Received January 14, 2020, accepted January 19, 2020, date of publication February 3, 2020, date of current version March 4, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2971264

Flood Prediction Using Rainfall-Flow Pattern in Data-Sparse Watersheds

YUELONG ZHU, JUN FENG, LE YAN, TAO GUO, AND XIAODONG LI
Computer and Information College, Hohai University, Nanjing 211100, China

Corresponding author: Jun Feng (fengjun@hhu.edu.cn)

This work was supported in part by the National Key Research and Development Program of China under Grant 2018YFC0407901, and in part by the National Key Research and Development Program of China under Grant 2018YFC1508100.

ABSTRACT Real-time flood forecasting of small- and medium-sized rivers in areas with scarce hydrological data is an urgent problem that needs to be solved. Traditional hydrological model parameters cannot be fully trained owing to a lack of data; thus, results obtained by such models are not satisfactory. We need a new way to solve the forecasting problem for small- and medium-sized rivers. We found that the time series of some feature variables have evident change trajectories in spatial dimension, and the change of some feature variables in the spatial dimension has a decisive influence on flooding processes, such as the spatial distribution of rainfall. To reflect the change of feature variables in spatial dimension with to solve the problem of the lack of hydrological data, we constructed a rainfall-flow pattern composed of a spatial-temporal dynamic time warping algorithm and multi-feature algorithm to measure the similarity of hydrological time series. In the experimental watersheds, we used rainfall-flow patterns to forecast the short-term flood streamflow, and satisfactory results were obtained. This suggests that the algorithm is suitable for hydrological studies and improves the accuracy of real-time flood forecasting for longer forecast periods.

INDEX TERMS Spatiotemporal sequence data, rainfall-flow pattern matching, similarity measurement algorithm, multi-feature algorithm, ST-DTW algorithm.

I. INTRODUCTION

In China, there are nearly 9,000 small- and medium-sized rivers, covering an area of 200 to 3000 km^2. In recent years, extreme weather conditions and frequent rapid floods caused by local heavy rainfall have become the main cause of casualties [1]. According to recorded statistics, flood damage to small rivers accounts for approximately 70% to 80% of the total loss of floods [2]. Therefore, accurate and timely flood forecasting can effectively reduce disaster losses and improving real-time forecast accuracy is a very important technical measure in flood control and disaster mitigation [3].

The hydrological conceptual model and data-driven model are two main methods in flood forecasting research [4]. Over the years, researchers in the filed have achieved relatively satisfactory results in the study of flood forecasting for large rivers in China [5], although research of flood forecasting for small and medium rivers is just in its infancy. Flooding of small and medium-sized rivers are short in duration, difficult to predict, difficult to prevent, and lack measured hydrological data. Therefore, it is difficult to meet the needs of existing hydrological model parameters, which makes it difficult to study flood forecasting for such rivers [6].

The data-driven model generally does not consider the physical mechanism of the hydrological process. It is a black box method that aims to establish an optimal mathematical relationship between input and output data [7]. The most commonly used data-driven model is the regression model. Owing to the introduction of neural network models, nonlinear time series analysis models, fuzzy mathematical methods, grey system models, and the development of hydrological data acquisition capabilities and computational capabilities, progress with data-driven models has been made by predicting and simulating nonlinear hydrological applications and the noise complexity of captured data. This has attracted the attention of hydrologists [8].

In recent years, the rapid development of artificial intelligence (AI) and big data has triggered revolutionary changes in many research fields [9]–[12]. In the field of hydrology, AI algorithms are used to extract predictive features embedded

The associate editor coordinating the review of this manuscript and approving it for publication was Ying Li.

in historical hydrometeorological data. AI is a popular tool for capturing linear and nonlinear relationships between flow, rainfall, climate indices, and related inputs [13], [14]. Popular AI algorithms include artificial neural networks (ANNs), support vector machines (SVMs), fuzzy logic, and evolutionary computations [15]. However, these AI models have their respective limitations. For example, the limitations of the ANN model include iterative adjustments to the parameters, as well as overfitting or overtraining, which can lead to large errors in the prediction of test samples [16].

Because of the optimization and popularization of deep learning algorithms, the data-driven model is widely used in stream-flow forecasting of large rivers. Bai *et al.* [15] proposed a daily reservoir inflow prediction model based on multi-scale depth feature learning. Liu *et al.* [17] proposed a method of integrating a stacked automatic encoder (SAE) and a back propagation neural network (BPNN); they developed a deep learning method for predictive streams that takes advantage of the SAE's powerful feature representation capabilities and the BPNN's superior predictive power. Other commonly used deep learning models include long-term short-term memory (LSTM) networks (a special type of recurrent neural network). LSTM networks exhibit superior performance in large watersheds. Kratzert *et al.* [18] highlighted the potential of LSTM in hydrological modeling applications for 241 freely available CAMELS5 watershed datasets. Some hydrologists expect AI to raise our hydrological forecasting capabilities to unprecedented levels, as it has in many other fields [7]. Indeed, AI has been well applied to large watershed-rich watersheds with big data. However, hydrological modeling does not currently perform comparably well in small watersheds. Nevertheless, the hydrological community still regards the AI model as a valuable supplement to the hydrological conceptual model in hydrological modeling of small and medium watersheds. In summary, small-watershed forecasting based on the AI model is limited in the following two parts:

1) The flow of small and medium-sized river watershed exports may be affected by upstream or downstream factors. Previous studies often ignored the spatial characteristics of the model input data and only used the averaged data within the watershed. For example, if the rainfall center is close to the watershed exit, the flow at the watershed exit will peak in a short period of time.
2) Meteorological and hydrological data in small and medium watersheds are often scarce, and the number of training samples rarely reaches the optimal value for an AI model.

Flood forecasting in small and medium-sized watersheds is influenced by many factors, such as soil water content and rainfall center location. The initial soil water content and the movement trajectory of rainfall play an important role; however, these factors are not easy to measure directly, resulting in a small amount of data obtained. To address these limitations, this study proposes a short-term flood forecasting model using feature factor decomposition in conjunction with weather-time rainfall-flow pattern matching. The input to the model includes soil water and rainfall from the watershed. The output of the model is the flood flow at the watershed exit for the next 10 h. The main contributions of the study are summarized below.

1) We propose a decomposition method of input features to mine the implicit rainfall-flow model. The multi-feature algorithm extends one-dimensional features to multi-dimensional features for rainfall-flow pattern matching. The proposed spatial-temporal dynamic time warping (ST-DTW) algorithm combines the measurement of temporal and spatial distances.
2) We propose a rainfall-flow pattern matching method that overcomes the paucity of hydrological data in small and medium watersheds. In contrast to traditional machine learning models, pattern matching does not require large amounts of training data to achieve optimal results.
3) We propose the use of hydrological data from wet and dry watersheds to construct rainfall-flow patterns. In addition, we verify the superiority of the proposed model from two aspects: model simulation performance and real-time prediction accuracy.

The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 introduces the preliminaries. Section 4 presents details of the rainfall-flow patterns. Model data and experiments are introduced in Section 5. Finally, Section 6 offers some conclusions and suggestions for future work.

II. RELATED WORK

The rainfall flow model is divided into a physics-based hydrological conceptual model and a data-driven model. The hydrological conceptual model proposes hypotheses, generalizations, and mathematical simulations of hydrological phenomena on a physical basis. The data-driven model only obtains information from the data, regardless of the characteristics and processes of the hydrological system.

A. CONCEPTUAL MODEL BASED ON HYDROLOGICAL PROCESS

The hydrological conceptual model describes the hydrological processes of its components, such as the process of water circulation in nature. The model simulates the flow process of the watershed by simulating the flow and the river evolution processes of the flow in the watershed. As early as the 1960s, the distribution of climatic conditions and uneven spatial variation to watersheds has been discovered by hydrologists, and its impact on the rainfall-flow relationship of watersheds has been studied. Several well-known physical models were developed from the 1970s to the mid-1980s, such as the American Stanford model [19], the Sacramento model [20], the Japanese tank model [21], and the Chinese Xinanjiang model [22]. In addition, the European SHE model [23] is the first highly representative distributed hydrological model, which was developed in 1986.

