

# Deep Learning-Based Rainfall Prediction Using Cloud Image Analysis

Jongyun Byun<sup>ID</sup>, Changhyun Jun<sup>ID</sup>, Member, IEEE, Jinwon Kim<sup>ID</sup>, Jaehoon Cha<sup>ID</sup>, and Roya Narimani<sup>ID</sup>

**Abstract**—This study presents a new research direction for predicting rainfall amount using cloud image data. Herein, we employ a convolutional neural networks (CNNs) to develop an image-value model from cloud image data collected from May 20, 2020 to October 24, 2020 using the Internet of Things (IoT) sensors installed at two research locations in Seoul, Republic of Korea. First, we refine the dataset using data preprocessing in three steps: 1) day/night discrimination; 2) ratio adjustment; and 3) image augmentation. Second, we construct a binary classification model using one-hot encoding for the existence of rainfall. This reduces no-rain data instances and increases model performance, thereby enabling the model to extract image features. Finally, we develop a CNN-based image-value model for rainfall prediction with a well-organized model configuration. Rainfall existence results derived from the binary classification model used for model input as preprocessed cloud image data. The proposed rainfall prediction model exhibited 85.59% accuracy on cloud images with an average mean squared error (MSE) of 3.05 for observation data under 3 mm/h. In particular, single application of the function that divides Boolean input by the standard deviation of the dataset within each characteristic resulted in a 17% increase in predicted rainfall accuracy. To the best of our knowledge, this is the first study to train CNN model to predict value (rainfall) with matched image data (cloud), which could be denoted as CNN-based image-value model. Notably, the proposed model can be further extended into other image datasets, including rain streaks with various backgrounds under different climatic conditions.

**Index Terms**—Cloud image, convolution neural networks (CNNs), deep learning, Internet of Things (IoT) sensor, rainfall prediction.

Manuscript received 17 August 2022; revised 3 January 2023 and 30 January 2023; accepted 9 March 2023. Date of publication 3 April 2023; date of current version 19 April 2023. This work was supported in part by the Korea Meteorological Administration Research and Development Program under Grant KMI2022-01910, in part by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) under Grant NRF-2022R1A4A3032838, and in part by the Korea Institute of Civil Engineering and Building Technology (KICT) Research Program (Development of IWRM-Korea Technical Convergence Platform Based on Digital New Deal) funded by the Ministry of Science and ICT under Project 20230115-001. (*Corresponding author: Changhyun Jun*)

Jongyun Byun and Roya Narimani are with the Department of Civil Engineering, Chung-Ang University (CAU), Seoul 06974, Republic of Korea (e-mail: whdbs0932@cau.ac.kr; roya.narimani1986@gmail.com).

Changhyun Jun is with the Department of Civil and Environmental Engineering and the Department of Smart Cities, Chung-Ang University (CAU), Seoul 06974, Republic of Korea (e-mail: cjun@cau.ac.kr).

Jinwon Kim is with the Department of Mobile Experience, Samsung, Suwon 16677, Republic of Korea (e-mail: xmuniv@gmail.com).

Jaehoon Cha is with the Scientific Machine Learning Research Group, Science and Technology Facilities Council (STFC), SN2 1SZ Swindon, U.K. (e-mail: Jaehoon.Cha@stfc.ac.uk).

Digital Object Identifier 10.1109/TGRS.2023.3263872

## I. INTRODUCTION

IN RECENT years, machine learning has assumed increasing importance. This rise in machine learning is intimately linked to data availability, which enables the automatic collection, storage, and distribution of large datasets. Growing data availability and the high popularity of deep learning can transform hydrological data (e.g., precipitation, runoff, and groundwater height) into actionable and practical knowledge, which may revolutionize the water industry.

Numerous studies have been conducted using machine learning for analyzing various hydrological data. Fan et al. [1] applied an artificial neural networks (ANNs) for runoff modeling and highlighted the potential of long short-term memory (LSTM) by comparing modeling results with those from the Soil and Water Assessment Tool (SWAT). Afzaal et al. [2] estimated groundwater level fluctuations using deep learning, ANNs, multilayer perceptron (MLP), LSTM, and convolutional neural networks (CNNs) and compared them with periodic dips to confirm its accuracy and convenience. Qiu et al. [3] verified the capabilities of machine learning models in forecasting daily river water temperature by comparing them with several other models commonly used worldwide. Furthermore, considering the tradeoffs between data availability and model complexity in data-driven models, Xie et al. [4] developed an ANN-based hybrid modeling approach to improve model performance in data-sparse regions.

Among the various concepts in machine learning, one of the most popular deep neural networks is CNNs. It is different from traditional neural networks because it utilizes multiple convolutional layers, pooling, local connections, and shared weights. The benefits of using CNNs can be summarized as follows. First, CNNs extract visual patterns directly and preserve the spatial information in latent spaces (see [5]). Second, CNNs process large inputs with a great computational efficiency due to sparsely connected with tied weights (see [6], [7]). Third, CNNs are equivariant to translation, which results that CNNs are robust to small transformations in the input data and can adapt to the different input sizes (see [8]). Due to these benefits, CNNs have been used in a wide range of academic fields related to data processing, such as image classification, computer vision, and even in the natural language processing (see [9], [10], [11]).

In water-related research fields, CNNs have been widely used for modeling patterns in hydrological components. Panahi et al. [12] developed groundwater potential maps using machine learning algorithms, particularly using CNNs

and support vector regression (SVR), to develop better strategies for the conservation and management of groundwater resources to alleviate freshwater shortage problems. Wang et al. [13] proposed a prediction algorithm for changes in tropical cyclone intensity based on a 3D-CNN to learn the implicit correlations between spatial distribution features and tropical cyclone intensity, considering the basic variables of the atmosphere and ocean. Baek et al. [14] simulated water level and water quality parameters (i.e., total nitrogen, total phosphorous, and total organic carbon) using CNNs and LSTM, respectively. Kabir et al. [15] presented a hydraulic model for the rapid prediction of fluvial flood inundation using a deep CNN, which can instantly predict the water depth for over half a million cells. Yu et al. [16] built a novel hybrid CNN-gated recurrent unit (CNN-GRU) model for predicting soil water content at different depths in the maize root zone using numerous selected meteorological and soil variables based on Pearson correlation analysis and soil water content autocorrelation analysis.

Meanwhile, rainfall is the most significant factor of hydrological cycle as input data. To understand various climatological phenomena within the Earth, the accurate prediction of rainfall is important. On the contrary, as the spatiotemporal variability of rainfall has shown high, numerous research projects have been carried out to reduce its variability. Markovic and Koch [17] examined the long-term precipitation and streamflow time series to quantify the difference in spectral properties and Hurt parameter estimates ( $\hat{H}$ ) of individual hydrological cycle components. Alahcoon et al. [18] revealed rainfall variability and trends with Tropical Applications of Meteorology Using Satellite Data and Ground-Based Observations (TAMSAT) data using Mann-Kendall (MK) test and Sen's slope estimator for identifying areas in Africa with significant changes in rainfall patterns. Ashraf et al. [19] analyzed satellite-based rainfall and gridded rainfall to validate the spatiotemporal variability and to identify the behavior and performance of satellite-based indices during calculated meteorological drought years. However, there is clear limitation of using long time scale data over existing studies.

