*Article*

# Application of Convolution Neural Networks and Hydrological Images for the Estimation of Pollutant Loads in Ungauged Watersheds

Chul Min Song

Department of Policy for Watershed Management, The Policy Council for Paldang Watershed, Yangpyeong 12585, Korea; kaii4@nate.com; Tel.: +82-01-3117-3817

**Abstract:** River monitoring and predicting analysis for establishing pollutant loads management require numerous budgets and human resources. However, it is general that the number of government officials in charge of these tasks is few. Although the government has been commissioning a study related to river management to experts, it has been inevitable to avoid the consumption of a massive budget because the characteristics of pollutant loads present various patterns according to topographic of the watershed, such as topology like South Korea. To address this, previous studies have used conceptual and empirical models and have recently used artificial neural network models. The conceptual model has a shortcoming in which it required massive data and has vexatious that has to enforce the sensitivity and uncertain analysis. The empirical model and artificial neural network (ANN) need lower data than a conceptual model; however, these models have a flaw that could not reflect the topographical characteristic. To this end, this study has used a convolution neural network (CNN), one of the deep learning algorithms, to reflect the topographical characteristic and had estimated the pollutant loads of ungauged watersheds. The estimation results for the biochemical oxygen demand (BOD) and total phosphorus (TP) loads for three ungauged watersheds were all excellent. However, prediction results with low accuracy were obtained when the hydrological images of a watershed with a land cover status different from the ungauged watersheds were used as training data for the CNN model.

**Keywords:** ungauged watershed; convolution neural network; hydrological image; curve number; biochemical oxygen demand; total phosphorus

## 1. Introduction

The control of pollution loads discharged from watersheds plays a crucial role in preserving healthy water resources. In particular, the case of South Korea is globally rather unusual in that approximately 50% (approximately 25 million people) of the Korean population uses a single water resource known as Paldang Lake; therefore, the South Korean government imposes strict control measures on the pollution loads to maintain the health status of Paldang Lake and ensure safety from a national security perspective. A qualitative management method was adopted by the Korean government for the water quality management policy in which a certain level of standard concentrations was set for management until the end of the 1990s. However, with this method, the water quality continued to degrade, thus requiring a paradigm shift in water quality management. Accordingly, the governmental water quality management policy was transformed into a quantitative management method, which has been applied to date in terms of the planning and administrative implementation for pollution load control. In establishing a plan for quantitative water quality management, the most important factor is the prediction of the pollution load discharged from watersheds. As pollution load is affected by certain conditions—such as the land cover in the watershed, physical characteristics of soil, and

hydrological phenomenon—the discharge characteristics of the pollution load show different patterns for each watershed. In this regard, when actual measurements are difficult due to physical limitations in terms of establishing objectives for pollution load management research or planning, using a model capable of estimating the related pollution load is a common practice.

Two types of models are employed for pollution load estimation, i.e., conceptual and empirical models. The conceptual model uses factors, such as variations in human activities or in natural conditions that generate pollution loads as parameters for empirical formulas or estimation equations based on physical laws, which enable the description of the causal relationship between the input value and resultant values. This type of conceptual model includes the agricultural non-point source pollution model [1], soil water assessment tool [2], storm water management model [3], and areal non-point source watershed environment response simulation [4]. However, contrary to the distinct advantages mentioned above, the conceptual model requires a large amount of data for human activities, as well as meteorological and spatial data, and has other disadvantages that require post-processing, such as testing and calibration to adjust the gap between measured and simulated data, as well as sensitivity and uncertainty analyses. In contrast, the empirical model is characterized by no dependence on physical laws and is developed based on a stable relationship between the input and output data [5]. The selection and use of an appropriate statistical analysis method, with associated interpretation skills, enable an accurate analysis of the research results, which leads to an efficient and valuable reference by reducing unnecessary trial and error and data loss across all stages of research [6]. The empirical model has advantages in terms of relatively simple development and operation compared with the conceptual model but is unable to describe the causal relationship between the input and output data, and only describes the size of the factors expected to affect the result. In particular, the artificial neural network (ANN), which is categorized as an empirical model, has gained significant attention by obtaining excellent research results, with mainly positive assessments of its applicability. While some studies, which preferred conceptual models, argue that the dynamic relationship between the input and output data cannot be explained by empirical models [7]. However, ANNs have the advantages of high accuracy and fast prediction, when applied to flood prediction in particular [8,9]. ANN has been employed to predict air pollution such as fine dust, $NO_3$, and $O_3$ and reported good applicability [10]. Also, ANN has been used for simulation of total nitrogen and total phosphorus in wet land, and denoted a good of Nash–Sutcliffe efficiency [11]. ANNs have been applied to estimate suspended solids (SS) and biochemical oxygen demand (BOD) concentrations in the effluent of a wastewater treatment plant [12]. Not only these mentioned studies before, a number of previous studies have reported excellent results using ANNs that are sufficiently capable of resolving the nonlinear relationships in data [13–16]. In addition, some studies have shown that the detection of physical processing in a hydrological model is possible with ANNs, reducing skeptical views on the use of empirical models [17–19].

As described above, when determining the suitability of the model applicability for the establishment of watershed management plans based on the strengths and weaknesses of each type of model, for local governments or related fields that lack experts, operating a simpler and more intuitive model is more efficient than handling complex models; one can argue that the use of empirical models is more appropriate than conceptual models. In particular, as is the case in South Korea, there are a number of situations where there is a substantial shortage in manpower with only three or four public officials in charge of the pollution load management for each local government. Furthermore, despite numerous sites that require actual measurements, there are a number of ungauged watersheds due to physical limitations on available budgets and human resources. Thus, there is an increasing consensus on the suitability of empirical models.

Therefore, in this study, we adopted an ANN from the empirical model category to efficiently examine pollution load management to improve the difficulty and inconvenience of collecting pollution load data from ungauged watersheds. Unlike the methodologies

used in previous studies, a deep neural network (DNN) with an architecture of multiple layers was used. In particular, this study aims to estimate the pollution load of ungauged watersheds by introducing the convolution neural network (CNN), a DNN algorithm, and use hydrological images as input data for the CNN, thereby analyzing the applicability and model performance for pollution load estimation.

## 2. Materials and Methods

### 2.1. Study Area

For the pollution load estimation of ungauged watersheds, three watersheds were set as target areas in this study. As shown in Figure 1, the three study areas are JJ, HC, and BH, three of the Paldang watersheds located in the Seoul metropolitan area. Paldang watershed encompasses the seven municipalities surrounding Paldang Lake, which was artificially developed through the construction of Paldang Dam. Paldang Lake is located approximately 27.3 km from Seoul, the capital of South Korea. The Namhan River, Bukhan River, and Gyeongan Stream flow into the lake, which has a storage capacity of 244 million $m^3$, an effective storage capacity of 18 million $m^3$, watershed area of 23,800 $km^2$, and the retention time of 5.4 days.
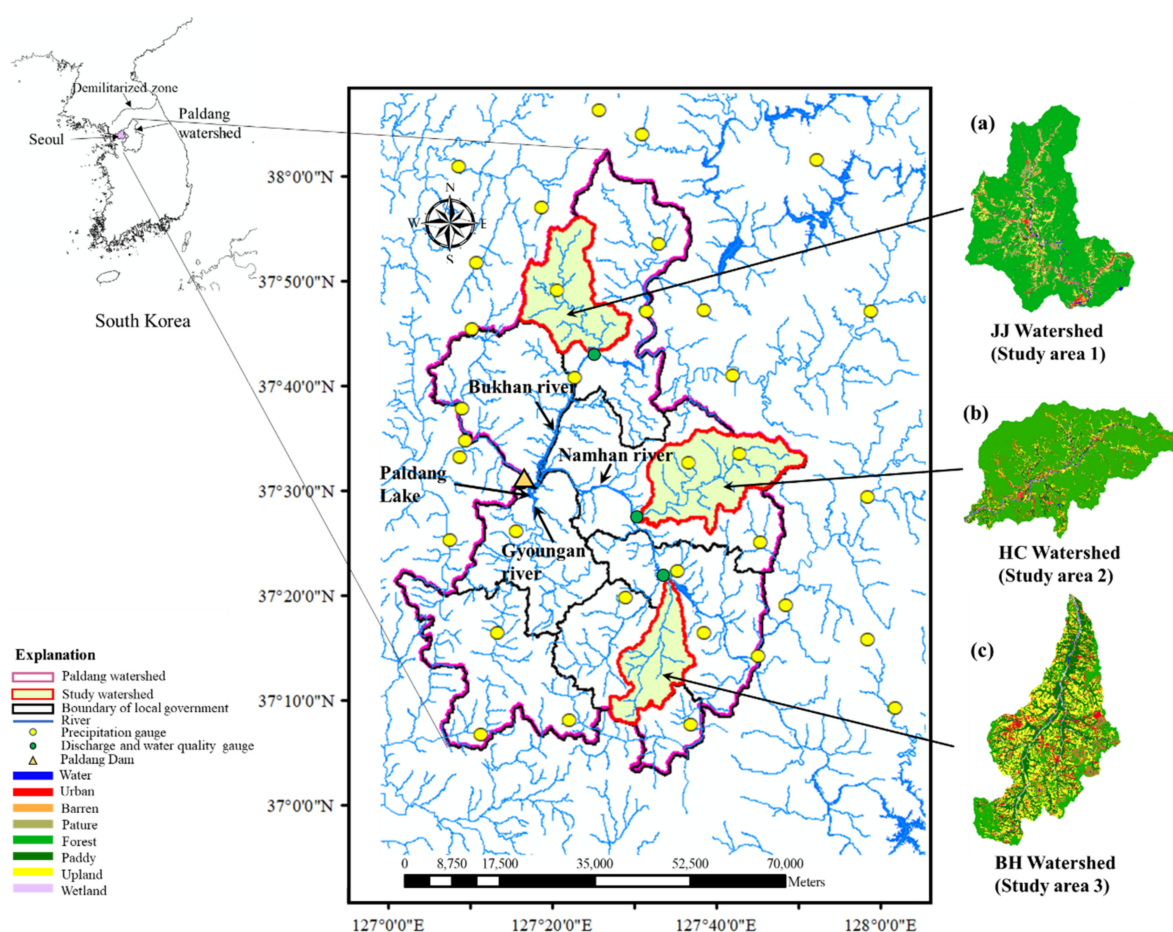


**Figure 1.** Location and land cover in the three study areas: (**a**) JJ watershed; (**b**) HC watershed; (**c**) BH watershed.

As described in the introduction, Paldang Lake is a water resource used by approximately 50% of the Korean population, and it is under special management by the national government of the Republic of Korea in terms of its water quality to ensure the stability of Paldang Lake in terms of its health and security. For the national target of Paldang Lake, the water quality standard is set to 1.0 ppm biochemical oxygen demand (BOD), and the water resource protection zone, Special Countermeasure Area, and waterfront

areas (such as a riparian buffer zone) have been established to enforce the highly strict regulations for the site. Furthermore, the Korean national government implemented a total maximum daily load (TMDL) system [20] in which BOD and total phosphorus (TP) were set as water quality management indicators in Paldang watershed. Therefore, while planned development activities are allowed at the site, owing to these regulations, there has been no local development, leading to little variation in the land cover of the area.