The structure of hydrological conceptual models is limited insofar as it is not yet possible to describe each of these subprocesses rigorously using mathematical equations derived from consideration of the physical attributes of watershed. In addition, they are limited in actual use because their optimization method determines the dependence of the model parameters on the measured rainfall-flow data.

B. DATA DRIVEN MODEL BASED ON AI ALGORITHM

The data-driven stream-flow prediction model captures linear and nonlinear relationships between stream-flow, rainfall, climate indices, and related inputs [13], [24]. Conventional black box time series models such as least squares (LS), autoregressive (AR), autoregressive moving average (ARMA), multiple linear regression (MLR), and stepwise cluster analysis (SCA) have been applied to hydrological forecasting [25]. However, these models cannot handle nonlinear hydrological processes.

In the past two decades, hydrologists have developed AI models to address the abovementioned limitations and simulate nonlinear hydrological processes. In recent years, the development of artificial neural networks has been relatively fast. Hsu *et al.* [26] proposed the use of an ANN for typical rainfall-flow forecasting problems. The method is designed to estimate the parameters and systems of ANN networks quickly and accurately, and to estimate uncertainty as well. However, the limitation of the ANN model is that the appropriate architecture must be designed through training and testing, and a small architecture may not have enough data for the ANN model to learn.

Guo *et al.* [27] used an SVM to predict monthly stream-flow, demonstrating that SVM offer a promising hydrological prediction method. Compared with physical models, an SVM requires less data and performs well in real-time predictions. Compared with the ANN model, an SVM has better generalization capabilities and a higher prediction accuracy.

Fuzzy logic algorithms solve the uncertainty of a model by determining the relevant input variables [28]. The most widely used fuzzy logic algorithm is the adaptive-network-based fuzzy inference system (ANFIS). El-Shafie *et al.* [29] used the ANFIS to construct a monthly stream-flow prediction model. The experimental results showed that the fuzzy system can deal with the inaccuracy and ambiguity of hydrological data, and that the ANFIS model is better at dealing with the prediction of high water-level scenarios in large rivers.

Savic *et al.* [30] first used the gene programming (GP) method in evolutionary computation (EC) to model rainfall flow, with the ability to reduce the large number of parameters required to identify conceptual model calibrations. The GP method clearly gives the form of a determined function, and has a greater ability to capture rainfall-flow relationships compared with the ANN.

Traditional data-driven models do not consider the physical process of the rainfall-flow pattern, which leads to an attempt to utilize all available hydrological data. This considerably increases the feature dimensions and requires a large number of training parameters [31]. In addition, ignoring the spatial feature of the input data leads to inaccuracies and uncertainties in complex data-driven model predictions [32].

III. PREPARED KNOWLEDGE

1) PRINCIPAL COMPONENT ANALYSIS

Principal component analysis (PCA) is probably the most popular multivariate statistical technique used in most disciplines, and it is often used for data preprocessing and dimensionality reduction [33]. The main function of PCA is that it can represent more information with fewer variables, which are several interrelated variables. The goal of PCA is to extract important information from variables and then reduce the data dimension by preserving important information, simplifying the complexity of the problem, and reducing consumption. The PCA is described as follows:

$$Y = \sum_{i=1}^{n} UX, \quad 1 \leq i \leq n, \tag{1}$$

where $U = (u_{i1}, u_{i2}, \cdots, u_{in})$ denotes weight, $X = (x_1, x_2, \cdots, x_n)$ denotes the original variable, and $Y = (y_1, y_2, \ldots, y_n)$ is the new variable changed by X.

2) DYNAMIC TIME WARPING

The dynamic time warping (DTW) algorithm was first proposed by Fumitada Itakura. It can be used to solve the similarity measure of non-equal time series. However, its application is more extensive, especially in the field of speech recognition, where it is used for the identification of isolated words, gesture recognition, data mining, and information retrieval.

In actual experiments, a situation that is often encountered is that of two time series of differing lengths having a high similarity in their overall shape (similar to speech similarity); additionally, the situation occurs where one of the time series needs to be distorted using dynamic programming such that the two time series are as similar as possible. The main idea of DTW is to use dynamic programming to extend or shorten two similar time series to obtain the shortest distance between them. This shortest distance is called the shortest warping path, which is the DTW distance.

Consider time series Q and test time series C, with lengths n and m, respectively, that is,

$$\begin{aligned} Q &= q_1, q_2, \ldots, q_i, \ldots, q_n \\ C &= c_1, c_2, \ldots, c_i, \ldots, c_n \end{aligned} \tag{2}$$

It is understood that the speech sequence Q has n frames, and the feature value of the *ith* frame is q_i. Similarly, the speech sequence C has m frames, and the feature value of the *ith* frame is c_i.

As shown in Fig. 1, a matrix of size $n * m$ can be skillfully constructed with the (i, j) element corresponding to the distance $d(q_i, c_j)$ of the two points q_i and c_j, which represents the similarity between each point of time series Q and each point of time series C. There is a plurality of distance formulas for

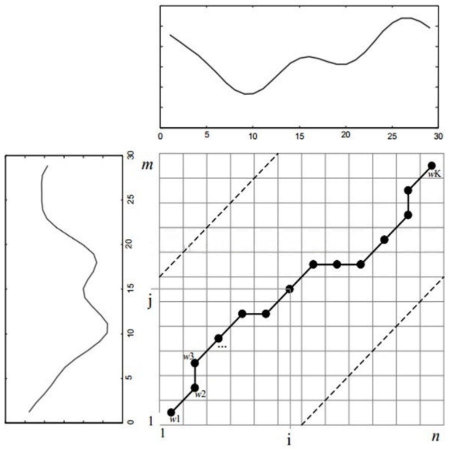


FIGURE 1. Illustration of warped path.

the calculation of the similarity. The Euclidean distance is generally adopted, namely, $d(q_i, c_j) = (q_i - c_j)^2$. Each matrix element (i, j) indicates the alignment of the point q_i with point c_j. The algorithm aims to find a path through several grid points of this grid. The point through which the path passes is the point at which the two time series are aligned. This path is defined as a warping path and is represented by W.

The k^{th} element of the regular path W is defined as $W_k = (i, j)_k$, and W defines the mapping of the time series Q and C, such that the warping paths of Eq. 3 can be obtained.

$$\begin{aligned} W &= w_1, w_2, \ldots, w_k, \ldots, w_k \\ max(m, n) &\leq K < m + n - 1 \end{aligned} \tag{3}$$

The warping path W must satisfy the following constraints: in $(a - a') \leq 1$ and $w_k = (m, n)$, the selection of the warping path points must satisfy the order of the sequences. The warping path must start from the lower left corner and complete the entire path to the upper right corner.

If $w_{k-1} = (a', b')$, the next point $w_k = (a, b)$ for the path must satisfy $(a - a') \leq 1$ and $(b - b') \leq 1$. When the point cannot be matched, it can only be aligned with its neighboring points. This ensures that every coordinate in Q and C appears in W.

If $w_{k-1} = (a', b')$, then the next point $w_k = (a, b)$ for the path must satisfy $0 \leq (a - a')$ and $0 \leq (b - b')$. The warping path W must be monotonic; this is necessary to ensure that the broken lines in Fig. 1 do not intersect. Once the above constraints are satisfied, the warping path can only select three directions— up, right, or diagonal.

For the above constraints, there are many rules for the constraint path that satisfies the constraint, and the shortest cumulative distance path is needed, which is defined as

$$DTW(Q, C) = min\{\frac{\sum_{k=1}^{K}}{w_k}\} \tag{4}$$

where the denominator K is used to compensate for the warping path W of different lengths. The dynamic programming approach requires the determination of a warping path with the smallest distance. Matching the two sequences Q and C from the point $(0, 0)$, each time a point is reached, the distance calculated by all previous points is accumulated. After reaching the end point (n, m), the distance is the final total distance, indicating the degree of similarity between the time series Q and C.