Recently, utilizing data from nontraditional observation device such as the Internet of Things (IoT) could be an appropriate alternative to improve the spatiotemporal variability of rainfall. During recent years, studies based on this concept have been conducted focused on rainfall prediction using IoT sensor. Among the various types of IoT data, image is attractive as research data type, which have been collected for various purposes and involve different types of information. Li et al. [20] obtained sensitivity classification results using a rainfall recognition model from road surveillance videos and labeled rainfall data with a temporal segment network (TSN). Hakim and Dewi [21] used images from closed-circuit television (CCTV) to create a real-time rain detection system using CNNs. Notarangelo et al. [22] used images from dashboard cameras to develop a cost-effective rainfall detection model using transfer learning and CNNs. In summary, excluding remote sensing image data (e.g., radar and satellite data), few previous studies have been conducted on rainfall prediction using images from IoT sensors. Most of the existing studies

have aimed only to detect rainfall based on various types of data and not to quantify the specific rainfall intensity.

In this study, we built a CNN-based image-value model using cloud image data and an automatic weather station (AWS) in Seoul, Republic of Korea. Our aim is to predict rainfall amounts using cloud image data from IoT sensor based on a CNN algorithm, which is optimized for extracting the feature of image data, which can be used to forecast microscale weather information.

In particular, We:

- 1) constructed a CNN-based image-value model that was divided into binary classification for rainfall existence and a rainfall prediction model after preprocessing steps;
- 2) evaluated the acceptability of rainfall prediction machine learning model using cloud image data.

## II. DATA

### A. IoT Sensor and Cloud Image

In this study, cloud image data were collected from an automatic weather device called "BloomSky." It is a community-based weather camera station that shares weather data among users. Using this device, real-time cloud images and hydrological factors can be obtained. The BloomSky device, the first community-based weather camera station, is divided into two parts (SKY2 and STORM) that collect different weather factors. SKY2 is a five-in-one weather camera station that can measure the temperature, humidity, barometric pressure, precipitation, and cloud images. STORM is a four-in-one wireless weather device that can measure real-time UV, rainfall, wind speed, and wind direction. The SKY2 device is equipped with a camera that records cloud images, which are the main components of the total dataset. Table I lists the specifications of the SKY2 device in BloomSky.

This study considers two research locations with the Han River between them. Specifically, Chung-Ang University and Korea University were selected on behalf of the Gangnam area (southern Seoul) and Gangbuk area (northern Seoul), respectively. Detailed information regarding the installed locations of BloomSky and their data specifications are listed in Table II.

### B. Observation Data From AWS

For the purpose of total dataset construction, ground truth values are required for matching with cloud images from BloomSky devices. Korea Meteorological Administration (KMA) (<https://www.kma.go.kr>) has provided an AWS and an automated synoptic observing system (ASOS) as ground-truth values to perform ground-based weather observations. It was confirmed that 494 AWS stations and 96 ASOS stations provide observation data containing weather data, such as wind direction, wind speed, precipitation, air pressure, and humidity on a minute timescale. The closest AWS locations to each BloomSky device (i.e., Dongdaemun, Hyeonchungwon) were selected as the ground-truth values of Location 1 and 2. The two locations have the coordinates (37.5004N, 126.9765E) and (37.5846N, 127.0604E), respectively. According to Table I, the BloomSky device also provides precipitation data. However, in this study, AWS stations were selected to match the cloud

TABLE I  
SPECIFICATION OF SKY2 DEVICE IN BLOOMSKY([HTTPS://BLOOMSKY.DESK.COM](https://bloomsky.desk.com))

Camera	<ul style="list-style-type: none"> <li>HD camera with ultra-wide angle lens (170 degree)</li> <li>Max resolution of original image: <math>1920 \times 1072</math> <ul style="list-style-type: none"> <li>Transmitted image size: <math>640 \times 640</math></li> </ul> </li> <li>Image capture frequency: ~5 minute intervals during daytime           <ul style="list-style-type: none"> <li>Turn on at dawn and turn off at dusk based on local area sunrise and sunset times</li> <li>Image interval changes to ~3 minutes for first and last hour of camera activity each day</li> </ul> </li> <li>Upload frequency: ~5 minutes           <ul style="list-style-type: none"> <li>Uploaded data and image immediately if a sudden change in temperature (greater than <math>2^{\circ}\text{C}</math> within 20 s) or precipitation is detected.</li> </ul> </li> </ul>
	<ul style="list-style-type: none"> <li>Powered by a rechargeable lithium-ion battery</li> <li>Battery operating and charging temperature range: <math>-4^{\circ}\text{F}</math> (<math>-20^{\circ}\text{C}</math>) to <math>131^{\circ}\text{F}</math> (<math>55^{\circ}\text{C}</math>)</li> <li>Charging options:           <ul style="list-style-type: none"> <li>via A/C Input: 100-240V~50/60Hz (worldwide use), Output: DC 12V-2A               <ul style="list-style-type: none"> <li>Approximate operating time: up to two weeks per charge</li> </ul> </li> <li>via solar panel: Output: DC 12V-1A               <ul style="list-style-type: none"> <li>Approximately 10-12 hours per week of direct sunlight will fully charge the internal battery</li> </ul> </li> </ul> </li> </ul>
	<ul style="list-style-type: none"> <li>WLAN Standard: Single-band 2.4GHz IEEE 802.11b/g/n</li> <li>Frequency Range: 2412 MHz ~ 2462 MHz (2.4 GHz ISM Band)</li> <li>Number of Channels: 2.4GHz: Ch1 ~ Ch11</li> <li>Bluetooth V4.0 of 1,2,3 Mbps</li> </ul>

TABLE II  
LOCATIONS OF IOT SENSORS AND INFORMATION ABOUT CLOUD IMAGE DATA

Alternative Name	Original Name	Location		Data Period	Image Data Number
		Latitude	Longitude		
Location 1	Chung-Ang University	37.5035N	126.9579E	20 May 2020 – 01 Aug 2020	5,384
Location 2	Korea University	37.5836N	127.0253E	01 Aug 2020 – 24 Oct 2020	7,150

image to the ground truth value. Because BloomSky data are unreliable due to signal errors and a low-quality control status owing to the time interval irregularity, we conducted our study using the data that contain the rainfall measured at different locations. For the same time point, the cloud image taken by the IoT sensor and the rainfall data acquired from AWS were matched as a pair and used as input data for the model.

### III. METHODOLOGY

In this study, a CNN-based image-value model was developed to accurately determine the existence of rainfall and predict rainfall amounts. Fig. 1 shows a flowchart of rainfall prediction using cloud image data. The proposed model was divided into three major parts: data preprocessing, model training, and robustness checks.

#### A. Data Preprocessing

1) *Day/Night Discrimination*: First, the collected image data were divided into two image groups (i.e., day and

night) to prevent overfitting during the model training process. Furthermore, because the camera device used in this study did not have a night vision mode, images taken at night required filtering. If not, a different amount of ground rainfall data can be matched to similar images consisting of nearly zero-pixel values, which can produce an overfitted model. We transferred the image color space from red, green, and blue (RGB) to hue, saturation, and value (HSV) and set up a specific threshold value to compare it with the average *V* component for all pixels in each image. Herein, we refer this process as “night drop”, which is used to adjust the ratio of day and night images.

