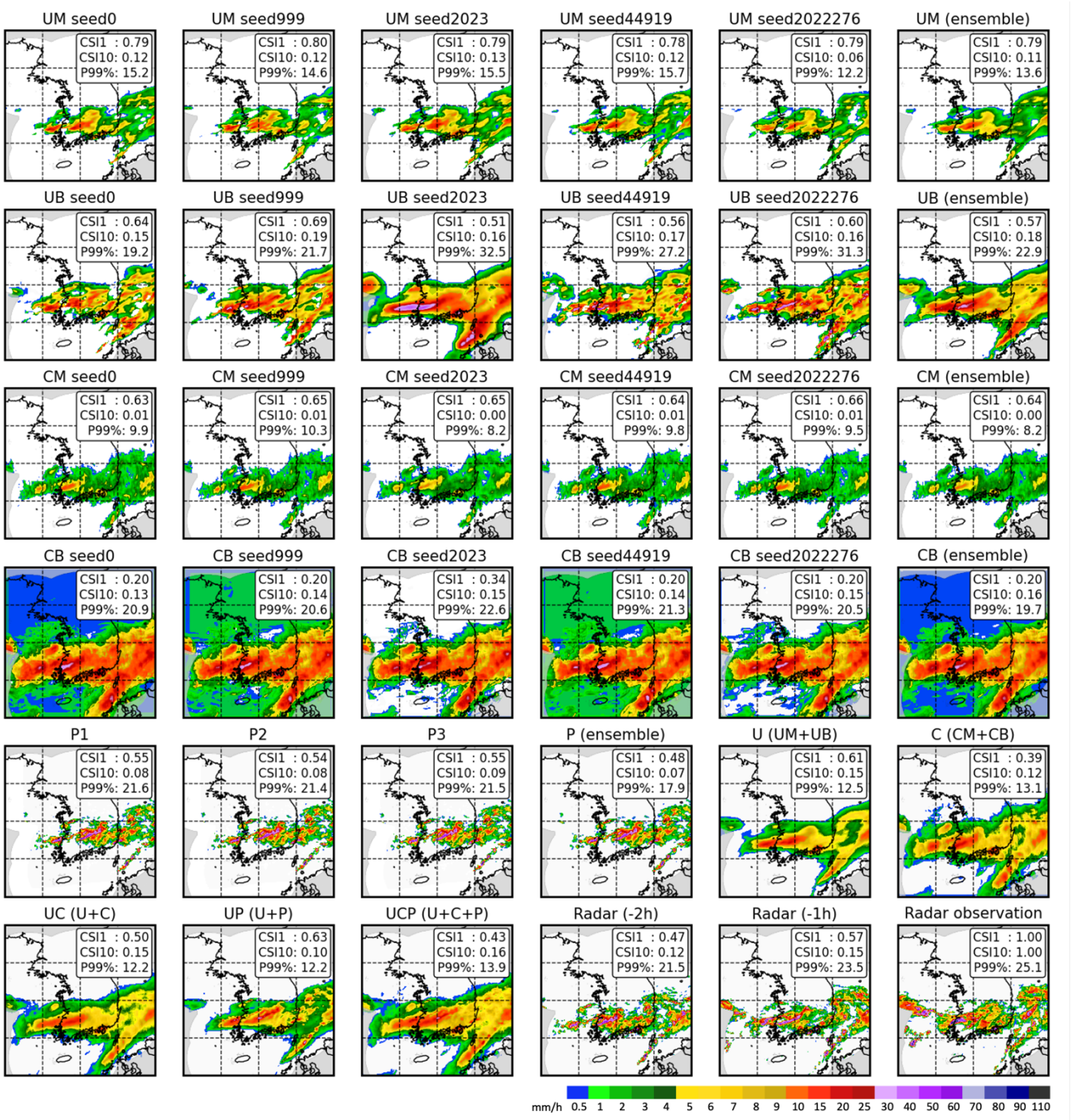

Figure 8. Comparison map for 05:00 on August 8, 2020, in KST with a 2h lead time. Refer to Table 2 for scheme names. Schemes with seed numbers are the averages of results for three scenes using 1h, 2h, and 3h input sequences. CSI1 and CSI10 indicate the CSI scores with thresholds of 1 and 10 mm/h, respectively.