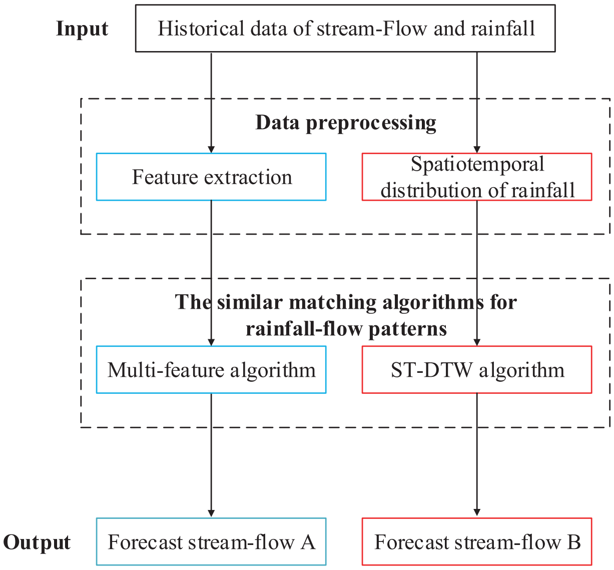

Input
Historical data of stream-Flow and rainfall
Data preprocessing
Feature extraction
Spatiotemporal distribution of rainfall
The similar matching algorithms for rainfall-flow patterns
Multi-feature algorithm
ST-DTW algorithm
Output
Forecast stream-flow A
Forecast stream-flow B

FIGURE 2. Rainfall-flow pattern prediction process.

As shown in Eq. 5, the cumulative distance $\gamma(i, j)$ is the current grid point distance $d(i, j)$, which is the grid point q_i. The sum of the Euclidean distances of the c_js is the cumulative distance of the smallest neighboring element that can reach that point.

$$\gamma(i, j) = d(q_i, c_j) + \min \left\{ \begin{array}{l} \gamma(i-1, j-1) \\ \gamma(i-1, j) \\ \gamma(i, j-1) \end{array} \right\} \tag{5}$$

IV. CONSTRUCTION OF RAINFALL-FLOW PATTERNS

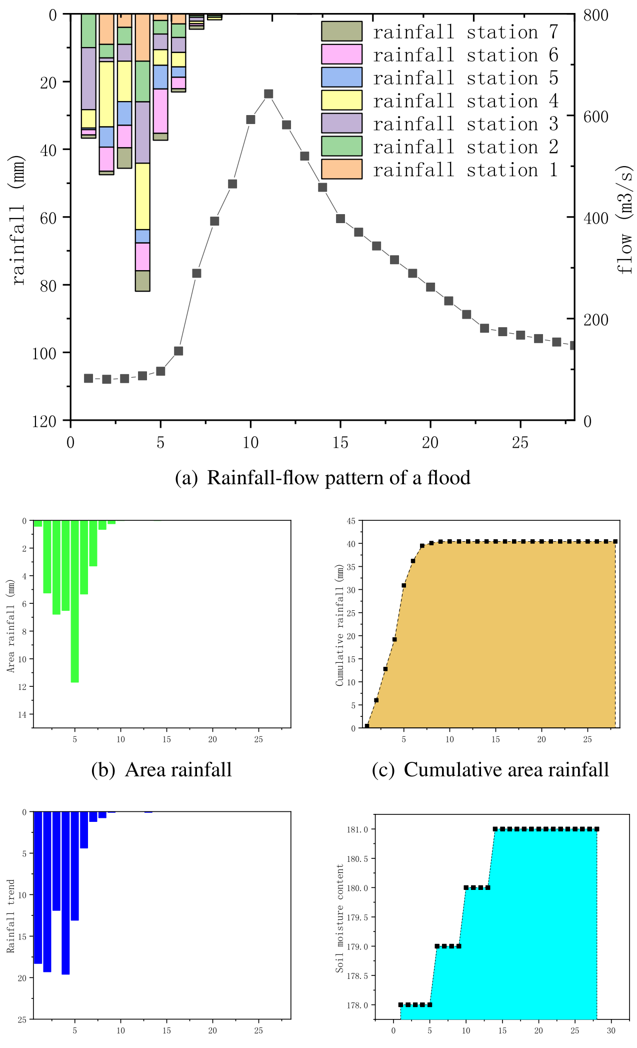
The specific process of the rainfall-flow pattern structure is shown in Fig. 2. The first step is the input of original flood data,which includes historical flow and rainfall data. The second step, data preprocessing, is mainly feature extraction and spatiotemporal distribution of rainfall. For the third step, we introduce two similar matching algorithms for rainfall-flow patterns. Finally, for the fourth step, we output a forecast of the stream-flow.

A. DATA PREPROCESSING

The two pattern matching algorithms use their own data preprocessing methods.

1) DATA PREPROCESSING OF THE MULTI-FEATURE ALGORITHM

The multi-feature algorithm is a hydrological sequence similarity calculation method, which mainly uses the method of dimensionality reduction to measure the similarity of



(a) Rainfall-flow pattern of a flood

(b) Area rainfall

(c) Cumulative area rainfall

(d) Rainfall trend

(e) Soil moisture content

FIGURE 3. Feature extraction of Rainfall-flow pattern. The dotted line denotes the flow of the flood. The column chart denotes the measurement of each rainfall station in Figure (a). Figures (b)–(d) denote four features after the extraction of the rainfall-flow pattern.

hydrological processes. The idea of the hydrological multi-feature algorithm is to extract the most important feature variables in the flooding process and to transform the similarity of the flooding process into the similarity of the time series of the feature variables. As shown in Fig. 3, PCA and expert experience are used to analyze and screen multivariate hydrological feature variables. We observe that four feature variables have the greatest impact on small- and medium-sized rivers: area rainfall, rainfall trend, cumulative area rainfall, and soil moisture content. Accordingly, in this section, we introduce four important feature variables in detail.

Area rainfall is a physical quantity that describes the average precipitation per unit area over an entire watershed. It can better reflect the precipitation over the entire area objectively. The area rainfall is calculated as Eq. 6:

$$R\left(t_p\right) = \sum_{i=1}^{n} r_i\left(t_p\right) HT_i A_i \tag{6}$$

where $R\left(t_p\right)$ is the area rainfall at the t_p during the flood process; $r_i\left(t_p\right)$ represents the rainfall measured by the *ith* rain station at the t_p; A_i is the sub-watershed area of the *ith* rainfall site as a percentage of the total area of the study watershed; n is the number of rainfall stations in the entire watershed; and HT_i is the influence of the confluence time.

Cumulative area rainfall reflects the amount of precipitation in the area for a period of time. The cumulative area rainfall is calculated as Eq. 7:

$$Sum_p = \frac{1}{S}\sum_{i=1}^{n}\sum_{t=0}^{t_0}(S_i * P(i)_t) \tag{7}$$

where Sum_p is the cumulative area rainfall; S denotes the area of the watershed; S_i represents the area of the sub-watershed; i represents the number of rainfall stations; t_0 represents the forecast time point; and $P(i)_t$ represents the rainfall at station i at time point t.

The standard deviation of the rainfall is used to compare the magnitude of rainfall trend intensity over time, and is calculated as Eq. 8:

$$RI_t = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(P(i)_t - \mu_t)^2} \tag{8}$$

where RI_t is the rainfall trend intensity at time point t; i represents the number of rainfall stations; $P(i)_t$ represents the rainfall at station i at time point t; and μ_t is the average rainfall of i rainfall stations at time t.

The soil water content has a very fine division in the hydrology discipline and is divided into the upper layer tension water, the lower tension water, and the deep tension water. The soil moisture content in this paper is expressed as Eq. 9:

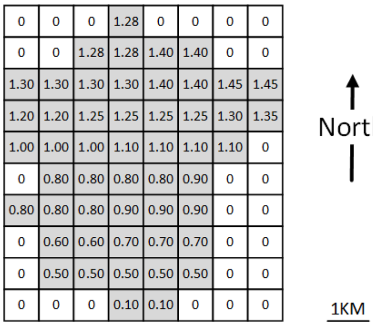
$$WM = WUM + WLM + WDM \tag{9}$$

where WM represents the soil moisture; WUM represents the upper layer tension water; WLM represents the lower layer tension water; and WDM represents the deep tension water.