2) *Ratio Adjustment*: In the second step, ratio adjustment was performed to prevent overfitting in the model. Because we were dealing with short-timescale rainfall data, the number of no-rainfall data was bound to increase rapidly. When the same (or zero) rainfall intensity and a large amount of different image data are matched as a single set to train the model, the learning efficiency of the model decreases. Recent studies have

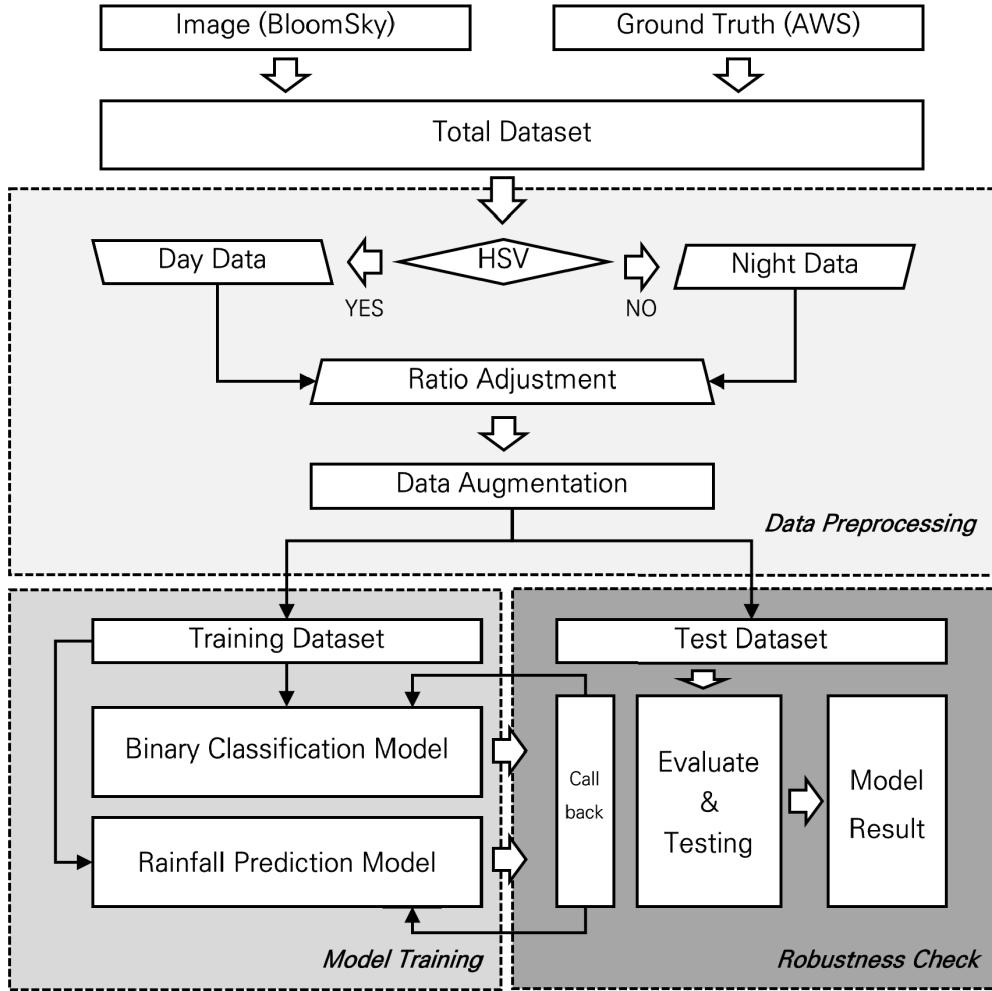


Fig. 1. Flowchart for the deep learning-based rainfall prediction using cloud image analysis.

promoted diverse approaches to solve this problem; a simple solution is to adjust the ratio of the dataset. In this study, the ratio between rain data and no-rain data was 80%:20%.

**3) Image Augmentation:** In the final stage, image augmentation was performed to improve model accuracy and solve the problem of limited data. Here, various functions were applied from “Keras Image Data Generator”. These functions included, geometric transformations, color space augmentations, mixing images, pixel normalization, and feature space augmentation. We considered that these functions would result in a high quality and quantity of image data, which would serve as an appropriate data source for obtaining highly robust deep learning models.

#### B. Model Training

**1) Binary Classification Model for Rainfall Existence:** In this study, the original data were filtered before matching the cloud image to the observation data, depending on the existence of rainfall, to maximize the model efficiency. Here, we call this process “binary classification for rainfall existence”. If there is image data with no rainfall, more time is required to extract the features when calculating the amount of rainfall. However, such data were filtered using this process and was considered as data with zero as the amount of rainfall.

Fig. 2 shows the number of days with no rainfall in each month in 2020 in Seoul, Republic of Korea. It should be noted that 263 out of 366 days had no rainfall, indicating a high proportion (71.86%) of the total observation data. Thus, binary classification was used to filter the no-rain period and adjust the ratio of the total dataset to prevent overfitting during model processing. Fig. 3 shows the structure of the proposed binary classification model for rainfall existence with model parameters and structural information. Here, the ratio of no-rain data to rain data is 1/4, and the model configuration can be denoted by two convolution and max pooling layers, which are finally connected to the fully connected (FC) layer.

**2) CNN Model for Rainfall Prediction:** Fig. 4 represents a structure of the CNN-based image-value model for rainfall prediction with model parameters and structural information. It considers the refined dataset after data preprocessing and binary classification. The final part of the model was required to produce a specific amount of rainfall. For this reason, we stacked the model layer with a composition of various layers: three convolutional layers, two pooling layers, two FC layers, and an output layer.

Each convolutional layer includes a rectified linear unit (ReLU) activation function (see [23]), which is widely used in machine learning models to successfully capture nonlinear

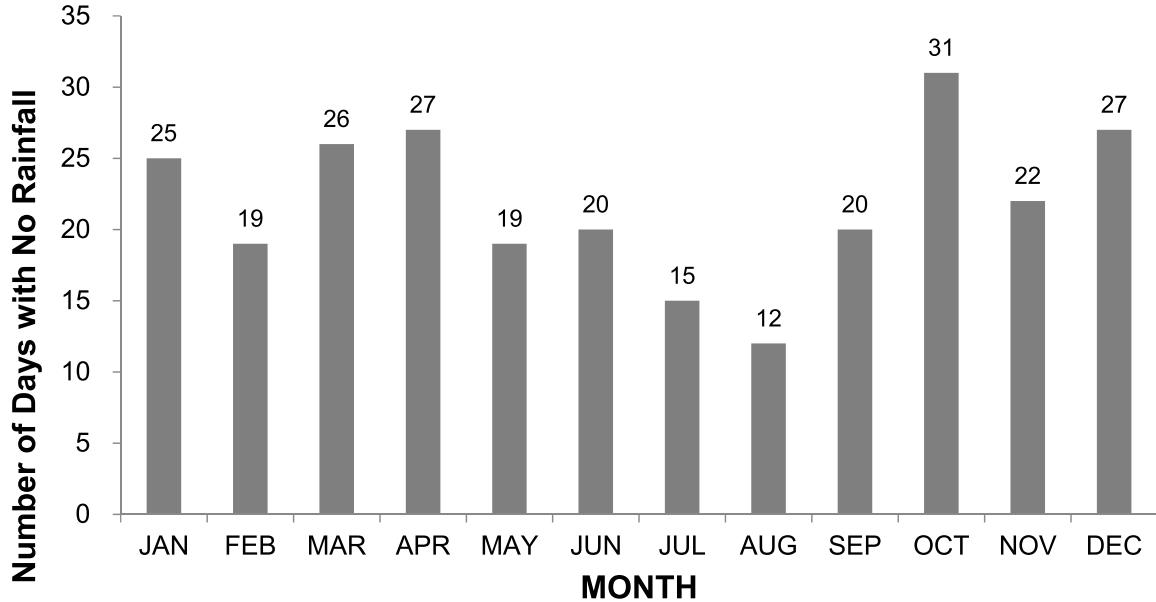


Fig. 2. Number of days without rainfall for each month in the year 2020 in Seoul, Republic of Korea.