380

5 Discussion

5.1 Performance comparison and considerations of key factors

To address data imbalance and improve skill scores, various loss functions have been considered in previous research. Our comparison of two representative losses, MSE and BMSE, revealed that each has its strengths and weaknesses. The selection of an appropriate loss function should be informed by a comprehensive evaluation of QPN results. Optimal loss functions may vary depending on the specific objectives as it provides guidance to DL modeling. For instance, if the model's focus is on severe weather, BMSE can be weighted to emphasize high intensities. In cases where the area of precipitation over a certain threshold is of key interest, a modified CSI loss can be used (Ko et al., 2022). As a single metric cannot fully evaluate a model, combinations of different losses can also be explored. Alternatively, the ensemble approach analyzed in our study can leverage different loss functions to create a synergistic effect for QPN.

In this study, U-Net consistently outperformed ConvLSTM in various respects, both in long-term evaluation and in a single heavy rainfall event. This finding is in line with previous research (Ayzel et al., 2020; Ko et al., 2022; Han et al., 2023). Additionally, U-Net demonstrated more stability across different random seeds than ConvLSTM (Figure 8). Contrary to the widespread expectation that DL models powered by RNN would excel in time-series forecasting, it was found that a model relying solely on CNN can perform better. However, this does not imply that all models using RNN structures are inferior to full CNN models. Considering the wide range of U-Net and ConvLSTM variants, there could be potential for RNN-powered models to exhibit superior results. Lastly, the input sequence length did not significantly impact the results compared to other factors in this study. Nevertheless, sequence length should still be carefully considered as DL-QPN relies significantly on past information. In other DL models and QPN designs, input sequence length may have a greater impact than it did in this study, therefore, we continue to regard this as a key factor in DL-QPN.

Due to the inherent randomness and stochastic nature of deep learning, modeling and evaluation need to be carefully conducted, taking into account relevant factors. As demonstrated in Figure 8, results can vary for each run with a different random seed. Thus, stability should be a priority when developing a DL-QPN model, a point often overlooked in previous studies. By treating each run as an ensemble member, we can avoid unstable results under varying conditions of randomness.

5.2 Common drawbacks of DL-QPN

In this study, all DL-QPN schemes demonstrated a dwindling intensity problem as the lead time increased for both the long-term experiment and the heavy rainfall event. Previous studies have also reported deformation and significant blurring effects in DL-QPN models (Ayzel et al., 2020; Shi et al., 2015; Trebing et al., 2021). Ravuri et al. (2021) introduced Deep Generative Models of Rainfall (DGMR) to provide realistic rainfall prediction maps. While DGMR reduced the blurring effect by leveraging GAN, it did not perform better than U-Net in terms of CSI (Ko et al., 2022).

Regarding the typical limitations of DL-QPN, the following two factors may play a significant role: (1) the uneven distribution of precipitation and the sparsity of precipitation events and (2) the dynamic movement of the atmosphere. A substantial issue with DL-QPN is the skewed distribution of precipitation towards weak intensities (Shi et al., 2017; Chen and Wang, 2022; Adewoyin et al., 2021). Even after rejecting radar scenes when precipitation event were too weak during the sampling phase, the pixel-level distribution remained skewed towards the weak range. The sparsity of precipitation events is also strongly related to data imbalances. Because most pixels have no radar signal, the background of the weather radar mainly consists of zero values. This sparsity distinguishes the DL-QPN from general video prediction and makes the model susceptible to the prediction of underestimated results. Data augmentation or patch-level sampling can be used during the sampling phase to reduce data imbalance and sparsity. Some loss functions have also been used to solve this skewed distribution during the model training phase (Ravuri et al., 2021; Shi et al., 2017; Ko et al., 2022).

Because the DL-QPN deals with very dynamic atmospheric data, the mismatched position between past and future sequences can result in significantly degraded performance as the forecasting time increases. The DL-QPN does not explicitly learn the movement of the precipitation cells. Because convolutional filters in a single layer cannot link remote information beyond the kernel size, multiple layers gradually extend the receptive field of interest. However, even with multiple layers, the model may fail to simulate a future precipitation cell whose position is too far from its origin. This limitation might be improved by the attention techniques which enable the model to identify high relationships between input and prediction over a global area.