2) DATA PREPROCESSING OF THE ST-DTW ALGORITHM

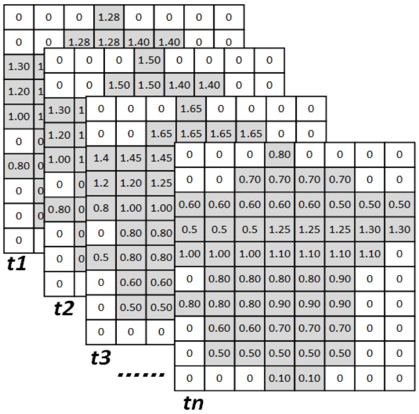
The measure of the similarity of the ST-DTW algorithm is based on the matrix data. We therefore first need to rasterize the flood-related data and generate a rasterized sequence of rainfall of distribution matrices. Rainfall data is an important determinant of the evolution of the flooding process. The accumulation of rainfall indirectly reflects the degree of soil moisture content, the amount of area rainfall, and the location of the center of the rainstorm. In this study, we mainly rasterize the rainfall data and finally generate the sequence of rainfall distribution matrices.

As shown in Fig. 4, the shape of a river watershed after rasterization is shown as a gray grid. A grid edge represents the actual distance of $1km$. After rasterization, the river watershed is a matrix of 10 rows by 8 columns. Based on the rainfall data obtained from the rainfall stations in the river



0	0	0	1.28	0	0	0	0
0	0	1.28	1.28	1.40	1.40	0	0
1.30	1.30	1.30	1.30	1.40	1.40	1.45	1.45
1.20	1.20	1.25	1.25	1.25	1.25	1.30	1.35
1.00	1.00	1.00	1.10	1.10	1.10	1.10	0
0	0.80	0.80	0.80	0.80	0.90	0	0
0.80	0.80	0.80	0.90	0.90	0.90	0	0
0	0.60	0.60	0.70	0.70	0.70	0	0
0	0.50	0.50	0.50	0.50	0.50	0	0
0	0	0	0.10	0.10	0	0	0

FIGURE 4. Rainfall spatial distribution matrix.



0	0	0	0.80	0	0	0	0
0	0	0.70	0.70	0.70	0.70	0	0
0.60	0.60	0.60	0.60	0.60	0.50	0.50	0.50
0.5	0.5	0.5	1.25	1.25	1.25	1.30	1.30
1.00	1.00	1.00	1.10	1.10	1.10	1.10	0
0	0.80	0.80	0.80	0.80	0.90	0	0
0.80	0.80	0.80	0.90	0.90	0.90	0	0
0	0.60	0.60	0.70	0.70	0.70	0	0
0	0.50	0.50	0.50	0.50	0.50	0	0
0	0	0	0.10	0.10	0	0	0

FIGURE 5. Rainfall spatiotemporal distribution matrices sequence.

watershed, the rainfall data of the river watershed are filled in with the rasterization matrix of the river watershed; areas outside the watershed are filled with zeros. In this manner, the rainfall rasterized matrix in unit time is generated, and we call this the rainfall distribution matrix. Because a flood lasts for several hours, it corresponds to multivariate rainfall distribution matrices. Multivariate rainfall distribution matrices form a sequence of rainfall distribution matrices, which is shown in Fig. 5.

B. ALGORITHM FOR THE RAIN-FLOW PATTERN MATCHING

1) MULTI-FEATURE ALGORITHM

The hydrological multi-feature algorithm is used to calculate the similarity between 2 flood process sequences.

The multi-feature variable time series similarity measure calculation is refined into three steps: firstly, input the historical flood process data and real-time flood process data; secondly, the flood process data are formatted into a feature variable time series; and finally, the time series similarity algorithm is used to calculate the sequence similarity (here we use the DTW algorithm to calculate the similarity of the time series). Ultimately, it is the historical flood process most similar to the real-time flood process that is sought in historical floods, and the historical flood process is used to make short-term predictions of real-time floods.

$$Dis(S_A, S_B) = \sum_{i=1}^{n} U_i * D_i, \quad s.t. \sum_{i=1}^{n} U_i = 1, \tag{10}$$

where $S_A = \{S_1, S_2, \cdots, S_n\}$, $S_B = \{S'_1, S'_2, \cdots, S'_n\}$, $D_i = DTW(S_i, S'_i)$, and $U_i \in U(|U| = n)$ is a kind of weight computed by PCA. Moreover, n is set 4 based on our practical problem.

2) ST-DTW ALGORITHM

To resolve the problem of similarity measure in hydrological studies using time series involving spatial dimension, this study proposes the ST-DTW algorithm. First, to satisfy data format requirements of the ST-DTW algorithm, the original data in the hydrological field were rasterized. Thereafter, the ST-DTW algorithm is described in detail. The description is mainly divided into two sections. The first section describes the similarity measure between the rainfall distribution matrices, and the second section further describes the similarity measure of the rainfall distribution matrices. Finally, the pseudocode of the ST-DTW algorithm is developed.

a: SIMILARITY MEASURE OF RAINFALL SPATIAL DISTRIBUTION MATRIX

A similarity measure between two rainfall distribution matrices is proposed. The two rainfall distribution matrices are R and T, where R is standard template rainfall distribution matrix and T is the rainfall distribution matrix of the test template; both are rainfall distribution matrices of n rows and m columns. The similarity between two matrices is defined as follows.

Definition 1, ($Matrix_{(R,T)}$): Distance of two rainfall distribution matrices.

$$Matrix_{(R,T)} = \begin{bmatrix} D_{R_1T_1} & D_{R_1T_2} & \cdots & D_{R_1T_n} \\ D_{R_2T_1} & D_{R_2T_2} & \cdots & D_{R_2T_n} \\ D_{R_3T_1} & D_{R_3T_2} & \cdots & D_{R_3T_n} \\ \vdots & \vdots & \ddots & \vdots \\ D_{R_mT_1} & D_{R_mT_2} & \cdots & D_{R_mT_n} \end{bmatrix} \tag{11}$$

where $D_{R_mT_n}$ represents the distance between the m^{th} row vector R_m of the standard rainfall distribution matrix R and the n^{th} row vector T_n of the template rain distribution matrix T (where the distance is calculated using the DTW).

e.q 1: The row vector of the rainfall distribution matrix denotes a division of the watershed into n subwatersheds, with each strip having an area of $1 * m(km^2)$. In practical applications, the calculated distance represents the similarity of rainfall over two long strips. Fig. 4 shows the rainfall distribution matrix in 10 rows and 8 columns ($n = 10, m = 8$). The $Matrix_{(R,T)}$ of two rainfall distribution matrices is the similarity matrix of 10 rows and 10 columns, in which the row vectors are $1 * 8$ matrix.

Definition 2, ($Dis_{(R,T)}$): The distance of the two $Matrix_{(R,T)}$.

$$Dis_{(R,T)} = Min\{DisWarping(Matrix_{(R,T)})\} \tag{12}$$

where $DisWarping(Matrix_{(R,T)})$ is the distance corresponding to each warping path in the matrix $Matrix_{(R,T)}$, which takes the minimum distance as the distance of the corresponding distance matrix; that is, the distance between the rainfall distribution R matrix and the rainfall distribution matrix T. In hydrology, this represents the degree of similarity of rainfall over the watershed at two points in time.

b: SIMILARITY MEASURE OF RAINFALL TEMPORAL DISTRIBUTION MATRICES

The previous section introduced the similarity measurement algorithm for two rainfall distribution matrices. Thereafter, we introduce the similarity calculation between the two sequences of rainfall distribution matrices (such as the sequence of rainfall distribution matrices shown in Fig. 5.

Definition 3, (S_R:) The training sequence of rainfall distribution matrices.

$$S_R = \{R_{t_1}, R_{t_2}, R_{t_3}, \ldots, R_{t_n}\} \tag{13}$$

Definition 4, (S_T:) The test rainfall distribution matrices sequence.

$$S_T = \{T_{t_1}, T_{t_2}, T_{t_3}, \ldots, T_{t_m}\} \tag{14}$$

where R_{t_n} represents the training rainfall distribution matrix at time t_n and, and T_{t_m} represents the test rainfall distribution matrix at time t_m.