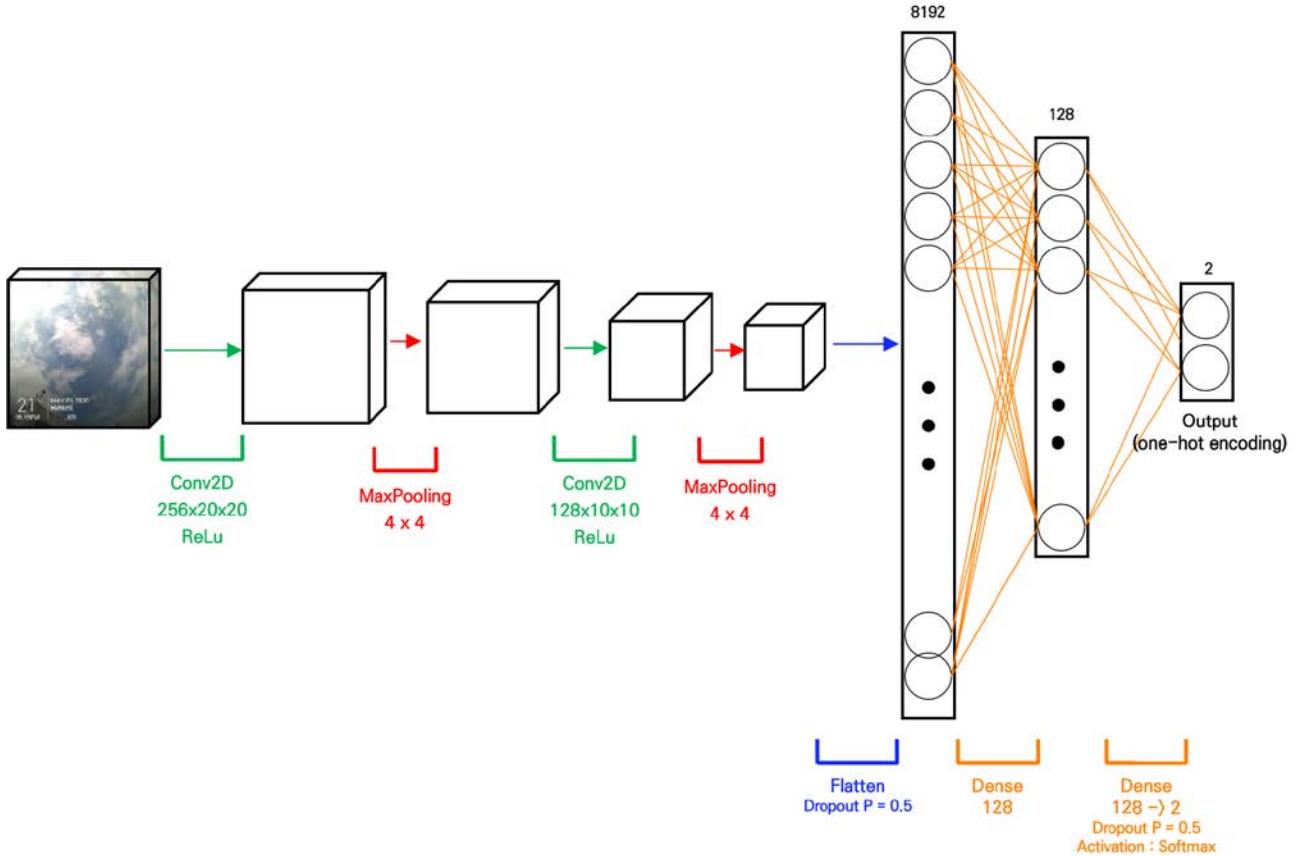


Fig. 3. Structure of binary classification model for rainfall existence.

relationships (see [24]). Here, we adopted ReLU to include the complex relationship between cloud image data and rainfall in the model and maximize the ability of different neural networks. Assuming that the feature maps in the  $j$ th layer are  $Z^j$ , the outputs of the  $(j + 1)$ th layer can be formulated as follows:

$$Z^{j+1} = f_s(W^{j+1} \cdot Z^j + b^{j+1})$$

where  $W^{j+1}$  and  $b^{j+1}$  are the weight matrix and bias vector to link the feature maps in the  $j$ th layer to the feature maps in the  $(j + 1)$ th layer, respectively (see [25]).  $f_s$  denotes the nonlinear activation function, which is the ReLU used in this study. After the convolutional layers, the feature maps of the last pooling layer were flattened, which gradually reduced the dimensions of the features and generated a compact feature representation before the output layer. Most of research related

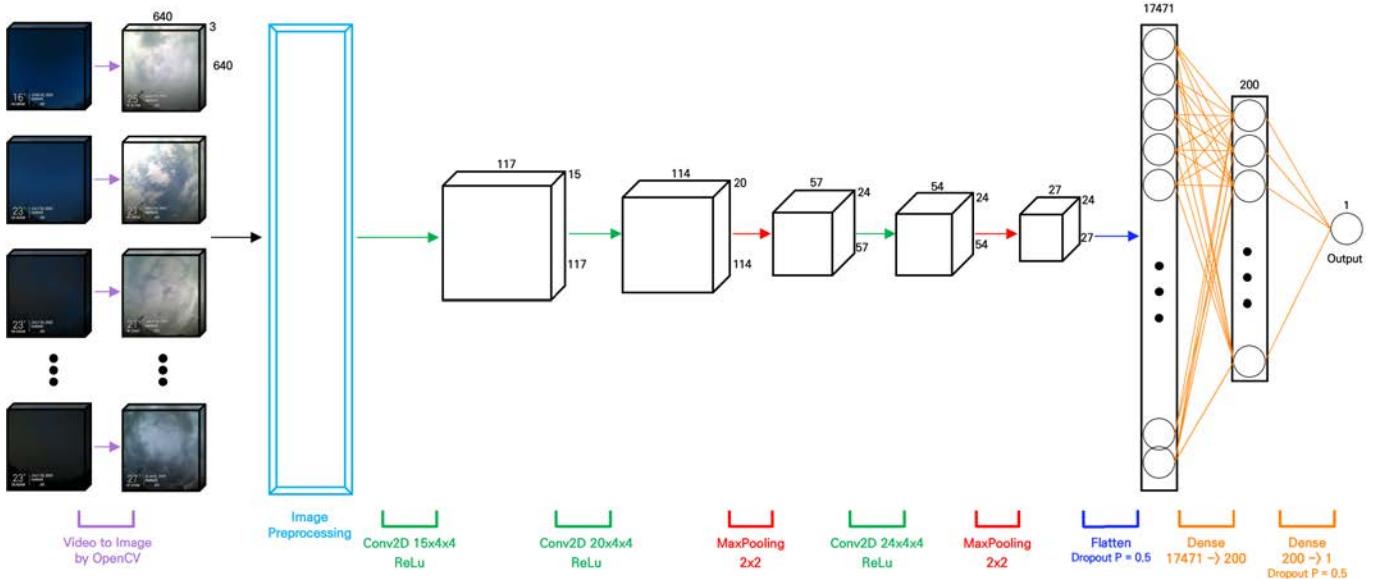


Fig. 4. Structure of the CNN-based image-value model used for rainfall prediction.

to CNN model are connected to transformed output layer consist of decimal probabilities to solve the multiclass problem (see [26], [27]).

In contrast, the purpose of the final part in this research is to calculate the amount of rainfall. The number of nodes in the output layer needs to be one, and its value should be noted as the rainfall corresponding to cloud image data. It appears that CNN model for rainfall prediction in this research is the most striking trial to modify the CNN model itself with cloud image data from IoT sensor and rainfall data from the ground. So far, no model has ever used CNN model to calculate specific value, not for image classification purposes. Detailed information about the proposed models (i.e., the binary classification model and CNN model) is summarized in the Appendix (Table III).