As shown in Figure 1a–c, the three study areas are mostly composed of forests and agricultural areas. JJ is a watershed that contains the Jojongcheon with a main stream length of 39 km (37°43′41.72″ N, 127°25′08.08″ E). The watershed has an average slope of 20.2° and an area of 260.6 km$^2$. It is a forest dominant area with forests accounting for most of the watershed (79.6%). HC is a watershed that contains the Heukcheon with a main stream length of 42.9 km without inflow from external rivers (37°27′37.69″ N, 127°31′55.39″ E). The watershed has an average slope of 18.4° and an area of 314.1 km$^2$. It is also a forest dominant area with forests accounting for most of the watershed (75.1%), similar to the JJ watershed. BH is a watershed that contains the Yanghwacheon with a main stream length of 32 km with almost no interference from external rivers (37°21′29.35″ N, 127°32′47.79″ E). The watershed has an average slope of 4.7° and an area of 181.1 km$^2$. It is a typical agricultural area mainly consisting of paddy fields (28.1%), fields (27.4%), and forests (23.0%) (Table 1).

**Table 1.** Summary of the three study areas.

| Study Area | | Land Cover | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Water | Urban | Barren | Pasture | Forest | Paddy | Upland | Wetland | Total |
| JJ Watershed (Study area 1) | Area (km$^2$) | 2.6 | 5.8 | 5.2 | 19.2 | 207.5 | 4.4 | 13.5 | 2.4 | 260.6 |
| | Proportion (%) | 1.0 | 2.2 | 2.0 | 7.4 | 79.6 | 1.7 | 5.2 | 0.9 | 100.0 |
| HC Watershed (Study area 2) | Area (km$^2$) | 2.3 | 6.5 | 2.9 | 22.5 | 235.8 | 20.8 | 19.6 | 3.7 | 314.1 |
| | Proportion (%) | 0.7 | 2.1 | 0.9 | 7.2 | 75.1 | 6.6 | 6.2 | 1.2 | 100.0 |
| BH Watershed (Study area 3) | Area (km$^2$) | 1.6 | 11.7 | 4.0 | 18.2 | 41.6 | 50.9 | 49.7 | 3.5 | 181.1 |
| | Proportion (%) | 0.9 | 6.5 | 2.2 | 10.1 | 23.0 | 28.1 | 27.4 | 1.9 | 100.0 |

*2.2. Data for the Study*

2.2.1. Data Collection

For the data required in this study, spatial data (i.e., land cover and soil maps), two hydrologic datasets (i.e., precipitation and discharge), and two water quality datasets (i.e., BOD and TP), were used for simplicity and convenience. Here, this study only used the BOD and TP for the water quality data was because they are used as the water quality management indicators in the TMDL.

All the data used in this study were obtained from the South Korean government; the land cover maps were obtained from the Environmental Geographic Information Service [21], the soil map from the Water Resources Management Information System [22], precipitation data from the Korea Meteorological Administration [23], and discharge, BOD, and TP data from the Water Environment Information System [24]. The data collection period was from 1 January 2010, to 31 December 2019; the precipitation data were collected daily (mm·day$^{-1}$) from 34 precipitation gauges located around the study area, and the discharge data were daily average data (m$^3$·s$^{-1}$·day$^{-1}$) collected from discharge gauges located at the terminus of each watershed. Water quality data were daily average data (mg·L$^{-1}$·day$^{-1}$) measured at the same points as the discharge gauge points, and the land cover and soil maps were in grid form with a resolution of 30 × 30 m and scale of 1:5000.

2.2.2. Data during the Study Period

The number of rainfall events during the study period was investigated, which were 554 events for JJ, 571 for HC, and 566 for BH. The annual average precipitation of JJ, HC, and BH watershed are 1104.5 mm, 1216.1 mm, and 1292.2 mm, respectively. The total average discharge of JJ, HC, and BH watershed was 23.3 m$^3$/s, 21.4 m$^3$/s, and 18.5 m$^3$/s,

respectively. The daily average discharge of JJ, HC, and BH watershed ranged from 0.9–299.5 m³/s, 0.2–603.2 m³/s, and 0.8–774.0 m³/s, respectively (Figure 2).
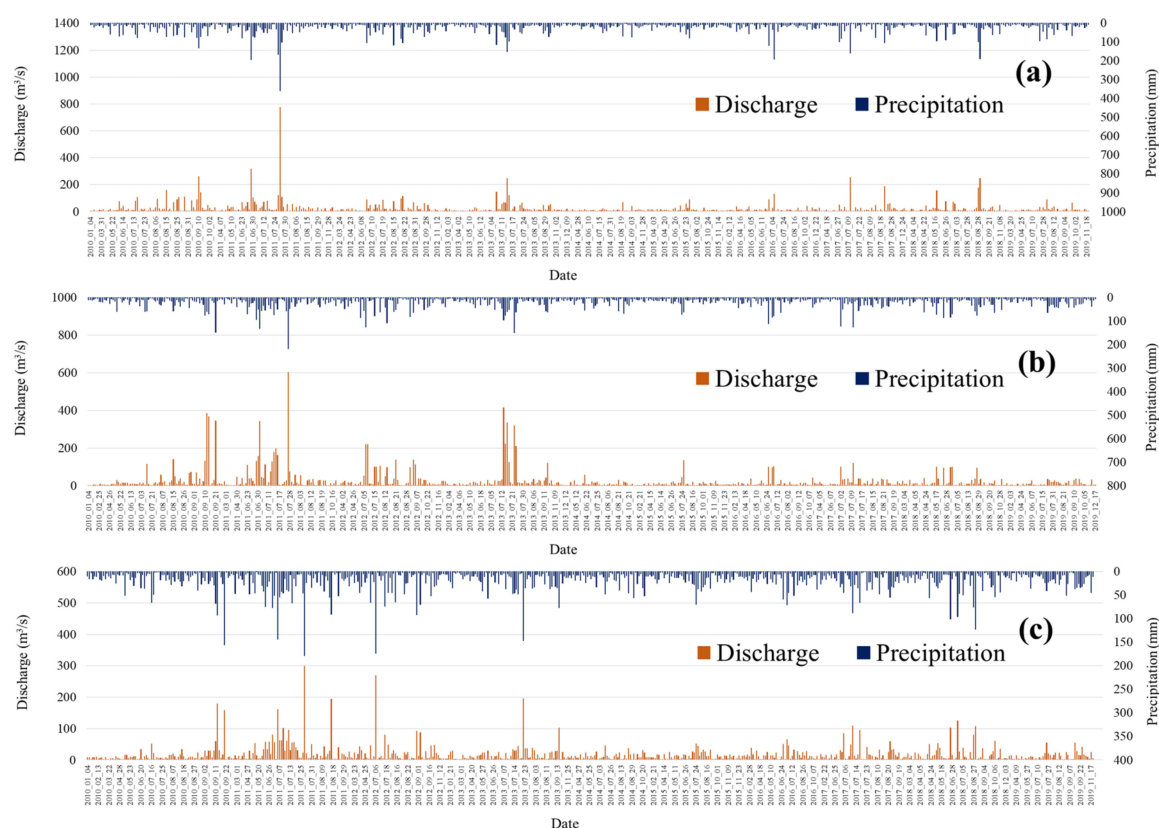


**Figure 2.** Variations in the discharge and precipitation according to the date of the rainfall event during the study period: (**a**) JJ watershed; (**b**) HC watershed; (**c**) BH watershed.

The variations in the daily average BOD concentration and daily average TP concentration in the JJ, HC, and BH watersheds corresponding to the same dates as in Figure 2 are shown in Figures 3 and 4, respectively. The daily average BOD concentration in the JJ watershed ranged from 0.1–3.8 mg·L$^{-1}$·day$^{-1}$; the BOD concentration tended to increase with increasing discharge (Figure 3a). The daily average BOD concentration in the HC and BH watershed ranged from 0.3–5.7 mg·L$^{-1}$·day$^{-1}$ and 0.4–16.7 mg·L$^{-1}$·day$^{-1}$, respectively (Figure 3b,c). Both of the BOD concentrations in the HC and BH watersheds also varied consistently with increases or decreases in the discharge. Out of the three study areas, the daily average BOD concentration in the BH watershed was observed to be higher than the other two study areas because, as listed in Table 1, the land use in the BH watershed for agricultural purposes is significantly higher at 55.5% compared with the other two study areas (JJ: 6.9%; HC: 12.8%).

The daily average TP concentration in the JJ and HC watershed ranged from 0.002–0.384 mg·L$^{-1}$·day$^{-1}$ and 0.003–0.378 mg·L$^{-1}$·day$^{-1}$, respectively (Figure 4a,b). The daily average TP concentration of BH watershed ranged from 0.159–1.865 mg·L$^{-1}$·day$^{-1}$, which was higher than that of the other two study areas (Figure 4c). For the same reason as the BOD concentration, these results are possibly because the agricultural land area in the BH watershed is larger than that in the other two study areas.

Figure 5 denotes the relationships between the discharge and pollutant loads. The daily BOD load corresponding to the rainfall event dates ranged from 12.4–104,195.6 kg/day for the JJ watershed and 8.6–297,054.1 kg/day for the HC watershed. The daily BOD load in the BH watershed ranged from 77.76–424,436.2 kg/day, which was higher than the daily BOD load range in the JJ and HC watersheds. The daily TP load in the JJ watershed

ranged from 0.3–10,529.2 kg/day, and the daily TP load in the HC watershed ranged from 0.1–19,699.4 kg/day. The daily TP load in the BH watershed was 12.4–48,277.0 kg/day, and similar to the daily BOD load, the TP load in the BH watershed was more than two-fold higher than that of the other study areas.

This is also thought to be due to the influence of higher TP concentrations than those in the other study areas. A *t*-test was performed to investigate the relationship between the BOD and TP loads and discharge in the JJ, HC, and BH watersheds, where the *p* values for all the study areas were not more than 0.001, indicating that the correlations between the BOD and TP loads and discharge in all of the study areas were highly significant.
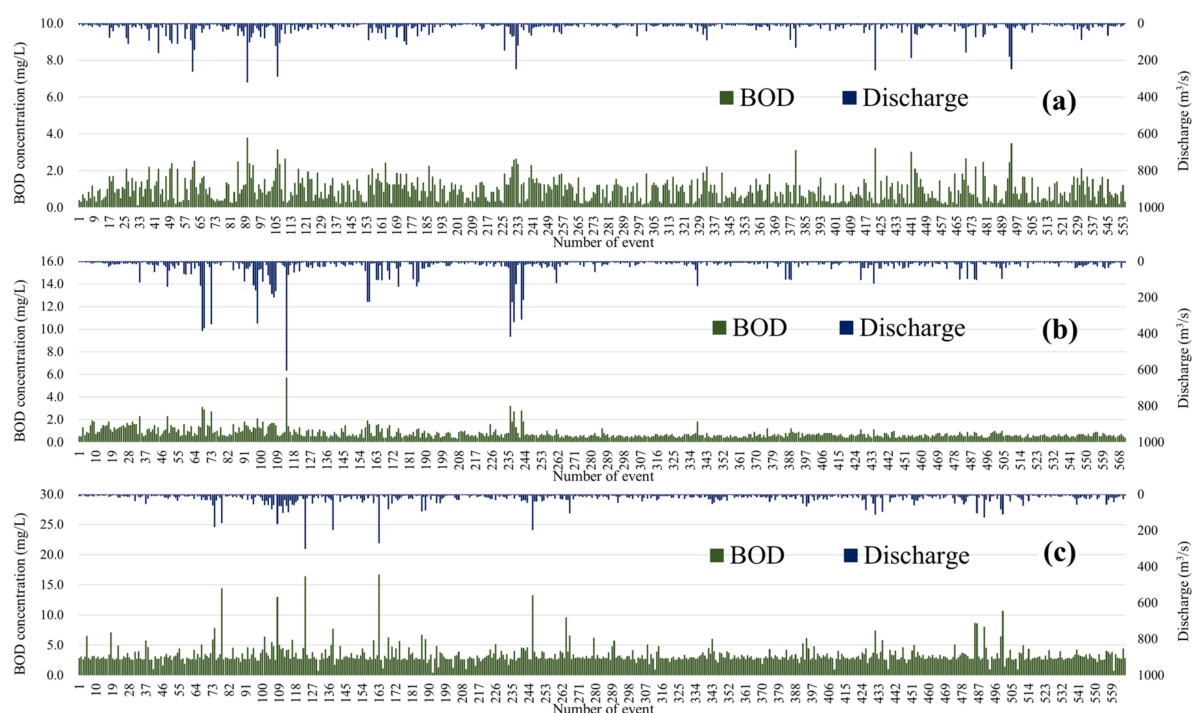