Definition 5, ($Dis_{(S_R,S_T)}$): Distance between the training sequence of rainfall distribution matrices S_R and the test sequence of rainfall distribution matrices S_T.

$$Dis_{(S_R,S_T)} = DTW(S_R, S_T) \tag{15}$$

Definition 6, ($Matrix_{(S_R,S_T)}$): The distance of two different matrices sequence $Dis_{(S_R,S_T)}$.

$$Matrix_{(S_R,S_T)} = \begin{bmatrix} Dis_{(R_{t_1},T_{t_1})} & \cdots & Dis_{(R_{t_1},T_{t_m})} \\ Dis_{(R_{t_2},T_{t_1})} & \cdots & Dis_{(R_{t_2},T_{t_m})} \\ Dis_{(R_{t_3},T_{t_1})} & \cdots & Dis_{(R_{t_3},T_{t_m})} \\ \vdots & \ddots & \vdots \\ Dis_{(R_{t_n},T_{t_1})} & \cdots & Dis_{(R_{t_n},T_{t_m})} \end{bmatrix} \tag{16}$$

c: ST-DTW ALGORITHM PROCESS

As shown in Fig. 6, the ST-DTW algorithm structure is divided into two layers. The first layer calculates the similarity of two sequences of rainfall distribution matrices using the DTW algorithm, and the second layer calculates the similarity of rainfall distribution matrices.

From the first layer: the lengths of the standard template and test template of the sequence of rainfall distribution matrices are 3 4, respectively. Because of $Dis_{(S_R,S_T)} = DTW(S_R, S_T)$, we need to calculate the DTW distance between the sequences S_R and S_T. Similar to the one-dimensional DTW algorithm, we need to construct a matrix of $n{*}m$ (where n and m represent the lengths of the stand template and test template of the sequence of rainfall distribution matrices).

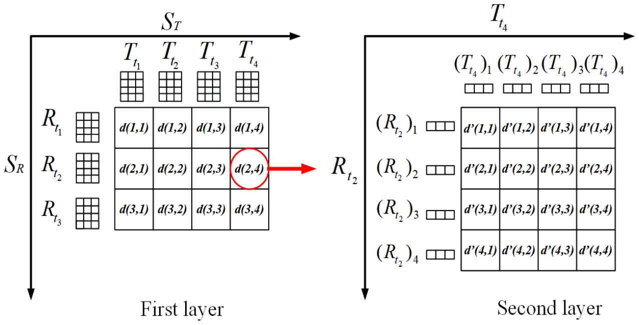

S_T
T_{t_1} T_{t_2} T_{t_3} T_{t_4}
S_R
R_{t_1}
R_{t_2}
R_{t_3}
d(1,1) d(1,2) d(1,3) d(1,4)
d(2,1) d(2,2) d(2,3) d(2,4)
d(3,1) d(3,2) d(3,3) d(3,4)
First layer
T_{t_4}
$(T_{t_4})_1$ $(T_{t_4})_2$ $(T_{t_4})_3$ $(T_{t_4})_4$
R_{t_2}
$(R_{t_2})_1$
$(R_{t_2})_2$
$(R_{t_2})_3$
$(R_{t_2})_4$
d'(1,1) d'(1,2) d'(1,3) d'(1,4)
d'(2,1) d'(2,2) d'(2,3) d'(2,4)
d'(3,1) d'(3,2) d'(3,3) d'(3,4)
d'(4,1) d'(4,2) d'(4,3) d'(4,4)
Second layer

FIGURE 6. **Rainfall distribution matrices sequences similarity hierarchical calculation.**

e.q 2: We construct a 3 $*$ 4 matrix, as shown in the first layer in Fig. 6. The difference between this algorithm and the one-dimensional DTW algorithm is that the one-dimensional DTW is a matrix between points that calculates the distance between the points. ST-DTW replaces points with matrices; that is, it calculates the distance between rainfall distribution matrices, where $d(1, 1)$ is the distance between the rainfall distribution matrix R_{t_1} at the time of t_1 of the standard template of the rainfall distribution matrices sequence and the rainfall distribution matrix T_{t_1} at the time of t_1 of the test template of the rainfall distribution matrices sequence. Similarly, $d(2, 4)$ is the distance between the rainfall distribution matrix R_{t_2} at time t_2 of the standard template of the rainfall distribution matrices sequence and the rainfall distribution matrix T_{t_4} at time t_4 of the test template of the rainfall distribution matrices sequence. The distance between two rainfall distribution matrices is calculated using Eq. 11 in **Definition 1** and Eq. 12 in **Definition 2**. After calculating the distance matrix of size 3 $*$ 4 in the first layer, the minimum cumulative distance is obtained, which is the similarity between the standard template and test template of the sequence of rainfall distribution matrices.

The matrix shown in the second layer in Fig. 6 is an expanded description of the first level $d(2, 4)$ distance calculation. The distance $d(2, 4) = Dis_{(R_{t_2},T_{t_4})}$ between two rainfall distribution matrices, $d'(1, 1)$ represents the DTW distance between the first row vector $(R_{t_2})_1$ of the rainfall distribution matrix R_{t_2} and the first row vector $(T_{t_4})_1$ of the rainfall distribution matrix T_{t_4}. The distance matrix $Matrix_{(R_{t_2},T_{t_4})}$ of two rainfall distribution matrices is calculated using Eq. 12 in **Definition 2**.

```
Algorithm 1 Similarity Between $S_R$ and $S_T$
Require: $S_R = \{R_{t_1}, R_{t_2}, \ldots, R_{t_n}\}$; $S_T = \{T_{t_1}, T_{t_2}, \ldots, T_{t_n}\}$;
    $L_{S_R}$;$L_{S_T}$; $MatrixRows$;
Ensure: $Dis_{(S_R,S_T)}$;
procedure $generateMatrix_{(S_R,S_T)}$
    while i $\leq L_{S_R}$ do
        while j $\leq L_{S_T}$ do
            $Dis_{(R_{t_i},T_{t_j})} = generateMatrix_{(R_{t_i},T_{t_j})}(R_{t_i}, T_{t_j})$
        end while
    end while
return $Dis_{(S_R,S_T)} = Min\{DisWarping(Matrix_{(S_R,S_T)})\}$
end procedure
procedure $generateMatrix_{(R_{t_i},T_{t_j})}(R_{t_i}, T_{t_j})$
    while x $\leq$ MatrixRows do
        while y $\leq$ MatrixRows do
            $D_{((R_{t_i})x,(T_{t_j})y)} = DTW((R_{t_i})_x, (T_{t_j})_y)$
        end while
    end while
return $Dis_{(R_{t_i},T_{t_j})} = Min\{DisWarping(Matrix_{(R_{t_i},T_{t_j})}\}$
end procedure
```

V. EXPERIMENT

This section describes the experiments in detail, including datasets, experimental methods, performance criteria, experimental results, and analysis.

In addition, we conduct a set of experiments to validate rainfall-flow pattern based on hydrological data released by the China Shaanxi and Zhejiang hydrology Administration. The experiments are designed to investigate the following three research questions:

- RQ1: Is the simulated performance of the rainfall-flow pattern better than the baselines as it becomes worse as the forecast timestep increases?
- RQ2: Does rainfall-flow pattern outperform deep learning models or approaches?
- RQ3: Does watersheds with different soil water content affect forecasting accuracy?

A. DATA SETS

Watersheds with different soil water content have different rainfall-flow patterns. We selected two study watersheds: a watershed in drought with less annual average rainfall (the *Heihe* watershed, Shanxi, China); and a wet watershed with more annual average rainfall (the *Changhua* watershed, Zhejiang, China). The hydrological and precipitation stations in the *Changhua* and *Heihe* watersheds are shown in Fig.7 and Fig.8, respectively.

As shown in Table 1, we selected hydrological data of the *Changhua* watershed from 1998 to 2010, containing 31 flood events. Correspondingly, the hydrological data of the *Heihe* watershed from 2003 to 2012 contained 29 flood events. In the *Changhua* watershed, we selected data from 1998 to 2008 as a training set, and data from 2009 to 2010 as a test set. In the *Heihe* watershed, we selected the data from 2003 to 2010 as a training set, and the data from 2011 to 2012 as a test set. The data collection frequency of rainfall stations and hydrological stations in the *Changhua* and *Heihe* watersheds is hourly.