### C. Robustness Check

1) *Callback Function:* Before training the main model, callbacks were used to ensure that the proposed model could be trained as intended. To ensure this, we used the default options implemented in the Keras callbacks application programming interface (API) to monitor the model after each iteration. The callback function included the following three steps: 1) EarlyStopping; 2) ReduceLROnPlateau; and 3) ModelCheckpoint approaches to solve this problem; a simple solution is to adjust the ratio of the dataset. In this study, the ratio between rain data and no-rain data was 80%:20%.

#### 1) EarlyStopping

The early stopping checks whether the loss decreases at the end of every epoch. Patience and the minimum value of delta are provided with parameters. If the callback detected no reduction in loss, the training was terminated.

#### 2) ReduceLROnPlateau

According to Keras, “models often benefit from reducing the learning rate once learning stagnates.” We agree with this statement, and hence, we used this callback. If there are no improvements, the callback function will decrease the learning

rate considering patience and other factors, if applicable. During the experiment, if learning decayed, it was considered as a good attempt to reduce the learning rate.

#### 3) ModelCheckpoint

We also used the ModelCheckpoint callback to save the trained model with the best performance. With this callback, we saved the model that achieved the best performance, which, in our case, showed highest reduction in learning rate. Using the callback API provided by Keras, we attempted to control the training environment to our expectations, avoid overfitting, and reduce the learning rate to boost training results.

2) *Evaluation and Testing:* To optimize the parameters in the proposed network, a suitable loss function is required. In this study, different loss functions were designed to train the binary classification for rainfall existence and the CNN-based image-value model for rainfall prediction. Motivated by classification tasks in the field of computer vision (see [28]), cross-entropy (CE) loss was adopted as the loss function for the binary classification model, which characterizes the distance between two probability distributions. Because the number of results in the classification model is two (rain and no rain), the loss function can be converted into binary CE (BCE)

$$L_{\text{BCE}} = - \sum_{i=1}^{C'=2} y_i \log(y'_i) \\ = -y_1 \log(y'_1) - (1 - y_1) \log(1 - y'_1).$$

In case of rainfall prediction model, mean squared error (MSE) was adopted as the loss function

$$L_{\text{MSE}} = \frac{1}{N} \sum_{i=1}^N \|y_i - y'_i\|_2^2$$

where  $y_i$  and  $y'_i$  denote the real and predicted values, respectively,  $N$  is the number of data,  $L_{\text{BCE}}$  is the BCE loss, and  $L_{\text{MSE}}$  is the MSE loss function.

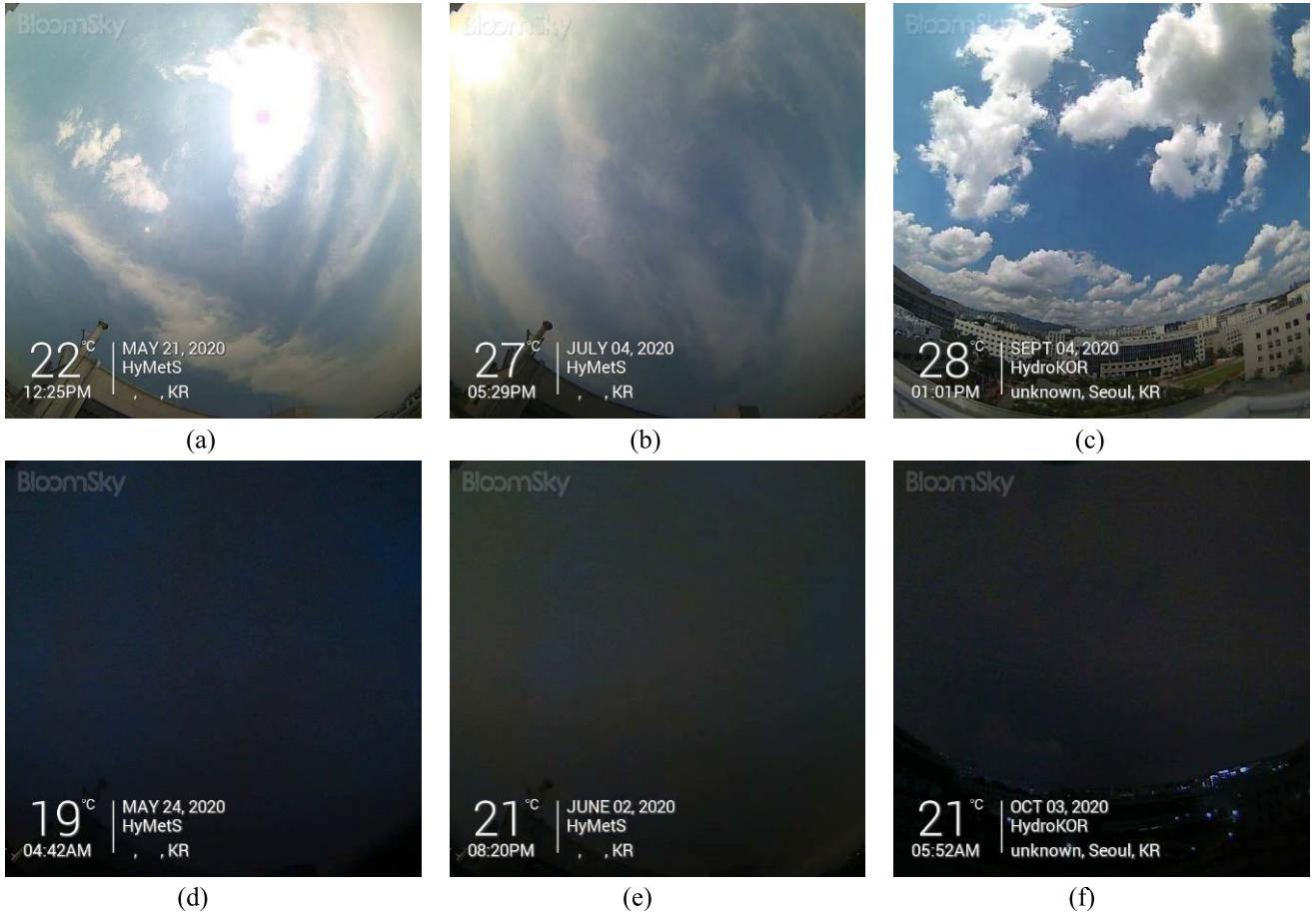


Fig. 5. Day and night discrimination result separated by average V pixel value. (a) Day Image 01: 199.55 at 12:25 on May 21, 2020, Location 1. (b) Day Image 02: 137.64 at 17:29 on July 04, 2020, Location 1. (c) Day Image 03: 167.81 at 13:01 on September 04, 2020, Location 2. (d) Night Image 01: 47.47 at 04:42 on May 24, 2020, Location 1. (e) Night Image 02: 46.88 at 20:20 on June 02, 2020, Location 1. (f) Night Image 03: 35.08 at 05:52 on October 02, 2020, Location 1.