**Figure 3.** Variations in the discharge and BOD concentration according to the date of the rainfall event: (**a**) JJ watershed; (**b**) HC watershed; (**c**) BH watershed.
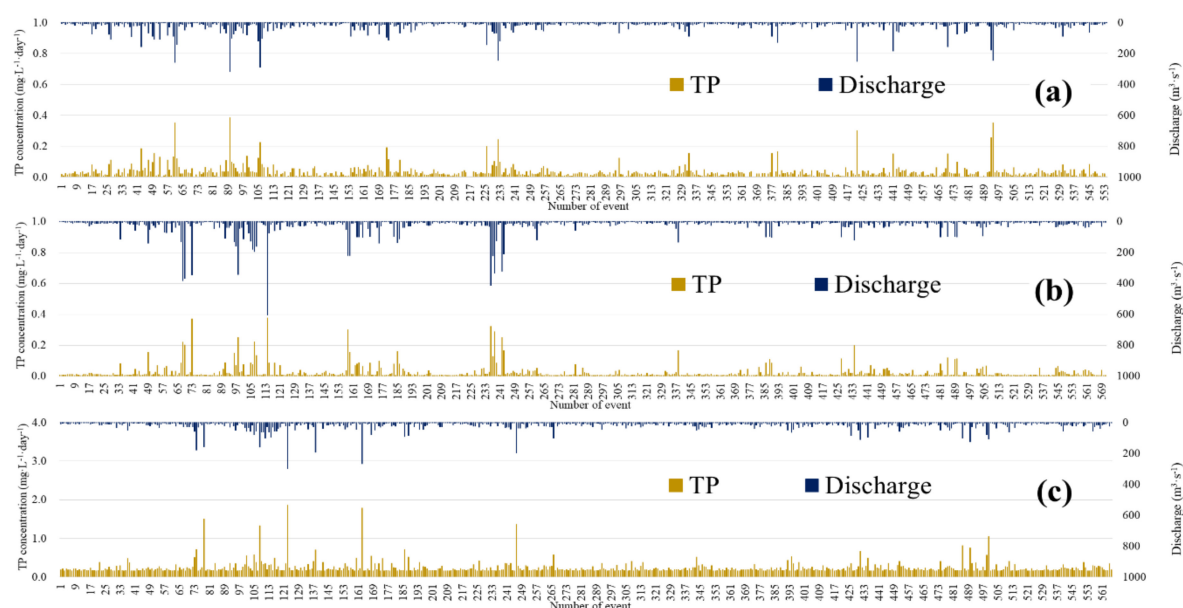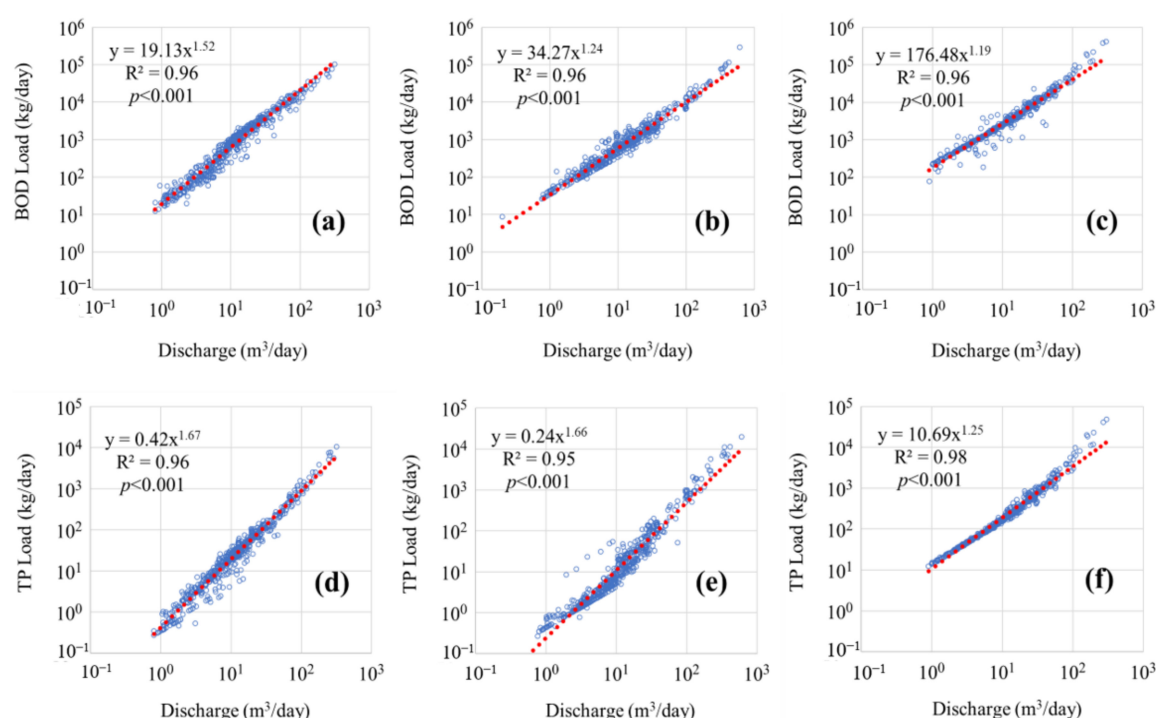


**Figure 4.** Variations in the discharge and TP concentration according to the date of the rainfall event: (**a**) JJ watershed; (**b**) HC watershed; (**c**) BH watershed.

**Figure 5.** Relationships between the discharge and pollution loads: (**a**) BOD of JJ watershed; (**b**) BOD of HC watershed; (**c**) BOD of BH watershed; (**d**) TP of JJ watershed; (**e**) TP of HC watershed; (**f**) TP of BH watershed.

### 2.3. Research Method

This study was conducted in two steps. As the CNN model was used for the estimation of the pollution load, the first step was to build an input dataset suitable for training the CNN model, and the CNN model was trained using the developed dataset. The second was estimating the BOD and TP loads of the ungauged watersheds.

#### 2.3.1. CNN Model Dataset

1. Building a Hydrological Image as a Feature for CNN Model Training

The CNN model is mainly used for classification problems, and in this study, RGB (red, green, and blue) photographs were used to train the CNN model. However, in this study, the hypothesis of using an empirical model cannot be established if ordinary RGB photos are used to train the CNN models. In other words, the pollution load shows responses according to major variables, such as the hydrological conditions and land cover; however, if RGB photos are used, the empirical relationship between the independent and dependent variables is not considered. Even if the prediction of the pollution load is highly accurate with the use of RGB photos, the relationship between the features in the training data and the target of prediction has no significance.

Therefore, in this study, to satisfy the causal relationship where the pollution loads are determined by variables, such as the hydrological conditions and land cover, the hydrological images proposed by Song [25] with hydrological attributes were used as training data for the CNN model. Song [25] defined a hydrological image as a set of grids with dimensionless hydrological attributes in a two-dimensional space for an arbitrary watershed, where the hydrological attributes of each grid are calculated using the hydrological curve number (CN) [26], published by the soil conservation service (SCS). The CN is also used in a number of recent studies for similar purposes. The CN is a dimensionless integer ranging from 1–100 derived using conditions such as the precipitation, soil characteristics, and land cover. The CN is used for the simulation of direct runoff or evaluations of hydrological effects [27,28].

As described above, each grid of a hydrological image represents the hydrological attributes and is defined as Q in Equations (1) and (2). This represents direct runoff as pro-

posed by the SCS, which has the same calculation method. The S parameter in Equation (3) is a parameter of Equation (2), representing the potential watershed storage determined by the CN. The SCS states that the CN should be assigned according to the four types of physical properties of soil, i.e., A, B, C, and D from the hydrologic soil group (HSG), and land cover, and provides CN values. However, as the topographic conditions in South Korea are different in terms of direct use for the CN values provided by the SCS, this study used CN values established by the Design Flood Estimation Techniques from the Ministry of Land, Transport and Maritime Affairs [29], South Korea. The Q and S parameters are calculated as

$$Q = \frac{(P - 0.2S)^2}{(P + 0.8S)} \quad (P \geq \text{ Ia}), \tag{1}$$

$$Q = 0 \quad (P < \text{ Ia}), \tag{2}$$

$$S = \frac{25,400}{\text{CN}} - 254. \tag{3}$$

Another parameter in Equations (1) and (2) is precipitation (P); the Thiessen polygon [30] has been traditionally used to estimate precipitation. However, discontinuity in the numerical variation occurs at the polygonal boundaries when using this method, causing distortion or discrepancy with respect to the actual precipitation. Therefore, in this study, using inverse distance weighting (IDW), the precipitation data were expressed as two-dimensional data in a grid format with a resolution of 30 × 30 m similar to the soil and land cover maps. IDW is an interpolation method based on Tobler's law [31] that can estimate continuous precipitation variations in grid units, which is able to compensate for the discontinuity problem of Thiessen polygons.

SCS recommends that adjustments to the CN in Equation (3) to consider the effect of precipitation, which accounts for the characteristics of direct runoff caused by antecedent precipitation. Therefore, SCS proposed adjustments to the CN as in Equations (4) and (5) by classifying the antecedent moisture condition (AMC), a concept based on the five-day antecedent cumulative precipitation, into AMC I, AMC II, and AMC III. Here, AMC I, AMC II, and AMC III indicate dry, normal, and wet conditions, respectively. These modified CNs can be expressed as

$$CN_I = \frac{4.2CN_{II}}{10 - 0.058CN_{II}} \quad \sum_{i=1}^{5} P_i < 12.7 \text{ (or 35.6) mm}, \tag{4}$$

$$CN_{III} = \frac{23CN_{II}}{10 + 0.13CN_{II}} \quad \sum_{i=1}^{5} P_i > 2.9 \text{ (or 53.3) mm}. \tag{5}$$

As shown in Equations (1) and (2), only hydrological images composed of fixed CNs are generated under non-precipitation conditions. Hydrological images composed of fixed CNs were excluded from this study because those images cannot serve as a variable, and only the hydrological images for days under a precipitation condition were collected and used for CNN model training. The hydrological images were built as two-dimensional images with 30 × 30 m resolution, identical to the land cover and soil maps, and were created as TIFF files. Here, the TIFF file format was used because creating a hydrological image in other file formats, such as BMP, JPEG, or PNG, the hydrological attribute values of all the grids do not follow the results calculated by Equations (1)–(3) but are recalculated into color values expressed from 0–255, resulting in information loss. The hydrological images were built using Python 3.7 [32], an open-source language, through the process illustrated in Figure 6.