TABLE 1. Details of watershed datasets.

Watershed	Changhua	Heihe
Type of soil	Wet	Drought
Type of climate	Frequent rainfall	Less rainfall
Time interval	1 hour	1 hour
Time span	1/1/1998 - 12/31/2010	1/1/2003 - 12/31/2012
Training set	1/1/1998 - 12/31/2008	1/1/2003 - 12/31/2010
Test set	1/1/2009 - 12/31/2010	1/1/2011 - 12/31/2012

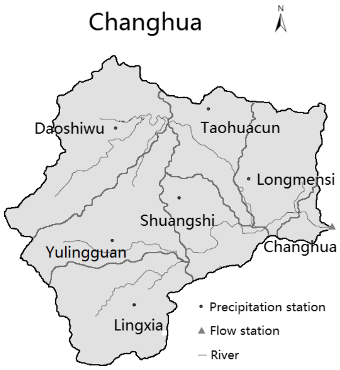


FIGURE 7. The boundary of the *Changhua* watershed, as well as the marking of hydrological stations and rainfall stations in the watershed.

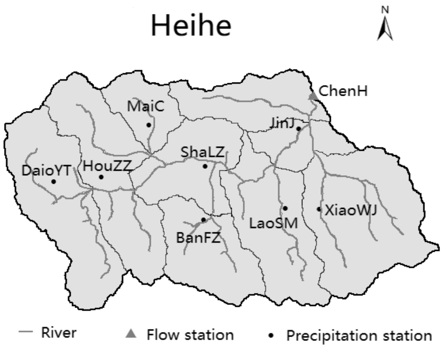


FIGURE 8. The boundary of the *Heihe* watershed, as well as the marking of hydrological stations and rainfall stations in the watershed.

B. BASELINE MODELS

In what follows, we compare our two proposed models with three baseline models. The first baseline is a traditional statistical method, and the second baseline is a classic machine learning method. The third baseline is a deep learning method. We detail them as follows:

TABLE 2. ***RMSE*** **comparison among various models in simulating stream-flow for next 1-10 h in the** ***Changhua*** **watershed.**

Models	Simulated time step									
	1h	2h	3h	4h	5h	6h	7h	8h	9h	10h
ARIMA	282.11	330.25	331.85	362.17	390.82	418.56	441.21	465.18	481.32	501.13
SVM	170.22	182.32	196.99	213.74	233.74	254.68	270.68	296.28	336.84	368.65
LSTM	152.65	154.36	162.54	182.52	200.25	216.82	230.61	242.18	256.35	270.65
Multi-feature	146.72	159.73	176.6 1	192.01	206.28	218.82	230.24	240.25	248.76	254.42
ST-DTW	168.77	178.92	189.93	201.86	213.42	221.95	228.06	233.57	239.15	243.60

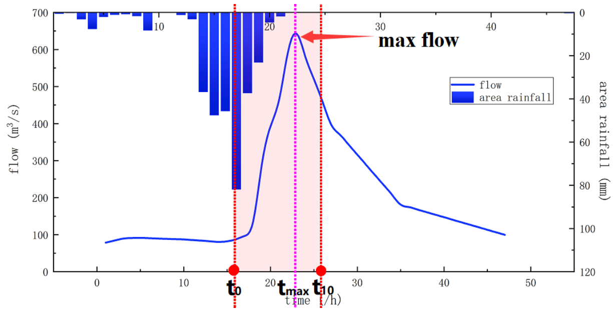


FIGURE 9. **Description of the flood forecasting process.**

- Autoregressive integrated moving average (ARIMA), a classic time series model used to predict flooding in hydrological studies in the 1990s [34].
- Support vector machine (SVM), an AI model based on structural risk minimization that uses the local rainfall and stream-flow data to predict the future stream-flow [35].
- Long short-term memory (LSTM): A time series model used to predict future river stream-flow by considering past hydrological, past rainfall, and future weather forecast data [36].

C. EXPERIMENTAL METHODS

A flood process of the experiment is shown in Fig. 9. When selecting the flood process in the historical flood data, starting point of the flood process is at time $t = 0$ in the figure, and $t = t_{max}$ is the onset of the flood peak. The starting forecast time is $t = t_0$, and the forecast period is set to $t_1 \sim t_{10}$.

Our models and baseline models were completely implemented on a computer workstation with an NVIDIA TITAN V GPU running the Ubuntu 16.04 operating system using the Python 3.6 programming language. The input step size of our models was the length of time from the onset of rainfall to the starting forecast point. For the baseline models, we used input and parameter settings based on the best results in the papers [34]–[36]. In particular, for the deep learning model (LSTM), the structure was constructed using the TensorFlow software library, with the learning rate (LR) set to 0.001 and the batch size set to 128.

D. PERFORMANCE CRITERIA

In this subsection we present three common real-time hydrological forecasting indicators.

- Root mean square error.

To illustrate the degree of dispersion of the sample, the mean square error is calculated as Eq. 17

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i^{pre} - y_i^{mea}\right)^2} \tag{17}$$

where y_i^{pre} is the predicted flow at time i; y_i^{mea} denotes the measured flow at time i; and n denotes the number of test set samples.

- Predicting the time error of the flood peak.

$$T = |max(T^{pre}) - max(T^{mea})| \tag{18}$$

where $max(T^{pre})$ is the time at maximum forecast flow, and $max(T^{mea})$ is the time at maximum measured flow.

- Predicting the flow value error of the flood peak

$$D = \frac{|max(y^{pre}) - max(y^{mea})|}{max(y^{mea})} \times 100\% \tag{19}$$

where $max(y^{pre})$ is the maximum value of forecast flow, and $max(y^{mea})$ is the maximum value of measured flow.

E. EXPERIMENTAL RESULTS AND ANALYSIS

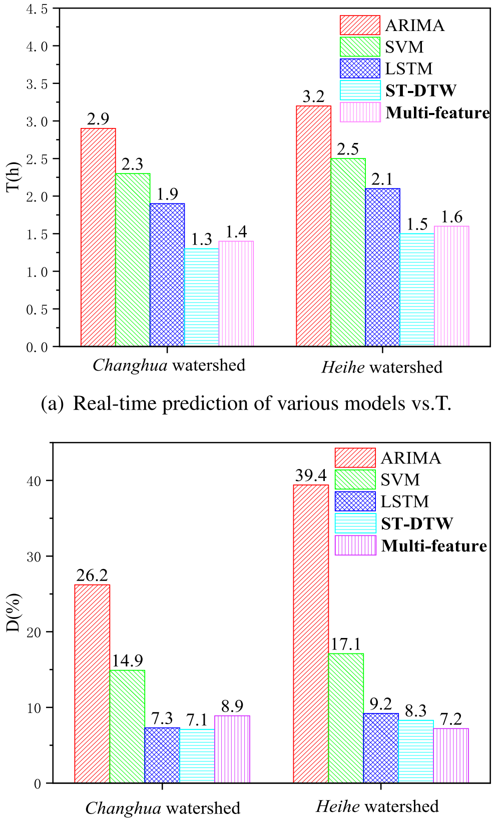
We compare the effectiveness of our models with the baseline models on the two watershed datasets. First, we show the best simulated performance of each model in Table2 and Table3. Second, to explore the proposed real-time prediction performance of the models, we calculated the average time error T (Fig.10(a)) and the average stream-flow value error D (Fig.10(b)) in all flood peaks. Finally, to visualize the real-time forecasting effect of the proposed model, we selected a real-time flood forecasting event from the *Changhua* watershed (Fig.11(a)) and the *Heihe* watershed (Fig.11(b)).