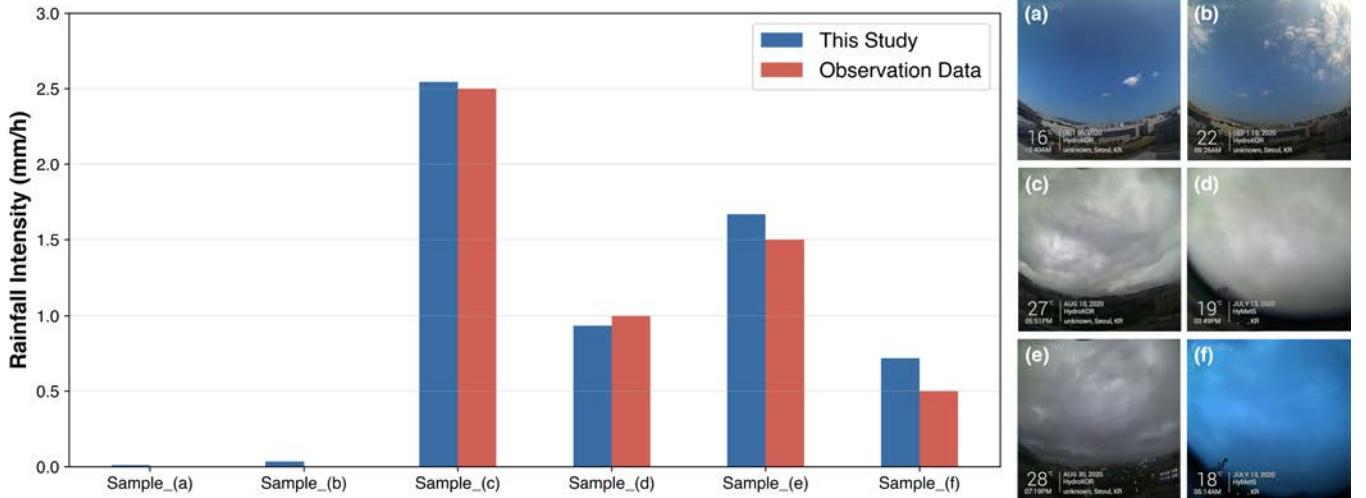


Fig. 6. Bar chart showing a comparison of the results of this study with the observation data. (a) (0.011, 0) at 10:40 on October 5, 2020, Location 1. (b) (0.035, 0) at 09:26 on September 19, 2020, Location 1. (c) (2.545, 2) at 17:51 on August 10, 2020, Location 2. (d) (0.093, 0) at 15:49 on July 13, 2020, Location 1. (e) (1.670, 1.5) at 19:19 on August 30, 2020, Location 2. (f) (0.719, 0.5) at 05:14 on July 13, 2020, Location 1.

## IV. RESULTS

### A. Data Handling

The discrimination of the entire dataset using the HSV threshold is shown in Fig. 5. Even though the amount of rainfall during the night is not negligible, we focused on

the fact that the nighttime image data do not have sufficient characteristics that can be distinguished according to rainfall. If the average of all pixels was less than the V value of 50, it was regarded as night image and was hence removed. The data augmentation process using “Keras Image Augmentation API” function led to 25% of the newly generated image tensor

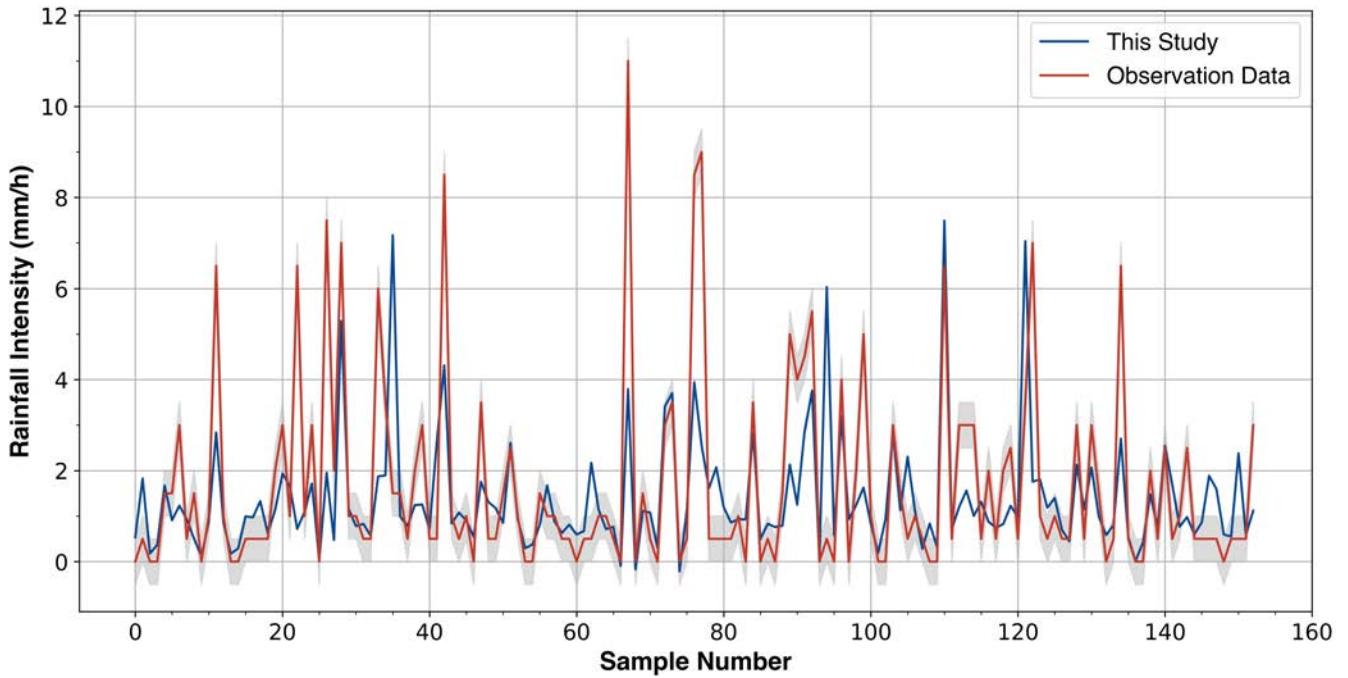


Fig. 7. Comparison of research and observation values according to sample number.

and a 37% decrease in total loss. In particular, function that divides Boolean input by the standard deviation of the dataset within each characteristic derived a conspicuous result that increased the accuracy of the predicted rainfall range by 17% with a single application.

#### B. Model Performance and Interpretation

The binary classification model correctly classified the existence of rain with 60% accuracy. As the original purpose of this model was to support the rainfall prediction model, the final results were converted to be provided as additional input data for the rainfall prediction CNN-based image-value model.

The results obtained from the rainfall prediction CNN-based image-value model are presented in Fig. 6. Six-sample test dataset was randomly selected to measure the difference between the values of the constructed model in this study and the observation data. The overall results of the model proposed in this study are discussed in the following. It should be noted that we assumed that the boundary rainfall value was 3 mm/h according to the standards presented by the KMA and that the margin of error was 1 mm/h.

The value of the loss decreased with an increase in the number of iterations and tended to be stable after approximately 30 epochs. When the training process was complete, the results were evaluated to adjust the overall model architecture. The final model shows that MSE values varied between 0.0035 and 0.0314, with an average value of 0.0057. The model error results related to the model history are schematized in the Appendix (Fig. 8).

Examining the rainfall prediction results in Fig. 7 to interpret the model performance shows that the CNN-based image value model identifies high stability in rainfall data less than 3 mm/h. Specifically, 77.12% of the training data showed

rainfall below 3 mm/h, and among them, 85.59% of the data showed a difference of less than 1 mm/h between this study and the observation value. In contrast, the model identified a strong anomaly tendency. Of the training data, 22.88% corresponded to more than 3 mm/h and 22.86% of the data showed a difference of less than 1 mm/h. At the end of the training period, the average MSE calculated with train dataset was 3.0453 for the total model assessment.