2.  Building Target Data

The target data were the result corresponding to the given input data and were used to train the model within the neural network. In this study, supervised learning was used as the model training method, which is a method that continuously reduces the error between

the predicted data and the target data by improving the weight of the neural network with iterations during training. In this study, the target data were actual observed data, representing the pollution load of the BOD and TP; the pollution load data corresponded to the same dates as the hydrological images. The target data consisted of the BOD and TP loads of the JJ, HC, and BH watersheds, and were developed in a CSV file format for recognition in the CNN model. Here, the pollution load was calculated using discharge and water quality data.
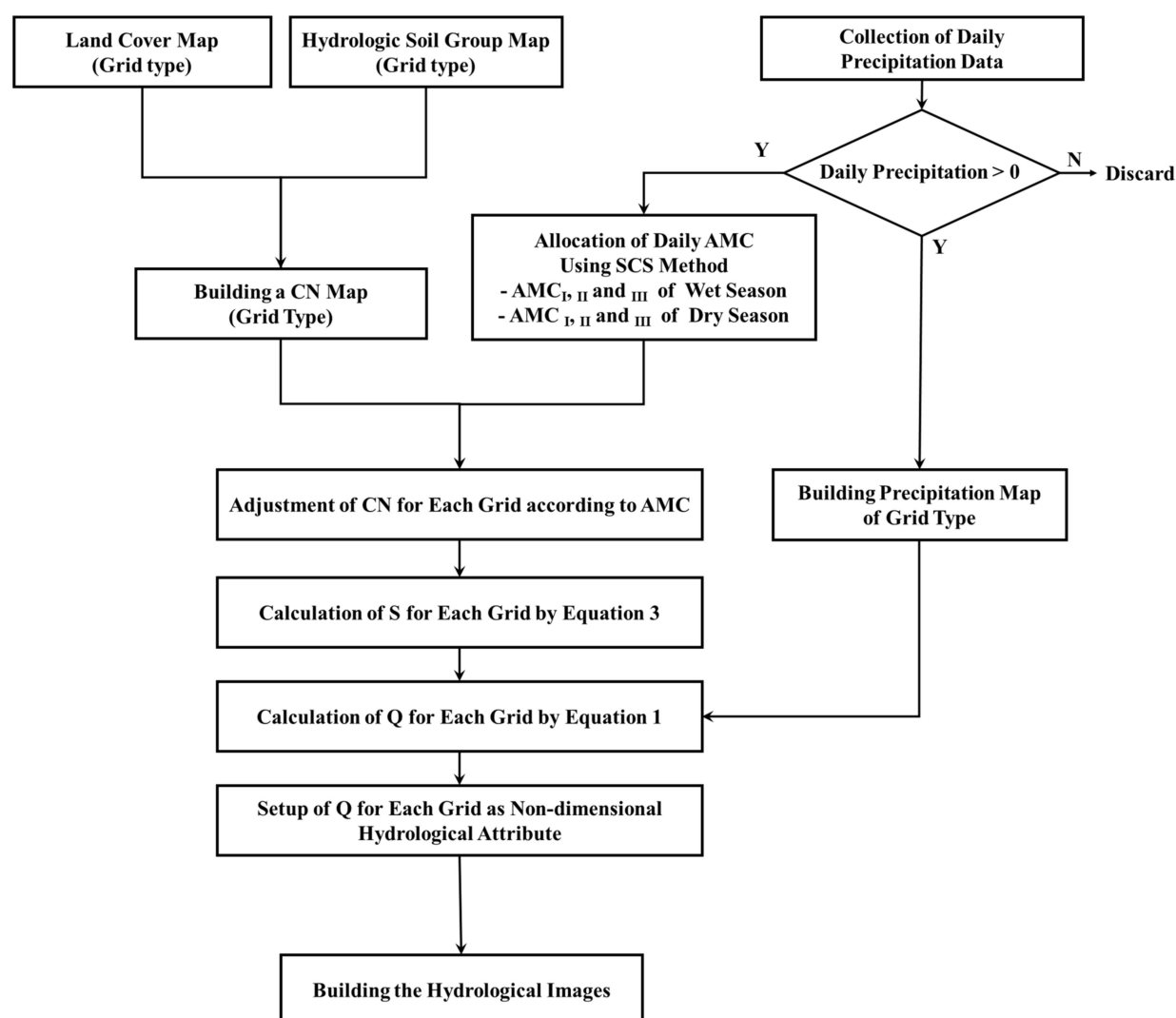


**Figure 6.** Schematic for the building process of a hydrological image.

3. Establishing the Dataset for CNN Model Training and Testing

Datasets are required to train CNN models; in this study, the datasets were built using the hydrological images and target data described above. In general, datasets in the CNN field are divided into input dataset and test datasets. The input datasets are divided into training and validation datasets while the test dataset is used to verify the prediction performance of the CNN model. The CNN model in this study was designed for six cases, where cases 1–3 were a model for the simulation of the BOD load and cases 4–6 were a model for the simulation of the TP load. As listed in Table 2, two watershed datasets were used as the input datasets for CNN model training, and the remaining watershed dataset was used as a test dataset, resulting in the final settings for the pollution load estimation of the ungauged watersheds.

**Table 2.** Dataset setting.

| Model | Image Data | Target Data | Dataset Classification |
|---|---|---|---|
| Case 1 | Hydrological images of HC and BH | BOD Load of HC and BH | Input Dataset |
| | Hydrological images of JJ | BOD Load of JJ | Test Dataset |
| Case 2 | Hydrological images of JJ and BH | BOD Load of HC and BH | Input Dataset |
| | Hydrological images of HC | BOD Load of JJ | Test Dataset |
| Case 3 | Hydrological images of JJ and HC | BOD Load of HC and BH | Input Dataset |
| | Hydrological images of BH | BOD Load of JJ | Test Dataset |
| Case 4 | Hydrological images of HC and BH | TP Load of HC and BH | Input Dataset |
| | Hydrological images of JJ | TP Load of JJ | Test Dataset |
| Case 5 | Hydrological images of JJ and BH | TP Load of JJ and BH | Input Dataset |
| | Hydrological images of HC | TP Load of HC | Test Dataset |
| Case 6 | Hydrological images of JJ and HC | TP Load of JJ and HC | Input Dataset |
| | Hydrological images of BH | TP Load of BH | Test Dataset |

### 2.3.2. CNN Model Structure

The CNN model mimics the visual processing of biological organisms with visual organs, and LeCun et al. [33] first proposed the development of the CNN model. The CNN model produces highly reliable results and has superior efficiency, as well as a reputation as a core technology for research and applications for rapid image classification. Recently, the model has been applied not only in image classification, but also in speech recognition and image semantic segmentation [34].

The structure of the CNN model can be largely divided into two parts. The first part of the CNN model appears at the front of the CNN model architecture, consisting of the repetition of the convolution layer and pulling layer while the second part appears at the back of the CNN model, consisting of flatten layers serving as a connection between the first and second parts, followed by dense layers and output layers. The first part is also known as a convolution layer (CL), which manages the core functions of the CNN. While maintaining the structure of the I/O data for each layer and spatial information for the images, this part effectively recognizes the features with adjacent images. The second part is known as a fully connected layer (FC), representing a deep neural network (DNN) composed of multiple dense layers between a flatten layer and an output layer, which derives simulation results [35,36].

In this study, the architecture of the CNN model was designed as five alternate repetitions of two convolution layers and one pooling layer for the CL part, as shown in Figure 7. In addition, CNN uses a two-dimensional plane function image filter known as a kernel, which is located in each convolution layer and plays a role as a parameter to identify features of the image. The kernels are a square matrix with arrangements of parameters, and their size and number can be adjusted arbitrarily according to the user's purpose. In this study, the size of all kernels was set to $3 \times 3$, and the numbers of 32, 32, 64, 64, 128, 128, 256, 256, 512, and 512 were allocated in the order of each convolution layer as shown in Figure 3. In addition, the kernel searches for images and extracts features while moving from left to right and top to bottom from the top of the image; in this case, the movement spacing of the kernel is referred to as stride. In this study, the stride of the first to third convolution layer was set to 2, and the stride of all other convolution layers was set to 1. The pooling layer is a sub-sampling technique that only leaves important information while reducing the size of the data received from the convolution layer. In addition, the convolution layer sets the activation function, which was set as a rectified linear unit (ReLu) function in this study. The ReLu function can prevent gradient vanishing in which the

gradient value disappears, such that it has been used in several prior studies reporting high optimization efficiency of the model [37,38].
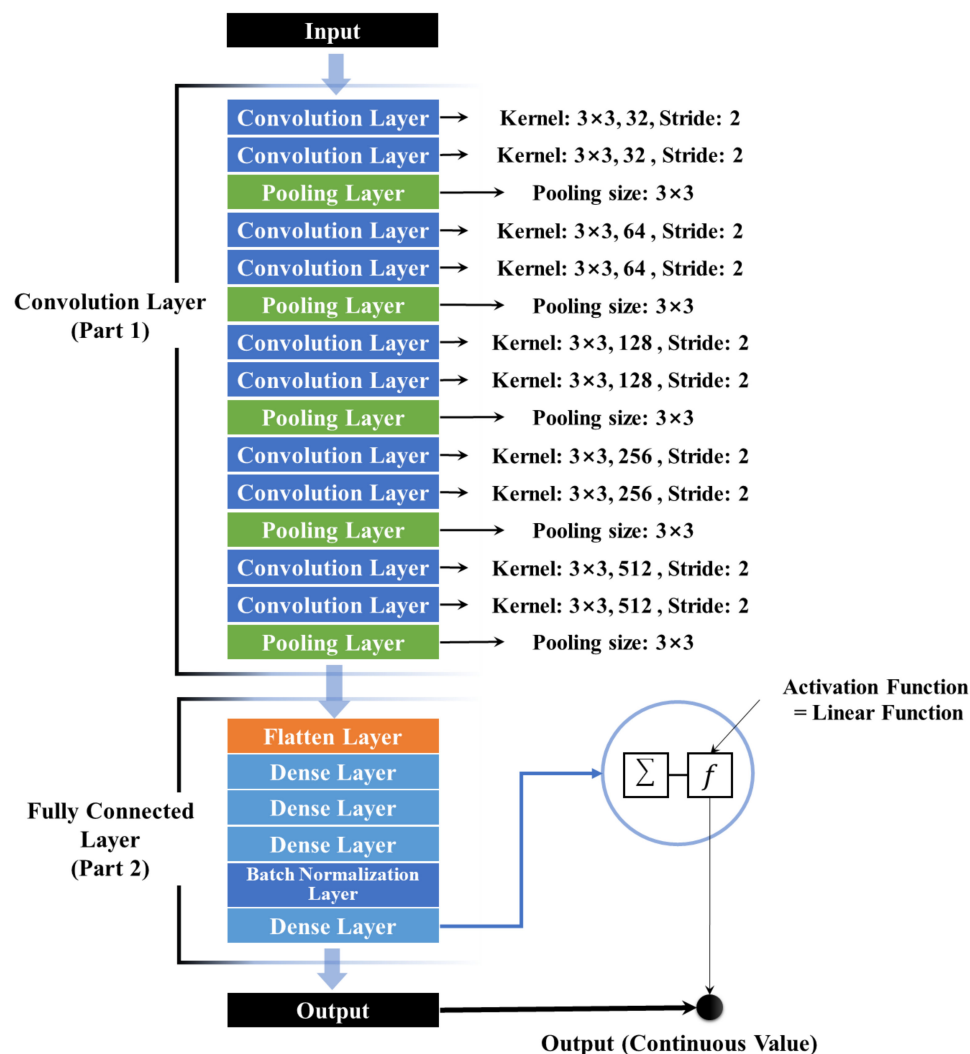


**Figure 7.** CNN model architecture for the simulation of the continuous variables in this study.

The pooling layer, with the max and average pooling method, can efficiently utilize the computer memory and reduces computation information, resulting in the effect of preventing overfitting. As average pooling methods cause information loss, the max pooling method was used in this study. In addition, the pooling size can be arbitrarily set as in the case for the kernels; therefore, the pooling size was set to $3 \times 3$, identical to the kernel size.

The difference between the CNN model used in this study and that of previous studies is the FC part. The CNN model employs a softmax or sigmoid function as the activation function in the last dense layer of the FC because it is designed to solve the classification problem. However, this is not suitable for the simulation of unspecified continuous variables as is the case in this study. A linear function frequently used in the regression model was applied to the last dense layer of the FC in this study, unlike previous studies. For all environments of the CNN model design, implementation, and operation, Keras [39], a machine learning library released by Google was used. Keras is a Python-based high-level deep neural network application programming interface (DNN API), with a Tensorflow [40] backend, which supports all special functions of Tensorflow and is highly flexible in implementing DNN models. Keras facilitates rapid experiments with the efficient use of the CPU and GPU.