1) STUDY THE MEASURED PERFORMANCE OF VARIOUS MODELS

For the simulation performance measured in terms of the RMSE of various models, the results of experiments on the Changhua and Heihe watershed dataset are shown

TABLE 3. ***RMSE* comparison among various models in simulating stream-flow for next 1-10 h in the *Heihe* watershed.**

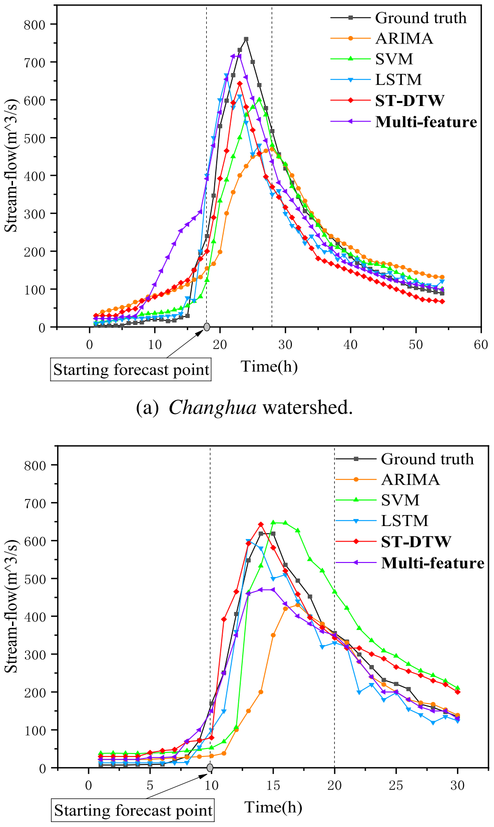
Models	Simulated time step									
	1h	2h	3h	4h	5h	6h	7h	8h	9h	10h
ARIMA	221.11	268.52	301.25	351.58	380.82	414.35	438.36	465.89	482.68	521.25
SVM	132.58	140.98	156.85	172.64	190.51	229.74	268.65	302.58	356.54	392.51
LSTM	107.22	112.32	120.23	136.21	150.43	178.85	200.64	232.21	266.95	302.23
Multi-feature	113.72	125.34	134.67	158.61	175.38	198.19	217.64	234.71	252.62	269.94
ST-DTW	119.66	128.55	141.53	164.77	187.88	204.37	215.34	228.73	241.27	257.46



(a) Real-time prediction of various models vs.T.

(b) Real-time prediction of various models vs.D.

FIGURE 10. **Performance comparisons of real-time prediction by the three three baseline models and our models. (a) Comparison of real-time prediction performance T(defined in Eq. 18) of various models.
(b) Comparison of real-time prediction performance D(defined in Eq. 19) of various models.**



(a) *Changhua* watershed.

(b) *Heihe* watershed.

FIGURE 11. **Comparison with the ground truth river stream-flow and real-time predicted stream-flow computed by the three baseline models and our models.**

in Table 2 and 3, respectively. It is quite apparent that the deep learning LSTM model in the three baseline models has a best simulation performance (have minimum RMSE value, the smaller is the better). The RMSE performance of proposed models are comparable to the best in three baseline models. In addition, as the simulation timestep increases, the RMSE of each model correspondingly improved. In particular, the RMSE increase in our multi-feature and ST-DTW models converges, compared with the RMSE increase of the three baseline models.

Our rainfall-flow pattern is not a traditional AI model. Compared with the deep learning model, the multi-feature and ST-DTW models are not limited by the training samples, which are insufficient to support the performance degradation caused by the increase of simulation timestep.

2) STUDY THE REAL-TIME FORECAST PERFORMANCE OF VARIOUS MODELS

The real-time forecasting performance of a model is the focus of hydrologists researching small and medium watersheds. The real-time forecasting performance is mainly reflected in the peak position, which is generally measured by the time error and stream-flow value error of the flood peak.

Our proposed multi-feature model and ST-DTW model perform better than the baseline models in terms of the time error T (defined in Eq. 18) of the flood peak. As shown in Fig.10(a), there is no doubt that the deep learning model (LSTM) has the best real-time prediction performance (the smaller T value is, the better) among the three baseline models. It is worth noting that the Multi-feature model and ST-DTW model have outstanding T values compared with the baseline models on the datasets of the two watersheds. This shows that our models have an absolute advantage in terms of flood warning capability for small and medium rivers.

As shown in Fig.10(b), the proposed multi-feature and ST-DTW models have a performance D (defined in Eq. 19) that is similar to that of the LSTM model. This shows that our models offer no significant advantage in terms of the error of the peak stream-flow value.

However, although the LSTM is the best performing model, it has a flood predicting process line that is unsmooth for real-time forecasting. As shown in Fig.11, the LSTM model has fluctuations in the flood forecasting process line on the two watershed datasets.

3) STUDY THE REAL-TIME FORECAST PERFORMANCE IN DIFFERENT SOIL WATER CONTENT WATERSHEDS

Different soil water content watersheds have their own rainfall-flow patterns, especially for dry and wet watersheds. As shown in Table 2 and 3, the degree of deterioration of various models on the *Heihe* watershed data increases with increasing simulation times.

In addition, as shown in Fig.11, the real-time flood forecast lines of various models on the *Heihe* watershed data are more dispersed. In particular, the process line fluctuations after 6 h are more obvious. Moreover, as shown in Fig.11(b), compared with the *Changhua* watershed, the *Heihe* watershed in a drought-ridden area has a more complex rainfall-flow pattern, which makes it more challenging to mine with ordinary AI models to achieve the desired effect with small datasets.

VI. CONCLUSION

Traditional AI models do not achieve expected results when real-time forecasting is performed on small and medium watersheds with limited hydrological data. Thus, we developed a different model: using a rainfall-flow pattern based on historical rainfall and flood flow data for real-time predictions of short-term flood stream-flow. Our models predict the flood process line in real-time using hydrological feature extraction and spatial-temporal metrics for similar rainfall-flow patterns. The experimental results based on the datasets of various models in wet and drought-ridden watersheds show that the proposed models offer considerable advantages in accurately predicting the peak time of floods in real time.

In future research, to improve the sample size and quality of small and medium watershed data, we will explore the fusion of radar rainfall data with ground station data. Moreover, we will endeavor to test the rainfall-flow pattern over large watersheds with ample hydrometeorological data. And we will consider large-scale watershed discharge forecasts and other types of discharge forecasts (such as urban underground drainage and high-sand rivers) forecasts. Finally, a fusion of radar rainfall data and ground station data could be developed.

REFERENCES

[1] Y. Zhang, Y. Hong, X. Wang, J. J. Gourley, J. Gao, H. J. Vergara, and B. Yong, "Assimilation of passive microwave streamflow signals for improving flood forecasting: A first study in Cubango River Basin, Africa," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 6, no. 6, pp. 2375–2390, Dec. 2013.

[2] B. He, X. Huang, and L. Guo, "China's mountain flood disaster prevention route and core construction content," *China Flood Drought Manage.*, vol. 22, no. 5, pp. 19–22, 2012.

[3] A. Behrangi, B. Khakbaz, T. C. Jaw, A. Aghakouchak, K. Hsu, and S. Sorooshian, "Hydrologic evaluation of satellite precipitation products over a mid-size basin," *J. Hydrol.*, vol. 397, nos. 3–4, pp. 225–237, Feb. 2011.

[4] E. Toth, A. Brath, and A. Montanari, "Comparison of short-term rainfall prediction models for real-time flood forecasting," *J. Hydrol.*, vol. 239, nos. 1–4, pp. 132–147, Dec. 2000.

[5] F. Fotovatikhah, M. Herrera, S. Shamshirband, K.-W. Chau, S. F. Ardabili, and M. J. Piran, "Survey of computational intelligence as basis to big flood management: Challenges, research directions and future work," *Eng. Appl. Comput. Fluid Mech.*, vol. 12, no. 1, pp. 411–437, Jan. 2018.

[6] J. Reynolds, S. Halldin, C. Xu, J. Seibert, and A. Kauffeldt, "Sub-daily runoff predictions using parameters calibrated on the basis of data with a daily temporal resolution," *J. Hydrol.*, vol. 550, pp. 399–411, Jul. 2017.

[7] Z. M. Yaseen, A. El-Shafie, O. Jaafar, H. A. Afan, and K. N. Sayl, "Artificial intelligence based models for stream-flow forecasting: 2000–2015," *J. Hydrol.*, vol. 530, pp. 829–844, Nov. 2015.