#### V. CONCLUSION

In this study, a binary classification model for verifying the existence of rainfall and a CNN-based image-value model for utilizing cloud images were used to accurately predict the amount of rainfall for selected AWS locations in Seoul, Republic of Korea. Two models with high interaction were applied to ensure data robustness, and reasonable prediction results were obtained. The following conclusions were drawn from the obtained results.

- 1) By preprocessing the data before model construction, we found an effective function that could increase the model performance. In particular, the function that divides Boolean input by the standard deviation of the dataset within each characteristic has a significant impact on the preprocessing of the cloud image and affects the overall loss reduction.
- 2) The binary classification model classifies rain with an accuracy of 60%. We confirmed that the binary classification model itself could not provide proper performance with cloud image binary classification and used it as additional input data for the rainfall prediction model.
- 3) The rainfall prediction model using the deep learning method performed well in predicting rainfall intensity

TABLE III  
INFORMATION ABOUT THE MODELS USED IN THIS STUDY

Model	Layer	Shape	Output Size
Binary Classification Model for Precipitation Existence	Input	3@120 × 120	3@120 × 120
	Conv1	256@20 × 20	256@ 101 × 101
	Pool1	4 × 4	256@ 25 × 25
	Conv2	128@10 × 10	128@ 16 × 16
	Pool2	4 × 4	128@ 8 × 8
	Flatten	8192	8192
	FC	128	128
	Output	2	2
CNN-based image-value Model for Precipitation Prediction	Input	3@120 × 120	3@120 × 120
	Conv1	15@4 × 4	15@117 × 117
	Conv2	20@4 × 4	20@114 × 114
	Pool1	2 × 2	24@57 × 57
	Conv3	24@4 × 4	24@54 × 54
	Pool2	2 × 2	24@27 × 27
	Flatten	17476	17476
	FC	200	200
	Output	1	1

\* Conv : Convolutional layer

\* Pool : Maxpooling layer

\* FC : Fully Connected layer

\* The number after the layer indicates n-th layer it is.

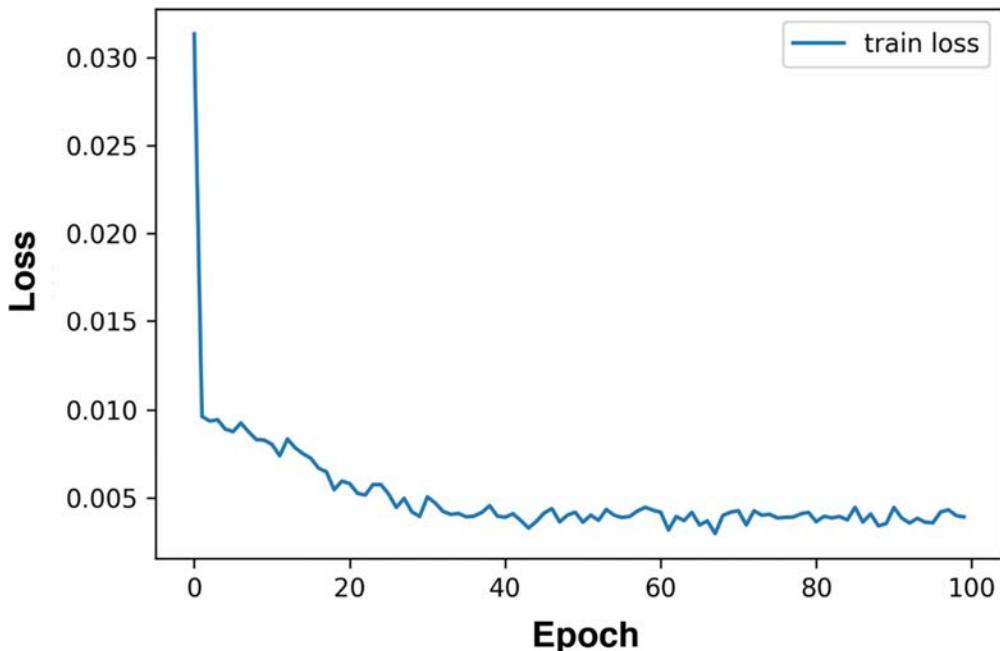


Fig. 8. Model train loss for 100 epochs.

less than 3 mm/h, 85.59% of the prediction results were within the 1 mm/h range. Moreover, the MSE from the test dataset was calculated to be 3.0453.

Based on the results of this study, which consist of a deep learning architecture using IoT sensor-based data and present a highly reliable predicted rainfall range, it can be asserted

that our proposed model is so far the ultimate solution to the problem of climate change in the observational blank area. Although our research is confined to limited data and an unfamiliar model, it demonstrates that implementing deep learning methods in IoT sensor-based data could provide a fresh perspective for physics-based observation methods. Even though it has the limitation of accurately calculating the amount of rainfall, it can suggest cost-effective method that reflects the microclimate in a small area. As the complete dataset was available only at specific locations, it could be readily retrained to analyze the correlation between cloud image data and rainfall, allowing the training for different locations and considering various types of rain events and cloud formations. A more scientific understanding of systemic model planning and elaborate data processing is required for future studies.

## APPENDIX

See Table III and Fig. 8.