Although recent studies on DL-QPN have reported that they achieve better performance than traditional models with end-to-end learning, it is crucial to investigate how to fully exploit precipitation characteristics in DL-QPN. The properties of each precipitation cell can be explicitly fed into the model to successfully simulate the relationship between the precipitation cells across different time steps. In a similar context, there is ample room to contribute to radar-based DL-QPN using additional input variables, such as atmospheric instability indices, temperature, vertical humidity profiles, wind vectors, and NWP-predicted precipitation. Although some previous studies have attempted to fuse heterogeneous datasets for DL-QPN (Zhang et al., 2021a; Adewoyin et al., 2021; Bouget et al., 2021), further research is required to determine their contribution and develop a synergetic model to maximize the multimodal dataset, instead of just stacking the variables as input features.

5.3 Other factors to consider

In addition to the four key factors investigated, there are other factors that could be considered. First, there are different prediction designs other than the multi-to-multi approach used in this study: Three representative approaches are summarized from the literature (Table 2 and Figure 9). In a single prediction, each model individually predicts each time step. Because of its simplicity and good performance focusing on a single lead time, image-to-image DL-QPN with radar (Chen et al., 2020) and image-to-point QPN with radar (Kim et al., 2020b) adopt this method. A single prediction employs *n*

distinct models for *n* time steps (Figure 9a). As shown in Figure 9b, recursive prediction only considers the next time step. The predicted output of the first future time step was fed into an input sequence to predict the next time step, and this process was repeated iteratively to forecast longer lead times. Ayzel et al. (2020) proposed RainNet v1.0, with a recursive approach, using the previous six sequences to forecast up to 12 future sequences with an interval of 5 min. Because the recursive model only considers the next time step, it is expected to yield accurate predictions. However, the primary disadvantage of this method is the accumulation of errors with increasing lead time because more predicted results with uncertainty are used as input data. In the multiple prediction, a model simultaneously forecasts all future sequences. This design has been widely adopted for DL-QPN using a weather radar (Kim and Hong, 2021; Ravuri et al., 2021; Shi et al., 2015; Shi et al., 2017; Franch et al., 2020; Han et al., 2023). A multiple prediction model can generate *n* time steps, as shown in Figure 9c. Because a multiple prediction model is calibrated for various lead times by minimizing the overall loss, its performance may be degraded for a particular lead time. Although only multiple prediction was evaluated in this study due to its popularity, other approaches may also be considered based on the specific goal of each model. Additionally, there is room for the aggregation of different prediction designs to create synergy among them.

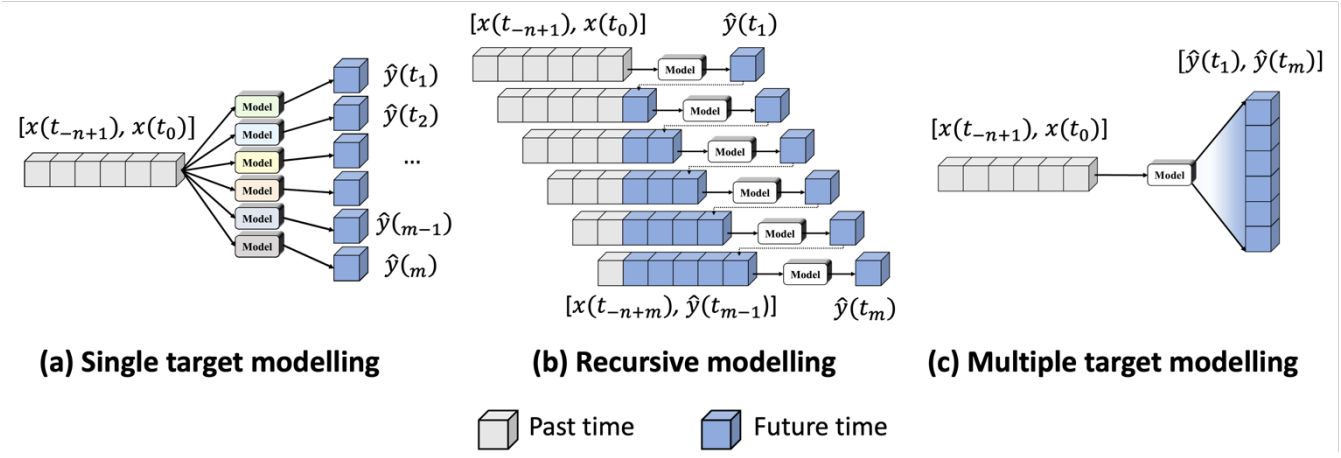

Figure 9. Three forecasting designs in DL-QPN with radar sequence. (a) Single, (b) recursive, and (c) multiple predictions.