### 2.3.3. Hyper-Parameter Setting of the CNN Model

In the field of artificial neural networks, there are various types of hyper-parameters. Depending on the hyper-parameter implementation method, the model prediction results can either show excellent performance or results that are not satisfactory with respect to the expected performance, thus indicating the importance of hyper-parameter implementation. In the field of neural networks, the hyper-parameters are not the main parameters for tuning or optimization, but they are set by the user with a priori knowledge or automatically set through the mechanism of an external model. For the CNN model in this study, the hyper-parameters were directly set without using an external model. The hyper-parameters of the CNN model in this study included the LOSS function, optimizer for model optimization, learning rate, number of layers and nodes of the FC, and epoch, which is the number of model training processes.

For learning method of the CNN model, supervised learning was used, in which the model is optimized by minimizing the error, which is the difference between the predicted value derived during training and the target value. To reduce the error, the values of the weights within the neural network were continuously updated. The error is referred to as the LOSS in neural networks, where the optimization of the model using supervised learning can be described as minimizing this LOSS. Functions, such as binary crossentropy, categorical crossentropy, and sparce categorical crossentropy, have been mainly used to calculate the LOSS of the CNN model. However, these functions are suitable for classification problems and are not adequate for models predicting continuous variables, as is the case in this study. Therefore, in this study, mean square error (MSE), which is mainly used in regression analysis, as shown in Equation (6), was used; through this, the optimization of the model was determined. The predictive performance of the model must be evaluated during model training; as a metric for the evaluation, the mean absolute error (MAE) was used in this study, as shown in Equation (7):

$$MSE = \frac{1}{n} \sum_{i=1}^{n} \left( y_i' - y_i \right)^2, \tag{6}$$

$$MAE = \frac{1}{n} \sum_{i=1}^{n} y_i' - y_i, \tag{7}$$

where, $y_i$ and $y_i'$ denote the observed value and predicted value, respectively, and MSE and MAE indicate that an increased number of values that converge to zero yields higher model performance. Even if the result of the LOSS calculation is good or excellent, it is the result of only using the training dataset for model training purposes; thus, it is necessary to perform validation for the data that the model has not been trained with (or unobserved data). The dataset used for this purpose was the validation dataset mentioned above, along with the LOSS of the training dataset and Equations (6) and (7); the functions used for prediction performance were used in the same manner. In addition, the names were separated to distinguish between the results of the training dataset and those of the validation dataset. The results of the training dataset were set to LOSS and MAE, and the results of the validation dataset were set to validation LOSS and validation MAE.

For the model optimization algorithm, the stochastic gradient descent (SGD) [41] method is regarded as the most basic methodology. To improve the optimization speed and accuracy by addressing the shortcomings of the SGD, previous studies have developed momentum [42], Nesterov accelerated gradient [43], Adagrad [44], Nadam [45], AdaDelta [46], RMSProp [47], and Adam [48]. In this study, ADAMAX [48] was used, an algorithm proposed as an extension of ADAM [48]. ADAMAX is an algorithm based on ADAM's update rule, which exhibits the best performance in a CNN model. ADAMAX shows superior performance to other algorithms under noisy conditions and has the characteristic of fast convergence when the data variation is large. Hence, ADAMAX was an appropriate to selection as the optimization algorithm for the CNN model in this study because the characteristics of the pollution loads to be simulated vary significantly depending on the hydrological phenomena, as well as substantial variations in the parameters and noise

due to natural conditions. In addition, the learning rate was set to 0.0001 and the epoch was set to 1000. The FC layer was designed with an architecture consisting of three dense layers and one batch normalization layer to avoid overfitting, followed by one dense layer. The number of nodes in the three dense layers was 1024, and the number of nodes in the last dense layer was one. In this study, CNN model construction and training were performed on a desktop with an Intel® Core™ i9-9900K CPU @ 3.60 GHz, 32 GB RAM, and an NDIVIA® GEFORCE RTX™ 2080Ti GPU with a Windows OS.

### 2.4. Evaluation of the CNN Model

For the evaluation of the BOD and TP load predictions of the ungauged watersheds with the CNN model, the performance of the CNN model was assessed by comparing the predicted value and observed value. As listed in Table 2, the models for evaluation were six models including cases 1–6 and cases 1–3, which were the models used to estimate the BOD load and cases 4–6 were the models used to estimate the TP load. For evaluation of this CNN model, four methods were used: Pearson correlation coefficient (*r*), Nash–Sutcliffe efficiency (NSE), root-mean-square error (RMSE), and mean absolute percent error (MAPE).

The Pearson correlation coefficient (*r*) is a method that examines the linearity between the observed value and predicted value in the range from –1 to 1. Values of r that are closer to 1 suggest stronger linearity between the predicted value and observed value. The Pearson correlation coefficient is calculated as

$$\gamma = \frac{\sum_{i=1}^{n}(y_i - \bar{y})\left(y_i' - \overline{y_i'}\right)}{\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2\left(y_i' - \overline{y_i'}\right)^2}}. \tag{8}$$

The NSE ranges from 0 to 1; if the NSE value is 1, the model result is in perfect agreement with the observed value. The NSE is calculated as

$$\text{NSE} = 1 - \frac{\sum_{i=1}^{n}(y_i - y_i')^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}. \tag{9}$$

The RMSE is a quantitative metric showing whether the predicted value of the model is close to the observed value; the range is expressed from 0 to infinity within the range of the data. As the RMSE usually follows the unit of the target to be simulated, the unit of the RMSE in this study is kg/day, which is the unit of the pollution load. The evaluation criteria for the RMSE itself are not clearly provided, but values closer to 0 indicate model results that are closer to the observed values. The RMSE is calculated as

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - y_i')^2}. \tag{10}$$

The MAPE is presented as a percentage. The key point of MAPE is to express the size of the error as a percentage rather than a quantification of the error as is the case for RMSE. Therefore, as MAPE is presented in terms of a percentage in the range from 0–100, the results can be easily interpreted with MAPE. For the evaluation of MAPE, values closer to 0 indicate better model performance, but there is no clear evaluation criterion for model performance. MAPE is calculated as

$$\text{MAPE} = \frac{100}{n}\sum_{i=1}^{n}\left|\frac{yi - y_i'}{yi}\right|, \tag{11}$$

where $y_i$ and $y_i'$ are the observed and predicted values, respectively, and $\bar{y}$ and $\bar{y}'$ represent the averages of the observed and predicted values, respectively.

## 3. Results and Discussion

### 3.1. Hydrological Image Generation and Image Datasets Construction Results

As mentioned in Section 2.3.1 for the method of generating hydrological images, the number of hydrological images is equal to the number of rainfall event during the study period, and the number of hydrological images created during the study period was 554 for the JJ watershed, 571 for the HC watershed, and 566 for the BH watershed. The sizes of the hydrological images in a grid format with a resolution of 30 × 30 m for the JJ, HC, and BH watersheds were 670 × 819, 1014 × 632, and 513 × 872, respectively. Figure 8 shows part of the results for the generated hydrological images.
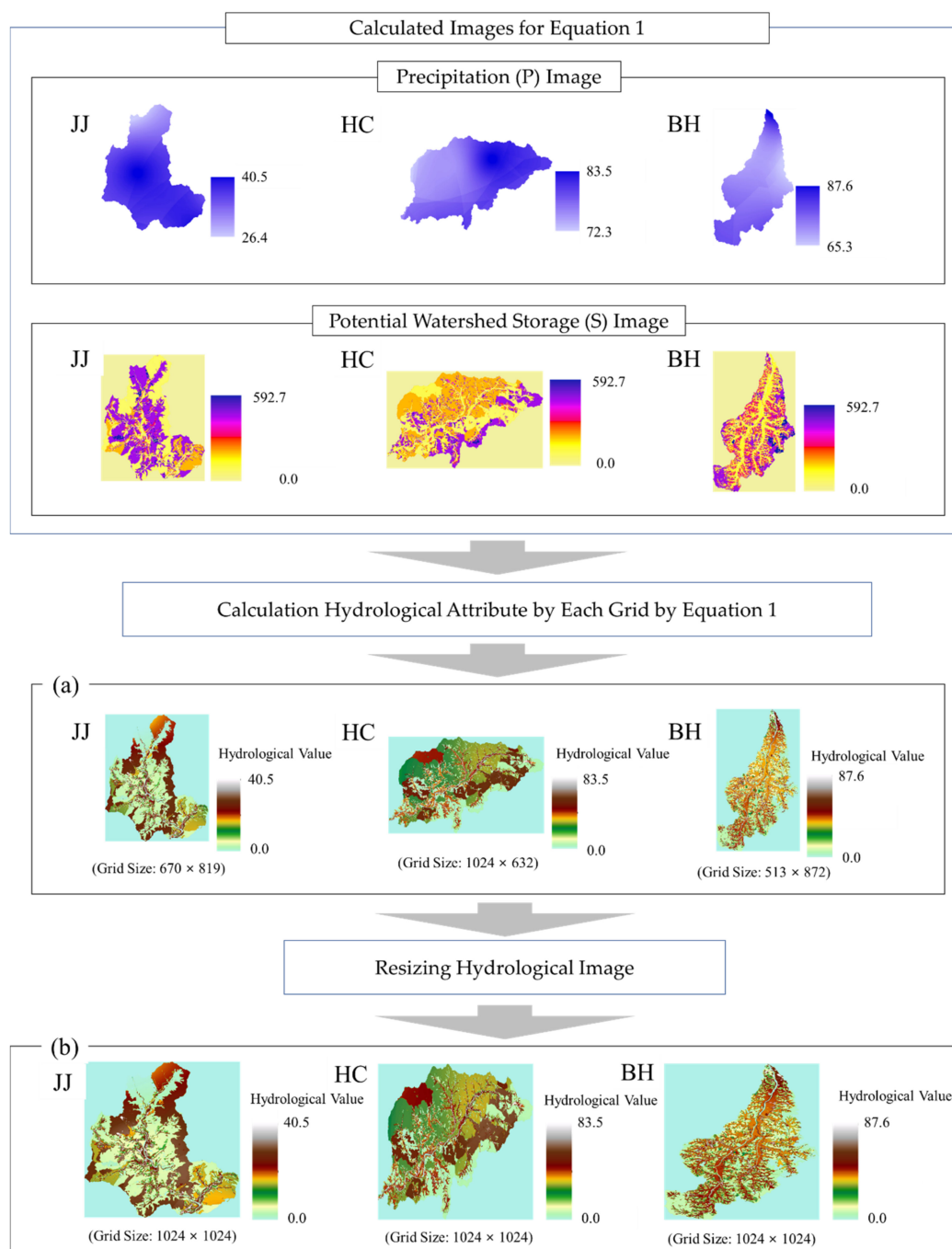


**Figure 8.** Scheme of generating hydrological images and resized hydrological images: (**a**) the initial generated hydrological images of the JJ (left), HC (middle), and BH (right) watersheds; (**b**) the resized hydrological images for the JJ (left), HC (middle), and BH (right) watershed at a grid size of 1024 × 1024.

As shown in Figure 8, the hydrological images generated in this study take the shape of an image as the color-code of the grid, which indicates that each grid has a hydrological attribute value; according to each value, hydrological attribute is represented by color-code (Figure 8a).