## REFERENCES

- [1] H. Fan, M. Jiang, L. Xu, H. Zhu, J. Cheng, and J. Jiang, "Comparison of long short term memory networks and the hydrological model in runoff simulation," *Water*, vol. 12, no. 1, p. 175, 2020.
- [2] H. Afzaal, A. A. Farooque, F. Abbas, B. Acharya, and T. Esau, "Groundwater estimation from major physical hydrology components using artificial neural networks and deep learning," *Water*, vol. 12, no. 1, p. 5, Dec. 2019. [Online]. Available: <https://www.mdpi.com/2073-4441/12/1/5>
- [3] R. Qiu et al., "River water temperature forecasting using a deep learning method," *J. Hydrol.*, vol. 595, Apr. 2021, Art. no. 126016.
- [4] S. Xie, W. Wu, S. Mooser, Q. J. Wang, R. Nathan, and Y. Huang, "Artificial neural network based hybrid modeling approach for flood inundation modeling," *J. Hydrol.*, vol. 592, Jan. 2021, Art. no. 125605.
- [5] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "SqueezeNet: AlexNet-level accuracy with 50× fewer parameters and <0.5 MB model size," 2016, *arXiv:1602.07360*.
- [6] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, May 2015, doi: [10.1038/nature14539](https://doi.org/10.1038/nature14539).
- [7] S. Kiranyaz, O. Avci, O. Abdeljaber, T. Ince, M. Gabboj, and D. J. Inman, "1D convolutional neural networks and applications: A survey," *Mech. Syst. Signal Process.*, vol. 151, Apr. 2021, Art. no. 107398.
- [8] T. S. Cohen and M. Welling, "Group equivariant convolutional networks," *Proc. Mach. Learn. Res.*, vol. 48, pp. 2990–2999, Jun. 2016.
- [9] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, in Lecture Notes in Computer Science, vol. 9351, 2015, pp. 234–241, doi: [10.1007/978-3-319-24574-4\\_28](https://doi.org/10.1007/978-3-319-24574-4_28).
- [10] O. Russakovsky et al., "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015, doi: [10.1007/s11263-015-0816-y](https://doi.org/10.1007/s11263-015-0816-y).
- [11] Y. Kang, M. Ozdogan, X. Zhu, Z. Ye, C. Hain, and M. Anderson, "Comparative assessment of environmental variables and machine learning algorithms for maize yield prediction in the U.S. Midwest," *Environ. Res. Lett.*, vol. 15, no. 6, Jun. 2020, Art. no. 064005, doi: [10.1088/1748-9326/ab7df9](https://doi.org/10.1088/1748-9326/ab7df9).
- [12] M. Panahi, N. Sadhasivam, H. R. Pourghasemi, F. Rezaie, and S. Lee, "Spatial prediction of groundwater potential mapping based on convolutional neural network (CNN) and support vector regression (SVR)," *J. Hydrol.*, vol. 588, Sep. 2020, Art. no. 125033.
- [13] X. Wang, W. Wang, and B. Yan, "Tropical cyclone intensity change prediction based on surrounding environmental conditions with deep learning," *Water*, vol. 12, no. 10, p. 2685, Sep. 2020.
- [14] S.-S. Baek, J. Pyo, and J. A. Chun, "Prediction of water level and water quality using a CNN-LSTM combined deep learning approach," *Water*, vol. 12, no. 12, p. 3399, Dec. 2020.
- [15] S. Kabir, S. Patidar, X. L. Xia, Q. H. Liang, J. Neal, and G. Pender, "A deep convolutional neural network model for rapid prediction of fluvial flood inundation," *J. Hydrol.*, vol. 590, Nov. 2020, doi: [10.1016/j.jhydrol.2020.125481](https://doi.org/10.1016/j.jhydrol.2020.125481).
- [16] J. Yu, X. Zhang, L. Xu, J. Dong, and L. Zhangzhong, "A hybrid CNN-GRU model for predicting soil moisture in maize root zone," *Agricul. Water Manage.*, vol. 245, Feb. 2021, Art. no. 106649.
- [17] D. Markovic and M. Koch, "Stream response to precipitation variability: A spectral view based on analysis and modelling of hydrological cycle components," *Hydrol. Processes*, vol. 29, no. 7, pp. 1806–1816, Mar. 2015.
- [18] N. Alahcoon, M. Edirisinghe, M. Simwanda, E. Perera, V. R. Nyirenda, and M. Ranagamage, "Rainfall variability and trends over the African continent using TAMSAT data (1983–2020): Towards climate change resilience and adaptation," *Remote Sens.*, vol. 14, no. 1, p. 96, Dec. 2021.
- [19] M. Ashraf, K. Ullah, and S. Adnan, "Satellite based impact assessment of temperature and rainfall variability on drought indices in southern Pakistan," *Int. J. Appl. Earth Observ. Geoinf.*, vol. 108, Apr. 2022, Art. no. 102726.
- [20] Z. Li, J. Hyeon, and H.-J. Choi, "Rainfall recognition from road surveillance videos using TSN," *J. Korean Soc. Atmos. Environ.*, vol. 34, no. 5, pp. 735–747, Oct. 2018, doi: [10.5572/Kosae.2018.34.5.735](https://doi.org/10.5572/Kosae.2018.34.5.735).
- [21] A. L. Hakim and R. Dewi, "Automatic rain detection system based on digital images of CCTV cameras using convolutional neural network method," *IOP Conf. Ser. Earth Environ. Sci.*, vol. 893, no. 1, Nov. 2021, Art. no. 012048.
- [22] N. Notarangelo, K. Hirano, R. Albano, and A. Sole, "Transfer learning with convolutional neural networks for rainfall detection in single images," *Water*, vol. 13, no. 5, p. 588, Feb. 2021, doi: [10.3390/w13050588](https://doi.org/10.3390/w13050588).
- [23] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. ICML*, 2010, pp. 807–814.
- [24] S. S. Talathi and A. Vartak, "Improving performance of recurrent neural network with ReLU nonlinearity," 2015, *arXiv:1511.03771*.
- [25] M. Xue, R. Hang, Q. Liu, X.-T. Yuan, and X. Lu, "CNN-based near-real-time precipitation estimation from Fengyun-2 satellite over Xinjiang, China," *Atmos. Res.*, vol. 250, Mar. 2021, Art. no. 105337.
- [26] C. G. de Vita, G. Mellone, F. Barchiesi, D. Di Luccio, A. Ciaramella, and R. Montella, "Artificial intelligence for mussels farm quality assessment and prediction," in *Proc. IEEE Int. Workshop Metrol. Sea; Learn. Measure Sea Health Parameters (MetroSea)*, Oct. 2022, pp. 33–38.
- [27] A. Sen and K. Deb, "Categorization of actions in soccer videos using a combination of transfer learning and gated recurrent unit," *ICT Exp.*, vol. 8, no. 1, pp. 65–71, Mar. 2022.
- [28] L. Hui and M. Belkin, "Evaluation of neural architectures trained with square loss vs cross-entropy in classification tasks," 2020, *arXiv:2006.07322*.



**Jongyun Byun** received the B.E. degree from the Department of Civil and Environmental Engineering, Chung-Ang University (CAU), Seoul, Republic of Korea, in 2021, and the M.Sc. degree in water resources and coastal engineering from the Department of Civil Engineering, CAU, in 2023, where he is currently pursuing the Ph.D. degree in water resources and coastal engineering.

His research interests include computer vision, the Internet of Things (IoT) sensor, remote sensing data processing, machine learning, and climate change.



**Changhyun Jun** (Member, IEEE) received the bachelor's degree in civil and environmental engineering and the Ph.D. degree in water resources engineering from Korea University, Seoul, South Korea, in 2010 and 2014, respectively.

During three years, he worked as a Post-Doctoral Research Fellow at the Hong Kong University of Science and Technology, Hong Kong, Politecnico di Milan, Milan, Italy, and Nanyang Technological University, Singapore. In 2017, he started to work as a Lecturer (i.e., Assistant Professor) at the Department of Civil Engineering, Xi'an Jiaotong-Liverpool University, Suzhou, China. Then, in 2019, he was hired as an Assistant Professor at Chung-Ang University (CAU). He is an Associate Professor at the Department of Civil and Environmental Engineering and the Department of Smart Cities, CAU, Seoul, Republic of Korea. As a Leader of the HydroMeteorology Sciences (HyMetS) Group at CAU, he has conducted investigations on resilient urban design for climate change adaptation and urban water disaster mitigation focusing on measurement and analyses of hydrological processes over a range of spatial and temporal scales in urban areas for water/climate sensitive cities toward sustainable nations. His research interests include the hydrological modeling, climate change, disaster and risk management, hydrometeorology, water cycle, artificial intelligence, and public health.



**Jaehoon Cha** received the Ph.D. degree in electrical engineering and electronics from the University of Liverpool, Liverpool, U.K., in 2020.

He is currently a Data Scientist with the Scientific Machine Learning Research Group, Rutherford Appleton Laboratory, Science and Technology Facilities Council (STFC), Swindon, U.K. His research interests include machine learning and its application to science and representation learning.



**Jinwon Kim** received the B.E. degree minored in media communications from the Department of Computer Science and Engineering, Chung-Ang University (CAU), Seoul, Republic of Korea, in 2022.

He is currently working as a Researcher with the Department of Mobile Experience, Samsung, Suwon, Republic of Korea. His research interests include computer vision and image/video understanding.



**Roya Narimani** is currently pursuing the Ph.D. degree in water resources and coastal engineering with the Department of Civil Engineering, Chung-Ang University (CAU), Seoul, South Korea.

Her research interests include hydrological modeling, deep learning, machine learning, climate change, water resources modeling and simulation, watershed hydrology, remote sensing, cluster analysis, damage cost estimation, risk and reliability analysis, landslides, data processing, rainfall-runoff modeling, geospatial analysis, weather generators, and multi-criteria decision making (MCDM) methods.