As precipitation is calculated from radar reflectivity, direct prediction of the original signal can also be considered. Some previous studies utilized radar reflectivity in DL-QPN (Bonnet et al., 2020; Lepetit et al., 2022; Albu et al., 2022; Han et al., 2022). To our knowledge, there has been few studies comparing radar reflectivity and precipitation intensity directly in DL-QPN. In this study, we chose to forecast precipitation intensity because our final interest is in the strength of the precipitation. However, as the precipitation intensity can be converted from predicted reflectivity, further investigation is needed in the future to find a better skill score.

5.4 Novelty and limitations

This study conducted a comprehensive comparison of the DL-QPN. The major novelties of this study are summarized as follows: First, we categorized and investigated four critical factors of DL-QPN. While most previous DL-QPN studies have focused on DL models, there has been a dearth of research on other factors to be considered. Second, a long-term evaluation was conducted for three years during the summer. This long-term evaluation provided helpful information from an operational perspective of DL-QPN. Randomness in training phase such as initialization of weights was addressed for the first time in this study. Stability is one of the most important aspects of the operational model for disaster forecasting; however, this was rarely considered in previous studies. By running each DL scheme five times with different random seeds, we could examine the stability of each scheme as well as test the effect of ensemble aggregation. The ensemble with different DL-QPN schemes, and even with a non-DL model was also newly conducted in this study, showing the potential of the ensemble approach. With advanced approaches to adjusting the optimal weights for each ensemble member, such as stacking ensemble (Cho et al., 2020), further improvements are expected. Lastly, we summarize the common drawbacks of DL-QPN and discuss their possible causes.

Despite the innovations in investigating key factors in DL-QPN, several limitations remain. One of the significant limitations of this study is the lack of investigation of more loss functions beyond MSE and BMSE. In addition to L1 and L2 losses, several loss functions have been suggested for DL-QPN, such as logcosh (Ayzel et al., 2020; Cuomo and Chandrasekar, 2021) or adversarial loss using GAN (Ravuri et al., 2021; Kim and Hong, 2021). Although the loss function in the DL-QPN is of the utmost importance, it is beyond the scope of this study, as there are too many aspects to cover in a single paper. Another limitation is the small study area. To generalize the research findings, it would be ideal for examining several study areas with different environments if possible. Some recent studies have attempted to extend their study areas using different radar data (Ravuri et al., 2021; Zhang et al., 2023). Consequently, it is highly expected to evaluate models over multiple study areas in the future to increase operational generalizations. As DL-QPN has garnered significant attention in recent years, advanced models have been continuously suggested, including GANs (Ravuri et al., 2021; Choi and Kim, 2022; Kim and Hong, 2022), latent diffusion (Leinonen et al., 2023), Transformer (Franch et al., 2023), simpler yet better video prediction (Han et al., 2023), and NowcastNet (Zhang et al., 2023). Despite the importance of using state-of-the-art (SOTA) models in DL-QPN, at this moment they are out of the scope of this paper due to their complexity, limitations of implementation, or issues of code availability. A comprehensive benchmark for different SOTA models is anticipated in our future studies, and if possible, determining their optimal ensemble for further improvement will also be one of our future areas of investigation.

6 Conclusion

This paper summarizes and compares the effects of the four critical factors in the DL-QPN. As previous studies mainly focused on developing DL models with less investigation of other considerations, we expect this study to contribute to future DL-QPN studies by drawing attention to other essential factors. We evaluated various DL schemes, considering the deep learning model, input sequence length, loss function, and ensemble method. Through our quantitative and qualitative comparisons, we found that the U-Net model with the MSE loss function appeared to be the optimal combination for weak precipitation prediction in this study, while the BMSE was effective for heavy precipitation (≥ 10 mm/h). In general, there was a weak tendency for longer past sequence lengths to yield lower MSE and BMSE for both DL models, but there was little difference in terms of the CSI. After running experiments five times for each scheme, we determined that U-Net and MSE were more stable than ConvLSTM and BMSE in terms of randomness. The aggregation of different schemes resulted in balanced skill scores across multiple metrics, especially when MSE and BMSE were combined. Common issues with DL-QPN include underestimation as the lead time gets longer and the production of smoothed spatial patterns. These drawbacks are likely due to the skewed distribution of weak intensity and the sparsity of precipitation events. They are expected to be mitigated with the implementation of an improved sampling strategy and the use of various loss functions beyond MSE or BMSE.