[8] J. Feng, L. Yan, and T. Hang, "Stream-flow forecasting based on dynamic spatio-temporal attention," *IEEE Access*, vol. 7, pp. 134754–134762, 2019.

[9] C. P. Chen and C.-Y. Zhang, "Data-intensive applications, challenges, techniques and technologies: A survey on big data," *Inf. Sci.*, vol. 275, pp. 314–347, Aug. 2014.

[10] H. Gao, W. Huang, and X. Yang, "Applying probabilistic model checking to path planning in an intelligent transportation system using mobility trajectories and their statistical data," *Intell. Automat. Soft Comput.*, vol. 25, no. 3, pp. 547–559, 2019.

[11] H. Gao, W. Huang, Y. Duan, X. Yang, and Q. Zou, "Research on cost-driven services composition in an uncertain environment," *J. Internet Technol.*, vol. 20, no. 3, pp. 755–769, 2019.

[12] H. Gao, Y. Duan, L. Shao, and X. Sun, "Transformation-based processing of typed resources for multimedia sources in the iot environment," *Wireless Netw.*, pp. 1–17, Nov. 2019.

[13] R. C. Deo and M. Şahin, "Application of the artificial neural network model for prediction of monthly standardized precipitation and evapotranspiration index using hydrometeorological parameters and climate indices in eastern Australia," *Atmos. Res.*, vols. 161–162, pp. 65–81, Jul. 2015.

[14] R. C. Deo, P. Samui, and D. Kim, "Estimation of monthly evaporative loss using relevance vector machine, extreme learning machine and multivariate adaptive regression spline models," *Stochastic Environ. Res. risk Assessment*, vol. 30, no. 6, pp. 1769–1784, Aug. 2016.

[15] Y. Bai, Z. Chen, J. Xie, and C. Li, "Daily reservoir inflow forecasting using multiscale deep feature learning with hybrid models," *J. Hydrol.*, vol. 532, pp. 193–206, Jan. 2016.

[16] M. Şahin, Y. Kaya, M. Uyar, and S. Yıldırım, "Application of extreme learning machine for estimating solar radiation from satellite data," *Int. J. Energy Res.*, vol. 38, no. 2, pp. 205–212, 2014.

[17] F. Liu, F. Xu, and S. Yang, "A flood forecasting model based on deep learning algorithm via integrating stacked autoencoders with bp neural network," in *Proc. IEEE 3rd Int. Conf. Multimedia Big Data (BigMM)*, Apr. 2017, pp. 58–61.

[18] F. Kratzert, D. Klotz, C. Brenner, and K. Schulz, "Rainfall-runoff modelling using long short-term memory (LSTM) networks," *Hydrol. Earth Syst. Sci.*, vol. 22, no. 11, pp. 6005–6022, 2018.

[19] B. R. Bicknell, J. C. Imhoff, J. L. Kittle, Jr., A. S. Donigian, Jr., and R. C. Johanson, "Hydrological simulation program—FORTRAN user's manual for version 11," US Environ. Protection Agency, Athens, GA, USA, Environ. Protection Agency Rep. EPA/600/R-97/080, 1997.

[20] R. J. Burnash, R. L. Ferral, and R. A. McGuire, "A generalized streamflow simulation system: Conceptual modeling for digital computers," US Dept. Commerce, Nat. Weather Service, State California, Silver Spring, MD, USA, Tech. Rep., 1973.

[21] V. P. Singh, *Computer Models of Watershed Hydrology*. 1995.

[22] R.-J. Zhao, "The xinanjiang model," in *Proc. Oxford Symp.*, 1980.

[23] K. Beven, R. Warren, and J. Zaoui, "SHE: Towards a methodology for physically-based distributed forecasting in hydrology," IAHS Pub., Wallingford, U.K., Tech. Rep., 1980, vol. 129, pp. 133–137.

[24] S. Salcedo-Sanz, R. C. Deo, L. Carro-Calvo, and B. Saavedra-Moreno, "Monthly prediction of air temperature in Australia and New Zealand with machine learning algorithms," *Theor. Appl. Climatol.*, vol. 125, nos. 1–2, pp. 13–25, Jul. 2016.

[25] Z. Zhang, Q. Zhang, and V. P. Singh, "Univariate streamflow forecasting using commonly used data-driven models: Literature review and case study," *Hydrol. Sci. J.*, vol. 63, no. 7, pp. 1091–1111, May 2018.

[26] K.-L. Hsu, H. V. Gupta, X. Gao, S. Sorooshian, and B. Imam, "Self-organizing linear output map (SOLO): An artificial neural network suitable for hydrologic modeling and analysis," *Water Resour. Res.*, vol. 38, no. 12, pp. 38-1–38-17, Dec. 2002.

[27] J. Guo, J. Zhou, H. Qin, Q. Zou, and Q. Li, "Monthly streamflow forecasting based on improved support vector machine model," *Expert Syst. Appl.*, vol. 38, no. 10, pp. 13073–13081, Sep. 2011.

[28] G. J. Klir and B. Yuan, *Fuzzy Sets and Fuzzy Logic: Theory and Applications*. Upper Saddle River, NJ, USA, 1995, p. 563.

[29] A. El-Shafie, M. R. Taha, and A. Noureldin, "A neuro-fuzzy model for inflow forecasting of the Nile river at Aswan high dam," *Water Resour Manage*, vol. 21, no. 3, pp. 533–556, Feb. 2007.

[30] D. A. Savic, G. A. Walters, and J. W. Davidson, "A genetic programming approach to rainfall-runoff modelling," *Water Resour. Manage.*, vol. 13, no. 3, pp. 219–231, 1999.

[31] S. Galelli and A. Castelletti, "Tree-based iterative input variable selection for hydrological modeling," *Water Resour. Res.*, vol. 49, no. 7, pp. 4295–4310, Jul. 2013.

[32] N. Jean, M. Burke, M. Xie, W. M. Davis, D. B. Lobell, and S. Ermon, "Combining satellite imagery and machine learning to predict poverty," *Science*, vol. 353, no. 6301, pp. 790–794, Aug. 2016.

[33] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley Interdiscipl. Rev., Comput. Statist.*, vol. 2, no. 4, pp. 433–459, 2010.

[34] H. M. Awwad, J. B. Valdés, and P. J. Restrepo, "Streamflow forecasting for Han River basin, Korea," *J. Water Resour. Planning Manage.*, vol. 120, no. 5, pp. 651–673, Sep. 1994.

[35] J.-Y. Lin, C.-T. Cheng, and K.-W. Chau, "Using support vector machines for long-term discharge prediction," *Hydrol. Sci. J.*, vol. 51, no. 4, pp. 599–612, Aug. 2006.

[36] L. Yan, J. Feng, and T. Hang, "Small watershed stream-flow forecasting based on LSTM," in *Proc. Int. Conf. Ubiquitous Inf. Manage. Commun.* Cham, Switzerland: Springer, 2019, pp. 1006–1014.

YUELONG ZHU is currently a Professor with the College of Computer and Information, Hohai University, Nanjing, China. His main research interests include intelligent information processing and data mining, and water conservancy informatization.

JUN FENG received the Ph.D. degree in computer science from Nagoya University, in 2004. She is currently a Professor with the College of Computer and Information, Hohai University, Nanjing, China. Her research interests include space-time data management, data mining, water conservancy big data, and water conservancy informatization.

LE YAN is currently pursuing the Ph.D. degree with the College of Computer and Information, Hohai University, Nanjing, China. His research interests include hydrological spatio-temporal data mining, hydrological time series data prediction, and deep learning modeling.

TAO GUO is currently pursuing the master's degree with the College of Computer and Information, Hohai University, Nanjing, China. His current research interests include data mining and hydrological forecasting.

XIAODONG LI received the B.Sc. degree from the Department of Computer Science and Technology, Nanjing University, in 2006, and the Ph.D. degree from the City University of Hong Kong, in 2014, under the supervision by Prof. D. Xiaotie. His research interests include quantitative trading and hydrological informatics.