The CNN model requires images of the same size during training. However, as shown in Figure 8a, the images of the three study areas had different sizes and therefore were not suitable for CNN model training. Hence, the image should be resized to identical sizes for each watershed. In this study, as shown in Figure 8b, the images were resized into square images based on a 1024 × 1024 grid, i.e., the largest size of the sides among the hydrological images of the three watersheds. For the resizing, linear interpolation by Pillow [49], an image module of Python, was used.

For training and testing of the CNN models, the input dataset and test dataset were classified as listed in Table 3 using hydrological images and target data. There is no specified criterion for the ratio used to divided training datasets and validation datasets in the CNN model; however, in previous studies, the ratio between the training dataset and the validation dataset was set to 7:3 or 8:2 [50–52]. Accordingly, in this study, the datasets for training and validation of the model were divided with a ratio of 7:3, as listed in Table 3, but the test datasets were composed of data for the watersheds for simulation only to fulfill the purpose of pollution load predictions in ungauged watersheds; these were not included in the input datasets. In this manner, the number of training datasets and validation datasets for the input datasets in cases 1 and 4 were 796 and 341, respectively, and the number of test datasets was 554. The number of training datasets and validation datasets for the input datasets in cases 2 and 5 were 784 and 336, respectively, and the number of test datasets was 571. For the input datasets in cases 3 and 6, the number of training datasets, validation datasets, and test datasets was 788, 337, and 566, respectively.

**Table 3.** Datasets for the CNN model used in this study.

| Model | Target | Dataset | | Number of Data | Note |
|-------|--------|---------|---|----------------|------|
| Case 1 | BOD Load | Input Dataset | Training Dataset | 796 | HC–BH |
| | | | Validation Dataset | 341 | HC–BH |
| | | Test Dataset | Test Dataset | 554 | JJ (Entire study period) |
| Case 2 | | Input Dataset | Training Dataset | 784 | JJ–BH |
| | | | Validation Dataset | 336 | JJ–BH |
| | | Test Dataset | Test Date Set | 571 | HC (Entire study period) |
| Case 3 | | Input Dataset | Training Dataset | 788 | HC–JJ |
| | | | Validation Dataset | 337 | HC–JJ |
| | | Test Dataset | Test Date Set | 566 | BH (Entire study period) |
| Case 4 | TP Load | Input Dataset | Training Dataset | 796 | HC–BH |
| | | | Validation Dataset | 341 | HC–BH |
| | | Test Dataset | Test Date Set | 554 | JJ (Entire study period) |
| Case 5 | | Input Dataset | Training Dataset | 784 | JJ–BH |
| | | | Validation Dataset | 336 | JJ–BH |
| | | Test Dataset | Test Date Set | 571 | HC (Entire study period) |
| Case 6 | | Input Dataset | Training Dataset | 788 | HC–JJ |
| | | | Validation Dataset | 337 | HC–JJ |
| | | Test Dataset | Test Date Set | 566 | BH (Entire study period) |

### 3.2. CNN Model Architecture and Training Results

The architecture of the CNN model in this study was composed of five alternating repetitions of two convolution layers and one pooling layer, with one flatten layer, three dense layers, and one batch normalization layer, finally followed by one dense layer such as Table 4. In this mode, the number of total parameters was 15,205,601 [1], the number of trainable parameters was 15,203,553 [2], and the number of non-trainable parameters owing to the batch normalization layer was 2048 [3].

**Table 4.** Summary of CNN model structure used in this study.

| Convolution Layer | Output Shape (Raw_Size, Column_Size, Image_Channel) | Number of Parameter | Activation Function |
|---|---|---|---|
| Conv2D 1 | 507, 507, 32 | 320 | ReLu |
| Conv2D 2 | 254, 254, 32 | 9248 | |
| MaxPooling 1 | 127, 127, 32 | 0 | |
| Conv2D 3 | 64, 64, 64 | 18,496 | ReLu |
| Conv2D 4 | 64, 64, 64 | 36,928 | |
| MaxPooling 2 | 32, 32, 64 | 0 | |
| Conv2D 5 | 32, 32, 128 | 73,856 | ReLu |
| Conv2D 6 | 32, 32, 128 | 147,584 | |
| MaxPooling 3 | 16, 16, 128 | 0 | |
| Conv2D 7 | 16, 16, 256 | 295,168 | ReLu |
| Conv2D 8 | 16, 16, 256 | 590,080 | |
| MaxPooling 4 | 8, 8, 256 | 0 | |
| Conv2D 9 | 8, 8, 512 | 1,180,160 | ReLu |
| Conv2D 10 | 8, 8, 512 | 2,359,808 | |
| MaxPooling_5 | 4, 4, 512 | 0 | |
| **Fully Connected Layer** | **Output Shape (Number of Node)** | **Parameter** | **Activation Function** |
| Flatten Layer | 8192 | 0 | |
| Dense Layer 1 | 1024 | 8,389,632 | ReLu |
| Dense Layer 2 | 1024 | 1,049,600 | ReLu |
| Dense Layer 3 | 1024 | 1,049,600 | ReLu |
| Batch Normalization Layer | 1024 | 4096 | ReLu |
| Dense Layer 4 | 1 | 1025 | Liner |

[1] Total Parameters: 15,205,601; [2] Trainable Parameters: 15,203,553; [3] Non-trainable Parameters: 2048; Optimizer Algorithm: ADAMAX; Learning ratio: $1 \times 10^{-4}$; Epoch: 1000 iterations.

Figure 9 shows the training results of the CNN model. There is no specified criterion for evaluating the training results of an artificial neural network model; however, in general, as the epoch progresses, when the graph of the learning result (loss, validation loss, MAE, and validation MAE) exhibits a similar shape to an exponential function with a base of zero or less as the corresponding values becomes closer to 0 on the *y*-axis, sufficient model training has been performed.

Examining Figure 9, the loss, validation loss, MAE, and validation MAE of all cases showed a sharp decrease in the early epoch progress, and then gradually decreased. As mentioned above, loss and MAE are indicators of the result of model training by training datasets while validation loss and validation MAE are indicators of the generalization result of the model by validation datasets. As shown in Figure 9, all of the graphs show a large amplitude, indicating an unstable training state; however, all values are heading for the zero, indicating that the training and generalization of the model have been sufficiently performed.
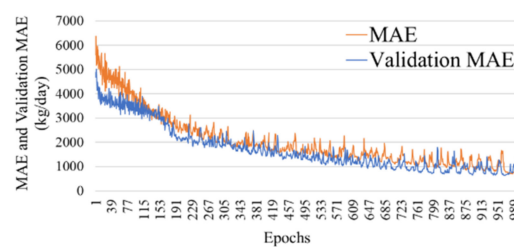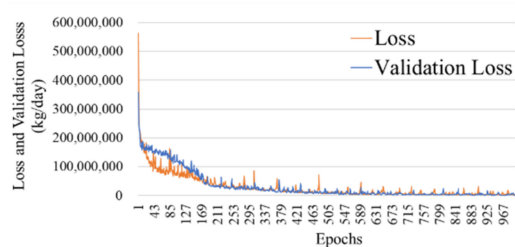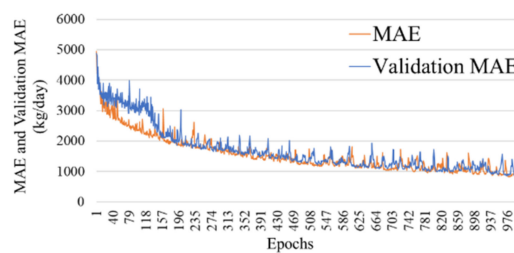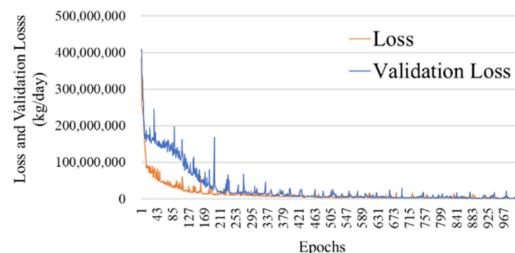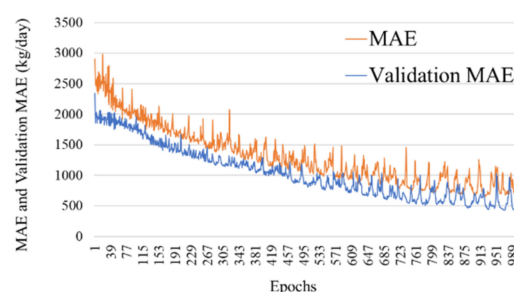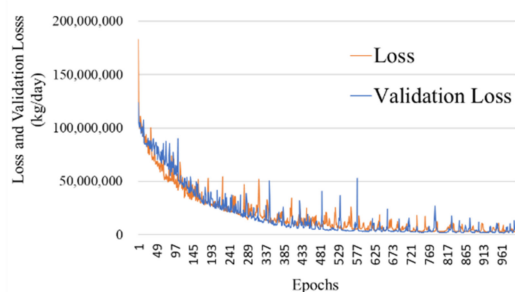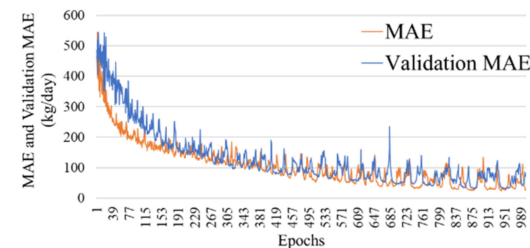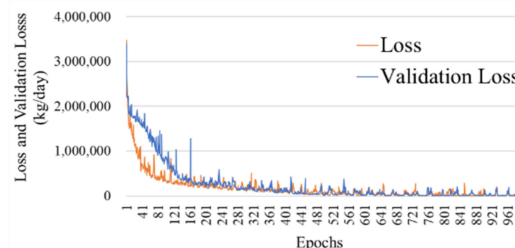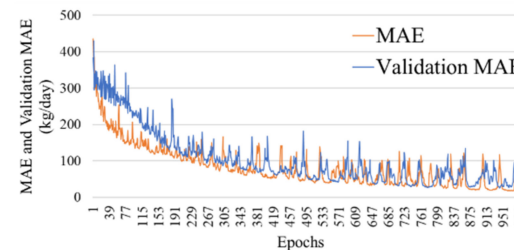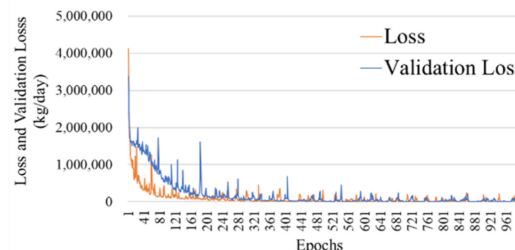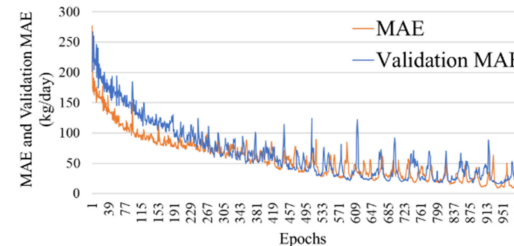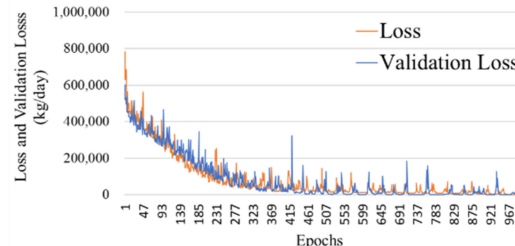
**Figure 9.** Training results for the CNN models in this study: (**a**) CNN models for BOD load prediction; (**b**) CNN models for TP load prediction.