While our study provided a comprehensive comparison of key factors in DL-QPN, it also underscored the need for continued exploration. Areas for future research include investigating more advanced loss functions, broadening the study areas to diverse environments for greater generalization, and comparing SOTA models. Furthermore, the potential of ensemble approaches for further enhancing the performance of DL-QPN models offers promising opportunities for future investigation. Ultimately, our study points out the importance of looking at DL-QPN as a whole, not just focusing on creating DL models, but also recognizing the need to consider many factors that affect precipitation nowcasting.

Code and data availability

The original RainNet v1.0 is an open source code provided by Ayzel et al. (2020) via GitHub (https://github.com/hydrogo/rainnet; last access: 11 November 2022). Ground weather radar and MAPLE over South Korea are available at the KMA radar center (http://radar.kma.go.kr; last access: 11 November 2022) and Korea Public Data Portal (https://www.data.go.kr/en/data/15068574/fileData.do, last access: 11 November 2022) upon request. Model structures, trained models, and validation datasets over heavy rainfall events in August 2020 can be found at https://doi.org/10.5281/zenodo.7312779 (Han, 2022).

Author Contributions

DH and YS initiated and led this study, conducted experiments and evaluations, and wrote the original article. DH conducted experiments for the revision, updated the revised manuscript, and wrote the responses to reviewers' comments. JI and JL reviewed and edited the original and revised manuscript. All authors contributed to designing the methodology. All authors analysed and discussed the results. JI supervised this study and acquired funding.

Competing interests

The contact author has declared that none of the authors has any competing interests.

Financial support

This research was supported by the National Research Foundation of Korea under Grant NRF-2021R1A2C2008561, by the Korea Meteorological Administration Research and Development Program under Grant KMIPA 2017-7010, and Korea Institute of Marine Science & Technology Promotion (KIMST) funded by the Ministry of Oceans and Fisheries, Korea (RS-2023-00256330; RS-2023-00238486).

References

Adewoyin, R. A., Dueben, P., Watson, P., He, Y., and Dutta, R.: TRU-NET: a deep learning approach to high resolution prediction of rainfall, Machine Learning, 110, 2035-2062, https://doi.org/10.1007/s10994-021-06022-6, 2021.

Agrawal, S., Barrington, L., Bromberg, C., Burge, J., Gazen, C., and Hickey, J.: Machine learning for precipitation nowcasting from radar images, arXiv preprint arXiv:1912.12132, https://doi.org/10.48550/arXiv.1912.12132, 2019.

Ayzel, G., Scheffer, T., and Heistermann, M.: RainNet v1. 0: a convolutional neural network for radar-based precipitation nowcasting, Geoscientific Model Development, 13, 2631-2644, https://doi.org/10.5194/gmd-13-2631-2020, 2020.

Barth, A., Alvera-Azcárate, A., Troupin, C., and Beckers, J.-M.: DINCAE 2.0: multivariate convolutional neural network with error estimates to reconstruct sea surface temperature satellite and altimetry observations, Geoscientific Model Development, 15, 2183-2196, https://doi.org/10.5194/gmd-15-2183-2022, 2022.

Bouget, V., Béréziat, D., Brajard, J., Charantonis, A., and Filoche, A.: Fusion of rain radar images and wind forecasts in a deep learning model applied to rain nowcasting, Remote Sensing, 13, 246, https://doi.org/10.3390/rs13020246, 2021.

Bowler, N. E., Pierce, C. E., and Seed, A. W.: STEPS: A probabilistic precipitation forecasting scheme which merges an extrapolation nowcast with downscaled NWP, Quarterly Journal of the Royal Meteorological Society: A journal of the atmospheric sciences, applied meteorology and physical oceanography, 132, 2127-2155, https://doi.org/10.1256/qj.04.100, 2006.

Chattopadhyay, A., Mustafa, M., Hassanzadeh, P., Bach, E., and Kashinath, K.: Towards physics-inspired data-driven weather forecasting: integrating data assimilation with a deep spatial-transformer-based U-NET in a case study with ERA5, Geoscientific Model Development, 15, 2221-2237, https://doi.org/10.5194/gmd-15-2221-2022, 2022.

Chen, G. and Wang, W. C.: Short-term precipitation prediction for contiguous United States using deep learning, Geophysical Research Letters, e2022GL097904, https://doi.org/10.1029/2022GL097904, 2022.

Chen, L., Cao, Y., Ma, L., and Zhang, J.: A deep learning-based methodology for precipitation nowcasting with radar, Earth and Space Science, 7, e2019EA000812, https://doi.org/10.1029/2019EA000812, 2020.

Chinita, M. J., Witte, M., Kurowski, M. J., Teixeira, J., Suselj, K., Matheou, G., and Bogenschutz, P.: Improving the representation of shallow cumulus convection with the Simplified Higher-Order Closure Mass-Flux (SHOC+ MF v1. 0) approach, Geoscientific Model Development Discussions, 1-23, https://doi.org/10.5194/gmd-2022-162, 2022.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y.: Learning phrase representations using RNN encoder-decoder for statistical machine translation, arXiv preprint arXiv:1406.1078, https://doi.org/10.3115/v1/D14-1179, 2014.

Chung, K.-S. and Yao, I.-A.: Improving radar echo Lagrangian extrapolation nowcasting by blending numerical model wind information: Statistical performance of 16 typhoon cases, Monthly Weather Review, 148, 1099-1120, https://doi.org/10.1175/MWR-D-19-0193.1, 2020.

Cuomo, J. and Chandrasekar, V.: Use of Deep Learning for Weather Radar Nowcasting, Journal of Atmospheric and Oceanic Technology, 38, 1641-1656, https://doi.org/10.1175/JTECH-D-21-0012.1, 2021.

Dixon, M. and Wiener, G.: TITAN: Thunderstorm identification, tracking, analysis, and nowcasting—A radar-based methodology, Journal of atmospheric and oceanic technology, 10, 785-797, https://doi.org/10.1175/1520-0426(1993)010<0785:TTITAA>2.0.CO;2, 1993.

Franch, G., Nerini, D., Pendesini, M., Coviello, L., Jurman, G., and Furlanello, C.: Precipitation nowcasting with orographic enhanced stacked generalization: Improving deep learning predictions on extreme events, Atmosphere, 11, 267, https://doi.org/10.3390/atmos11030267, 2020.

Gardoll, S. and Boucher, O.: Classification of tropical cyclone containing images using a convolutional neural network: performance and sensitivity to the learning dataset, EGUsphere, 1-29, https://doi.org/10.5194/egusphere-2022-147, 2022.

Geiss, A., Silva, S. J., and Hardin, J. C.: Downscaling atmospheric chemistry simulations with physically consistent deep learning, Geoscientific Model Development, 15, 6677-6694, https://doi.org/10.5194/gmd-15-6677-2022, 2022.

Germann, U. and Zawadzki, I.: Scale-dependence of the predictability of precipitation from continental radar images. Part I: Description of the methodology, Monthly Weather Review, 130, 2859-2873, https://doi.org/10.1175/1520-0493(2002)130<2859:SDOTPO>2.0.CO;2, 2002.

Germann, U. and Zawadzki, I.: Scale dependence of the predictability of precipitation from continental radar images. Part II: Probability forecasts, Journal of Applied Meteorology, 43, 74-89, https://doi.org/10.1175/1520-0450(2004)043<0074:SDOTPO>2.0.CO;2, 2004.

Haiden, T., Kann, A., Wittmann, C., Pistotnik, G., Bica, B., and Gruber, C.: The Integrated Nowcasting through Comprehensive Analysis (INCA) system and its validation over the Eastern Alpine region, Weather and Forecasting, 26, 166-183, https://doi.org/10.1175/2010WAF2222451.1, 2011.

Han, D.: Supplementary code and data: 'Key factors for quantitative precipitati on nowcasting using ground weather radar data based on deep learning, Zenodo, https://doi.org/10.5281/zenodo.7312779, 2022.

Han, D., Lee, J., Im, J., Sim, S., Lee, S., and Han, H.: A novel framework of detecting convective initiation combining automated sampling, machine learning, and repeated model tuning from geostationary satellite data, Remote Sensing, 11, 1454, https://doi.org/10.3390/rs11121454, 2019.

Hochreiter, S. and Schmidhuber, J.: Long short-term memory, Neural computation, 9, 1735-1780, https://doi.org/10.1162/neco.1997.9.8.1735, 1997.