*3.3. Prediction Results of the CNN Model and Model Evaluation*

Using the trained CNN models for cases 1–6, the BOD and TP loads were predicted by assuming that the three study watersheds were ungauged watershed, which is the purpose of this study. The BOD load was predicted using the CNN models for cases 1–3, and the TP load was predicted using the CNN models for cases 4–6. Figure 10a shows the relationship between the predicted BOD load and observed value while Figure 10b shows the relationship between the predicted TP load and observed value. Cases 1–3 represent the BOD loads of the JJ, HC, and BH watersheds, respectively (Figure 10a), and cases 4–6 represent the TP loads in the JJ, HC, and BH watersheds, respectively (Figure 10b). In all cases, the scatter plots of the predicted and observed values are shown with a 1:1 line, indicating that the predicted and observed values demonstrate a linear relationship, which confirms that the predicted values agree with the observed values.
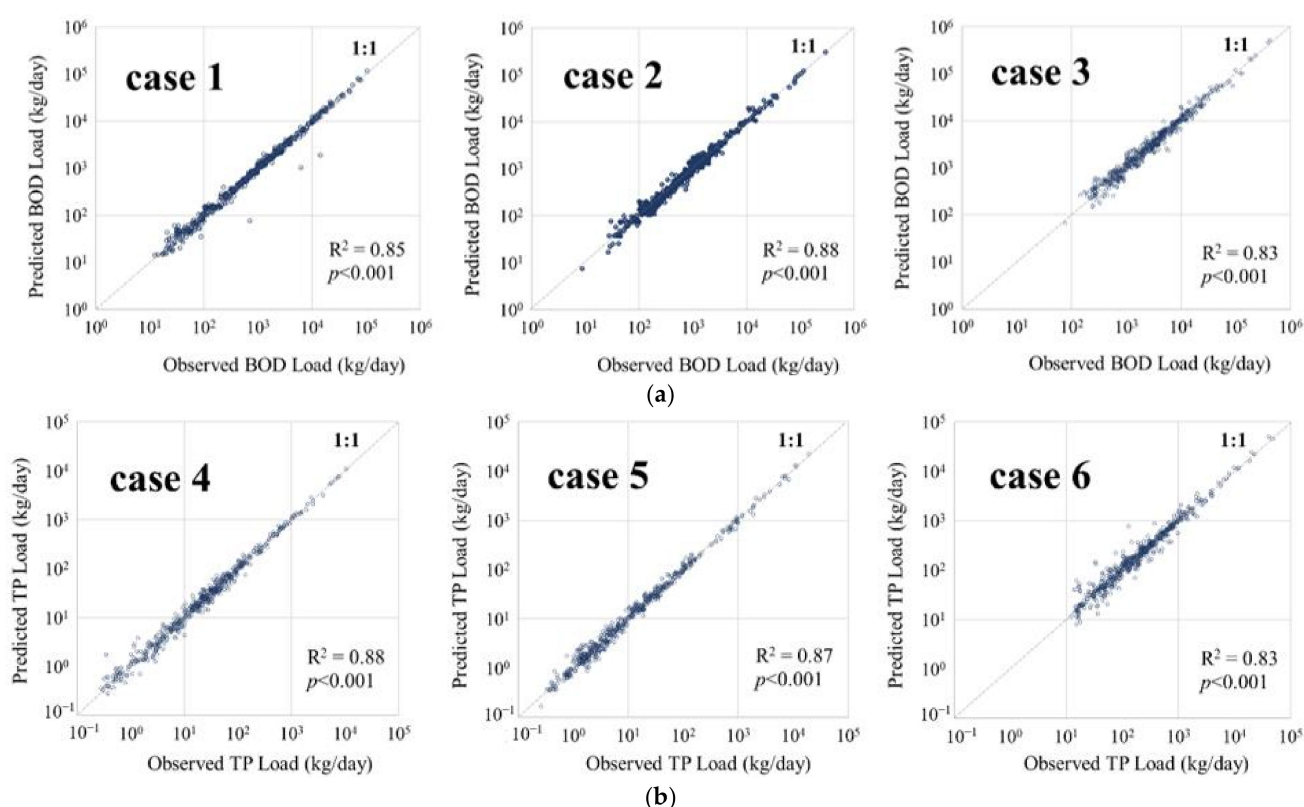


**Figure 10.** Scatter plot between the predicted values of the CNN model and observed values: (**a**) BOD load; (**b**) TP load.

In this study, $r$, NSE, RMSE, and MAPE were used to evaluate the performance of the CNN model, whose results are listed in Table 5. For the evaluation of the model using $r$, the model was weak when the value of $r$ ranged 0.1–0.3, moderate from 0.4–0.6, strong from 0.7–0.9, and perfect when the value was 1 [53]. For the NSE, the model was moderate at NSE values from 0.6–0.8 and good at values above 0.8 [54]. In all cases, $r$ showed a strong correlation of 0.9, and all cases were evaluated as strong, while all NSE values were 0.9, such that all cases were good. In the evaluation using the RMSE, in cases 1–3 for the BOD load simulation, the values were 1031.1, 1152.2, and 5376.5 kg·day$^{-1}$, respectively, and the RMSE value was the largest for case 3, which represented the BH watershed. In cases 4–6 for the TP load simulation, the values were 53.6, 169.5, and 476.5 kg·day$^{-1}$, respectively, and the RMSE value was the largest for case 6, which represented the BH watershed. In the evaluation using MAPE, the values were 11.5, 15.9, and 20.1% for cases 1–3, and 17.9, 16.3, and 21.3% for cases 4–6, respectively, and similar to the case for the RMSE, the value of MAPE was the highest for the BH watershed.

**Table 5.** Results of the model evaluation.

| Contents | Target | Predicted Study Area | r | NSE | RMSE (kg·day$^{-1}$) | MAPE (%) |
|---|---|---|---|---|---|---|
| Case 1 | BOD Load | JJ (Study Area 1) | 0.92 | 0.90 | 1031.1 | 11.5 |
| Case 2 | | HC (Study Area 1) | 0.94 | 0.92 | 1152.2 | 15.9 |
| Case 3 | | BH (Study Area 1) | 0.91 | 0.84 | 5376.5 | 20.1 |
| Case 4 | TP Load | JJ (Study Area 1) | 0.94 | 0.93 | 53.6 | 17.9 |
| Case 5 | | HC (Study Area 1) | 0.93 | 0.91 | 169.5 | 16.3 |
| Case 6 | | BH (Study Area 1) | 0.90 | 0.84 | 476.5 | 21.3 |

When determining the model using RMSE and MAPE, it is difficult to evaluate the models because there is no specific criterion, but the closer the values are to 0, the model performance can be evaluated to be better. In this regard, this study conducted a comparison with the results of previous studies using artificial neural networks and the CNN model performance of this study. Azadi et al. [55] used a model combining principal component analysis-M5P and ANN for simulation of the chemical oxygen demand (COD) load of landfill, and the Pearson correlation coefficient between the simulated COD load and the observed value was 0.98, and MAPE was 4%, and reported the excellence performance of the model. Also, Yu et al. [56] developed a kernel principal component analysis and extreme learning machine (KPCA-ELM) model for prediction of the COD and BOD concentrations of influent for wastewater treatment. As a result, MAPE of COD was 1.2% and MAPE of BOD was 1.3% in their research. Yan et al. [57] used back-propagation neural network, genetic algorithm combined BPNN, particle swarm optimization combined with BPNN and model combining PSO-BPNN and GA-BPNN for prediction of dissolved oxygen. The results of MAPE were 25.1%, 8.5%, 15.7%, and 6.7%, respectively, reporting that the performance of model combining PSO-BPNN and GA-BPNN was the highest for prediction of dissolved oxygen concentration. In addition, Shao et al. [58] reported a considerably high performance of their model with MAPE values ranging from 0.04% to 1.2% as a result of developing and applying a water quality prediction model based on the backpropagation neural network optimized by the Cuckoo Search algorithm for nine water quality items. As can be seen from these prior studies, excellent results have been demonstrated with the application of artificial neural network technology for simulation of contaminants. The results of cases 1, 2, 4, and 5 of this study also showed similar results compared to previous studies, but cases 3 and 6 showed relatively low performance of the model.

The reason for the low performance results for cases 5 and 6 is a result of the input data not used by the CNN model during the training process. The land cover status of the study areas in the JJ and HC water are forest-dominant with similar ratios of agricultural land area. As discussed in Section 3.1, the average discharge or BOD and TP concentrations in the JJ and HC watersheds showed similar levels. However, the BH watershed is an agriculture-dominant region, showing discharge higher levels of BOD and TP concentrations compared with the other study areas. In other words, cases 1 and 5 simulated the BOD and TP loads in the JJ watershed; during the training of the CNN model, data for the HC watershed was also used, providing indirect experience for the JJ watershed, which resulted in a good prediction performance for the JJ watershed. In addition, in cases 2 and 5, during the training of the CNN model, data for the JJ watershed was also used, providing indirect experience for the HC watershed, which resulted in a good prediction performance for the BOD and TP loads in the HC watershed. In contrast, in cases 5 and 6, during the training of the CNN model, there was no training data available similar to the BH watershed, resulting in lower model performance compared with cases 1, 2, 4, and 5, and it is thought that if CNN models are trained.

Consequentially, this study believes the CNN model using hydrological image could enable to be a role as a pollutant loads prediction model through the result of this study and the comparison between this study and previous study. In particular, the CNN model in this study presented the improvement of being able to simulate pollutant loads by reflecting two-dimensional spatial information, unlike the preceding study using a normal ANN model. In addition, this study showed the merit that could enable convenient and simple modeling with limited input data. This paper thought this CNN model might be helpful to the government's policy decision-making and insufficient administrative power for water management. Furthermore, the methodology proposed of this study could be applied to the remote sensing field, so that this study judged it could be used to predict the pollution load of rivers using satellite images.

## 4. Conclusions

In this study, hydrological images and CNN models were used to simulate the pollution load of ungauged watersheds. The model performance of cases 1, 2, 4, and 5 was good, but cases 3 and 6 showed relatively low performance. This was a result of a lack of predictive power for the data not used during CNN model training, which is considered a limitation of the CNN model when using hydrological images. Although this study cannot clearly demonstrate whether spatial feature extraction, the core function of the CNN model, was properly applied to the hydrological images to simulate the pollution load, hydrological images with features similar to the simulation target were used to train the CNN model, thereby obtaining better model performance. In addition, the fact that continuous simulation is not possible using hydrological images and the CNN model is a limitation of this study. Through improvements to these methods, this study aims to perform further research that enables the continuous simulation of the pollution load even at non-precipitation conditions.

Nevertheless, for the results of this study based on the simulation of the pollution loads in ungauged watersheds affected by precipitation and land cover, the CNN model showed a good level of performance and demonstrated simplicity, i.e., not requiring various data or additional calibration and adjustment, as well as sensitivity analyses required by conceptual models. Furthermore, a CNN model with high accuracy should be obtained if the model can be trained using the methodology proposed in this study and an increased number and variety of hydrological images.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Conflicts of Interest:** The author declares no conflict of interest.