Jeong, C. H., Kim, W., Joo, W., Jang, D., and Yi, M. Y.: Enhancing the encoding-forecasting model for precipitation nowcasting by putting high emphasis on the latest data of the time step, Atmosphere, 12, 261, https://doi.org/10.3390/atmos12020261, 2021.

Kang, J., Wang, H., Yuan, F., Wang, Z., Huang, J., and Qiu, T.: Prediction of precipitation based on recurrent neural networks in Jingdezhen, Jiangxi Province, China, Atmosphere, 11, 246, https://doi.org/10.3390/atmos11030246, 2020.

Kim, D.-K., Suezawa, T., Mega, T., Kikuchi, H., Yoshikawa, E., Baron, P., and Ushio, T.: Improving precipitation nowcasting using a three-dimensional convolutional neural network model from Multi Parameter Phased Array Weather Radar observations, Atmospheric Research, 262, 105774, https://doi.org/10.1016/j.atmosres.2021.105774, 2021.

Kim, Y. and Hong, S.: Very Short-Term Rainfall Prediction Using Ground Radar Observations and Conditional Generative Adversarial Networks, IEEE Transactions on Geoscience and Remote Sensing, https://doi.org/10.1109/TGRS.2021.3108812, 2021.

Kim, Y.-T., Park, M., and Kwon, H.-H.: Spatio-temporal summer rainfall pattern in 2020 from a rainfall frequency perspective, Journal of Korean Society of Disaster and Security, 13, 93-104, https://doi.org/10.21729/ksds.2020.13.4.93, 2020a.

Kim, Y. J., Kim, H.-C., Han, D., Lee, S., and Im, J.: Prediction of monthly Arctic sea ice concentrations using satellite and reanalysis data based on convolutional neural networks, The Cryosphere, 14, 1083-1104, https://doi.org/10.5194/tc-14-1083-2020, 2020b.

Kingma, D. P. and Ba, J.: Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980, https://doi.org/10.48550/arXiv.1412.6980, 2014.

Ko, J., Lee, K., Hwang, H., Oh, S.-G., Son, S.-W., and Shin, K.: Effective training strategies for deep-learning-based precipitation nowcasting and estimation, Computers & Geosciences, 161, 105072, https://doi.org/10.1016/j.cageo.2022.105072, 2022.

Lee, J., Kim, M., Im, J., Han, H., and Han, D.: Pre-trained feature aggregated deep learning-based monitoring of overshooting tops using multi-spectral channels of GeoKompsat-2A advanced meteorological imagery, GIScience & Remote Sensing, 58, 1052-1071, https://doi.org/10.1080/15481603.2021.1960075, 2021.

Lee, S., Lee, M., and Kang, H.: 2020 년 홍수 현황과 항구적 대책 방향, 2020.

Liu, J., Xu, L., and Chen, N.: A spatiotemporal deep learning model ST-LSTM-SA for hourly rainfall forecasting using radar echo images, Journal of Hydrology, 127748, https://doi.org/10.1016/j.jhydrol.2022.127748, 2022.

Lu, Y., James, T., Schillaci, C., and Lipani, A.: Snow detection in alpine regions with Convolutional Neural Networks: discriminating snow from cold clouds and water body, GIScience & Remote Sensing, 59, 1321-1343, https://doi.org/10.1080/15481603.2022.2112391, 2022.

Marshall, J. S. and Palmer, W. K. M.: The distribution of raindrops with size, J. meteor., 5, 165-166, https://doi.org/10.1175/1520-0469(1948)005<0165:TDORWS>2.0.CO;2, 1948.

Mecklenburg, S., Joss, J., and Schmid, W.: Improving the nowcasting of precipitation in an Alpine region with an enhanced radar echo tracking algorithm, Journal of Hydrology, 239, 46-68, https://doi.org/10.1016/S0022-1694(00)00352-8, 2000.

Mun, J.-J., Gang, S.-U., and Lee, J.-J.: 2020 년 홍수의 전국 강수 분석, Water for future, 53, 135-143, 2020.

Prudden, R., Adams, S., Kangin, D., Robinson, N., Ravuri, S., Mohamed, S., and Arribas, A.: A review of radar-based nowcasting of precipitation and applicable machine learning techniques, arXiv preprint arXiv:2005.04988, https://doi.org/10.48550/arXiv.2005.04988, 2020.