## References

1. Young, R.A.; Onstad, C.A.; Bosch, D.D.; Anderson, W.P. AGNPS: A nonpoint source pollution model for evaluating agricultural watersheds. *J. Soil Water Conserv.* **1989**, *44*, 168–173.
2. Arnold, J.G.; Allen, P.M.; Bernhardt, G. A comprehensive surface-groundwater flow model. *J. Hydrol.* **1993**, *142*, 47–69. [CrossRef]
3. Lewis, A.R. *Storm Water Management Model User's Manual*; Water Supply and Water Resources Division National Risk Management Research Laboratory: Cincinnati, OH, USA, 2004.
4. Aisha, M.S. Evaluation of SWAT Model Applicability for Water Impairment Identification and TMDL Analysis. Ph.D. Thesis, University of Maryland, College Park, MD, USA, 30 October 2007.
5. Patel, A.B.; Joshi, G.S. Modeling of rainfall-runoff correlations using artificial neural network-a case study of dharoi watershed of a sabarmati river basin, India. *Civ. Eng. J.* **2017**, *3*, 78–87. [CrossRef]
6. Song, C.M.; Kim, J.S. Applicability evaluation of the hydrological image and convolution neural network for prediction of the biochemical oxygen demand and total phosphorus loads in agricultural areas. *Agriculture* **2020**, *10*, 529. [CrossRef]
7. Mercier, G.; Blais, J.F.; Chartier, M. Décontamination à l'échelle pilote de sols pollués en métaux toxiques par des procédés miniers et lixiviation chimique. *J. Environ. Eng. Sci.* **2007**, *6*, 53–64. [CrossRef]

8.   Salas, F.R.; Somos-Valenzuela, M.A.; Dugger, A.; Maidment, D.R.; Gochis, D.J.; David, C.H.; Yu, W.; Ding, D.; Clark, E.P.; Noman, N. Towards real-time continental scale streamflow simulation in continuous and discrete space. *JAWRA J. Am. Water Resour. Assoc.* **2018**, *54*, 7–27. [CrossRef]

9.   Akhtar, M.K.; Corzo, G.A.; van Andel, S.J.; Jonoski, A. River flow forecasting with artificial neural networks using satellite observed precipitation pre-processed with flow length and travel time information: Case study of the Ganges river basin. *Hydrol. Earth Syst. Sci.* **2009**, *13*, 1607–1618. [CrossRef]

10.  Cakir, S.; Sita, M. Evaluating the performance of ANN in predicting the concentrations of ambient air pollutants in Nicosia. *Atmos. Pollut. Res.* **2020**, *11*, 2327–2334. [CrossRef]

11.  Kiiza, C.; Pan, S.; Bockelmann-Evans, B.; Babatunde, A. Predicting pollutant removal in constructed wetlands using artificial neural networks (ANNs). *Water Sci. Eng.* **2020**, *13*, 14–23. [CrossRef]

12.  Hamed, M.M.; Khalafallah, M.G.; Hassanien, E.A. Prediction of wastewater treatment plant performance using artificial neural networks. *Environ. Model. Softw.* **2004**, *19*, 919–928. [CrossRef]

13.  Mishra, P.K.; Karmakar, S. Performance of optimum neural network in rainfall–runoff modeling over a river basin. *Int. J. Environ. Sci. Technol.* **2018**, *16*, 1289–1302. [CrossRef]

14.  Pradhan, P.; Tingsanchali, T.; Shrestha, S. Evaluation of soil and water assessment tool and artificial neural network models for hydrologic simulation in different climatic regions of Asia. *Sci. Total Environ.* **2020**, *701*, 13–144. [CrossRef] [PubMed]

15.  Rajaee, T.; Khani, S.; Ravansalar, M. Artificial intelligence-based single and hybrid models for prediction of water quality in rivers: A review. *Chemom. Intell. Lab. Syst.* **2020**, *200*, 103978. [CrossRef]

16.  Zhu, S.; Heddam, S. Prediction of dissolved oxygen in urban rivers at the Three Gorges Reservoir, China: Extreme learning machines (ELM) versus artificial neural network (ANN). *Water Qual. Res. J.* **2019**, *55*, 106–118. [CrossRef]

17.  Wilby, R.L.; Abrahart, R.; Dawson, C. Detection of conceptual model rainfall—runoff processes inside an artificial neural network. *Hydrol. Sci. J.* **2003**, *48*, 163–181. [CrossRef]

18.  Jain, A.; Sudheer, K.P.; Srinivasulu, S. Identification of physical processes inherent in artificial neural network rainfall runoff models. *Hydrol. Process.* **2004**, *18*, 571–581. [CrossRef]

19.  Sudheer, K.P.; Jain, A. Explaining the internal behaviour of artificial neural network river flow models. *Hydrol. Process.* **2004**, *18*, 833–844. [CrossRef]

20.  Biswas, A.K. Integrated water resources management: A reassessment. *Water Int.* **2004**, *29*, 248–256. [CrossRef]

21.  EGIS: Environmental Geographic Information Service. Available online: https://www.egis.me.go.kr (accessed on 9 January 2019).

22.  WAMIS: Water Management Information System, National Institute of Environmental Research. Available online: https://www.water.nier.go.kr (accessed on 1 March 2019).

23.  KMA: Korea Meteorological Administration. Available online: https://www.kma.go.kr (accessed on 3 January 2020).

24.  NIER: National Institute of Environmental Research. Available online: https://www.nier.go.kr (accessed on 11 November 2019).

25.  Song, C.M. Hydrological image building using curve number and prediction and evaluation of runoff through convolution neural network. *Water* **2020**, *12*, 2292. [CrossRef]

26.  United States Department of Agriculture; Natural Resources Conservation Service (NRCS); Conservation Engineering Division. *Urban Hydrology for Small Watersheds*; Natural Resources Conservation Service: Washington, DC, USA, 1986. Available online: https://www.nrcs.usda.gov/Internet/FSE_DOCUMENTS/stelprdb1044171.pdf (accessed on 2 November 2019).

27.  Li, C.; Liu, M.; Hu, Y.; Shi, T.; Zong, M.; Walter, M.T. Assessing the impact of urbanization on direct runoff using improved composite cn method in a large urban area. *Int. J. Environ. Res. Public Health* **2018**, *15*, 775. [CrossRef]

28.  Wang, H.; Chen, Y. Identifying key hydrological processes in highly urbanized watersheds for flood forecasting with a distributed hydrological model. *Water* **2019**, *11*, 1641. [CrossRef]

29.  Ministry of Land, Infrastructure and Transport. *Design Flood Estimation Techniques*; Ministry of Land Transport and Maritime Affairs: Seoul, Korea, 2012. (In Korean)

30.  Schumann, A.H. Thiessen polygon. In *Encyclopedia of Hydrology and Lakes. Encyclopedia of Earth Science*; Springer: Dordrecht, The Netherlands, 1998; pp. 648–649. [CrossRef]

31.  Miller, H. Tobler's first law and spatial analysis. *Ann. Assoc. Am. Gerogr.* **2004**, *94*, 284–289. [CrossRef]

32.  Python. Available online: https://www.python.org/ (accessed on 4 January 2020).

33.  LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *IEEE* **1998**, *86*, 2278–2324. [CrossRef]

34.  Taravat, A.; Del Frate, F.; Cornaro, C.; Vergari, S. Neural networks and support vector machine algorithms for automatic cloud classification of whole-sky ground-based images. *IEEE Trans. Geosci. Remote Sens. Lett.* **2014**, *12*, 666–670. [CrossRef]

35.  Medina, E.; Petraglia, M.R.; Gomes, J.G.R.C.; Petraglia, A. Comparison of CNN and MLP classifiers for algae detection in underwater pipelines. In Proceedings of the 2017 Seventh International Conference on Image Processing Theory, Tools and Applications (IPTA), Montreal, QC, Canada, 28 November–1 December 2017; pp. 1–6.

36.  Zeiler, M.D.; Fergus, R. Visualizing and understanding convolutional networks. In *Computational Data and Social Networks*; Springer Science and Business Media LLC: Berlin, Germany, 2014; Volume 8689, pp. 818–833.

37.  Ide, H.; Kurita, T. Improvement of learning for CNN with ReLU activation by sparse regularization. In Proceedings of the 2017 International Joint Conference on Neural Networks (IJCNN), Anchorage, AK, USA, 14–19 May 2017; pp. 2684–2691.

38. Chen, Z.; Ho, P.-H. Global-connected network with generalized ReLU activation. *Pattern Recognit.* **2019**, *96*, 106961. [CrossRef]
39. Keras. The Python Deep Learning API. Available online: https://keras.io/ (accessed on 27 August 2020).
40. TensorFlow. An End-to-End Open Source Machine Learning Platform. Available online: https://www.tensorflow.org/ (accessed on 27 August 2020).
41. Bottou, L. Large-scale machine learning with stochastic gradient descent. In Proceedings of the 19th International Conference on Computational Statistics, Paris, France, 22–27 August 2010; pp. 177–186. [CrossRef]
42. Qian, N. On the momentum term in gradient descent learning algorithms. *Neural Netw.* **1999**, *12*, 145–151. [CrossRef]
43. Nesterov, Y. A method for unconstrained convex minimization problem with the rate of convergence. *Doklady USSR* **1983**, *269*, 543–547. Available online: https://www.semanticscholar.org/paper/A-method-for-unconstrained-convex-minimization-with-Nesterov/ed910d96802212c9e45d956adaa27d915f5d7469 (accessed on 18 April 2020).
44. Duchi, J.; Hazan, E.; Singer, Y. Adaptive subgradient methods for online learning and stochastic optimization. *JMLR* **2011**, *12*, 2121–2159.
45. Dozat, T. Incorporating nesterov momentum into Adam. In Proceedings of the ICLR Workshop, San Juan, Puerto Rico, 2–4 May 2016; pp. 2013–2016. Available online: https://openreview.net/pdf/OM0jvwB8jIp57ZJjtNEZ.pdf (accessed on 21 April 2020).
46. Zeiler, M.D. ADADELTA: An Adaptive Learning Rate Method. *arXiv* **2012**, arXiv:1212.5701v1.
47. Hinton, G.; Tieleman, T. RMSprop Gradient Optimization. Lecture 6e of his Coursera Class. 2014. Available online: https://www.cs.toronto.edu/~{}\{\}tijmen/csc321/slides/lecture_slides_lec6.pdf (accessed on 10 February 2019).
48. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. In Proceedings of the International Conference Learnning Representations (ICLR), San Diego, CA, USA, 5–8 May 2015.
49. Pillow. Available online: https://www.python-pillow.org (accessed on 10 February 2020).
50. Lee, K.; Choi, C.; Shin, D.H.; Kim, H.S. Prediction of heavy rain damage using deep learning. *Water* **2020**, *12*, 1942. [CrossRef]
51. Alsumaiei, A.A. Utility of artificial neural networks in modeling pan evaporation in hyper-arid climates. *Water* **2020**, *12*, 1508. [CrossRef]
52. Mulualem, G.M.; Liou, Y.-A. Application of artificial neural networks in forecasting a standardized precipitation evapotranspiration index for the upper blue nile basin. *Water* **2020**, *12*, 643. [CrossRef]
53. Dancey, C.; Reidy, J. *Statistics without Maths for Psychology*, 5th ed.; Prentice Hall: Upper Saddle River, NJ, USA, 2011.
54. Lipiwattanakarn, S.; Saengsawang, S. Performance comparison of a conceptual hydrological model and a back-propagation neural network model in rainfall-runoff modeling. *Eng. J. Res. Dev.* **2005**, *16*, 35–42.
55. Azadi, S.; Amiri, H.; Rakhshandehroo, G.R. Evaluating the ability of artificial neural network and PCA–M5P models in predicting leachate COD load in landfills. *Waste Manag.* **2016**, *55*, 220–230. [CrossRef] [PubMed]
56. Yu, T.; Yang, S.; Bai, Y.; Gao, X.; Li, C. Inlet water quality forecasting of wastewater treatment based on kernel principal component analysis and an extreme learning machine. *Water* **2018**, *10*, 873. [CrossRef]
57. Yan, J.; Xu, Z.; Yu, Y.; Xu, H.; Gao, K. Application of a hybrid optimized bp network model to estimate water quality parameters of beihai lake in beijing. *Appl. Sci.* **2019**, *9*, 1863. [CrossRef]
58. Shao, D.; Nong, X.; Tan, X.; Chen, S.; Xu, B.; Hu, N. Daily water quality forecast of the south-to-north water diversion project of china based on the cuckoo search-back propagation neural network. *Water* **2018**, *10*, 1471. [CrossRef]