Ravuri, S., Lenc, K., Willson, M., Kangin, D., Lam, R., Mirowski, P., Fitzsimons, M., Athanassiadou, M., Kashem, S., and Madge, S.: Skilful precipitation nowcasting using deep generative models of radar, Nature, 597, 672-677, https://doi.org/10.1038/s41586-021-03854-z, 2021.

Ronneberger, O., Fischer, P., and Brox, T.: U-net: Convolutional networks for biomedical image segmentation, International Conference on Medical image computing and computer-assisted intervention, 234-241, https://doi.org/10.1007/978-3-319-24574-4_28,

Samsi, S., Mattioli, C. J., and Veillette, M. S.: Distributed deep learning for precipitation nowcasting, 2019 IEEE High Performance Extreme Computing Conference (HPEC), 1-7, https://doi.org/10.1109/HPEC.2019.8916416,

Sato, H. and Ise, T.: Predicting global terrestrial biomes with the LeNet convolutional neural network, Geoscientific Model Development, 15, 3121-3132, https://doi.org/10.5194/gmd-15-3121-2022, 2022.

Shi, X., Chen, Z., Wang, H., Yeung, D.-Y., Wong, W.-K., and Woo, W.-c.: Convolutional LSTM network: A machine learning approach for precipitation nowcasting, Advances in neural information processing systems, 28, https://doi.org/10.48550/arXiv.1506.04214, 2015.

Shi, X., Gao, Z., Lausen, L., Wang, H., Yeung, D.-Y., Wong, W.-k., and Woo, W.-c.: Deep learning for precipitation nowcasting: A benchmark and a new model, Advances in neural information processing systems, 30, https://doi.org/10.48550/arXiv.1706.03458, 2017.

Trebing, K., Stańczyk, T., and Mehrkanoon, S.: SmaAt-UNet: Precipitation nowcasting using a small attention-UNet architecture, Pattern Recognition Letters, 145, 178-186, https://doi.org/10.1016/j.patrec.2021.01.036, 2021.

Turner, B., Zawadzki, I., and Germann, U.: Predictability of precipitation from continental radar images. Part III: Operational nowcasting implementation (MAPLE), Journal of Applied Meteorology and Climatology, 43, 231-248, https://doi.org/10.1175/1520-0450(2004)043<0231:POPFCR>2.0.CO;2, 2004.

Wang, P., Smeaton, A., Lao, S., O'Connor, E., Ling, Y., and O'Connor, N.: Short-term rainfall nowcasting: Using rainfall radar imaging, Eurographics Ireland 2009: The 9th Irish Workshop on Computer Graphics, Dublin, Ireland2009.

Wu, X., Liu, X., Zhang, D., Zhang, J., He, J., and Xu, X.: Simulating mixed land-use change under multi-label concept by integrating a convolutional neural network and cellular automata: A case study of Huizhou, China, GIScience & Remote Sensing, 59, 609-632, https://doi.org/10.1080/15481603.2022.2049493, 2022.

Xiong, T., He, J., Wang, H., Tang, X., Shi, Z., and Zeng, Q.: Contextual sa-attention convolutional LSTM for precipitation nowcasting: A spatiotemporal sequence forecasting view, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 14, 12479-12491, https://doi.org/10.1109/JSTARS.2021.3128522, 2021.

Yano, J.-I., Ziemiański, M. Z., Cullen, M., Termonia, P., Onvlee, J., Bengtsson, L., Carrassi, A., Davy, R., Deluca, A., and Gray, S. L.: Scientific challenges of convective-scale numerical weather prediction, Bulletin of the American Meteorological Society, 99, 699-710, https://doi.org/10.1175/BAMS-D-17-0125.1, 2018.

Zhang, F., Wang, X., Guan, J., Wu, M., and Guo, L.: RN-Net: A deep learning approach to 0–2 h rainfall nowcasting based on radar and automatic weather station data, Sensors, 21, 1981, https://doi.org/10.3390/s21061981, 2021a.

Zhang, Y., Bi, S., Liu, L., Chen, H., Zhang, Y., Shen, P., Yang, F., Wang, Y., Zhang, Y., and Yao, S.: Deep Learning for Polarimetric Radar Quantitative Precipitation Estimation during Landfalling Typhoons in South China, Remote Sensing, 13, 3157, https://doi.org/10.3390/rs13163157, 2021b.