International Journal of Smart Home
Vol. 9, No. 12, (2015), pp. 81-90
http://dx.doi.org/10.14257/ijsh.2015.9.12.09

A Big-Data-Based Urban Flood Defense Decision Support System

Taimeng Yang[1,2], Guanlin Chen[1,2 *] and Xinxin Sun[3]

[1]*School of Computer and Computing Science, Zhejiang University City College, Hangzhou, 310015, P.R. China*
[2]*College of Computer Science, Zhejiang University, Hangzhou, 310027, P.R. China*
[3]*Department of Computer Science and Information, Zhejiang University of Water Conservancy and Electric Power, Hangzhou, 310018, P.R. China*
**Corresponding author email: chenguanlin@zucc.edu.cn*

Abstract

As cities in developing countries are expanding rapidly in recent years, flood has an increasing impact on urban management. In this paper, we present the design and implementation of an urban flood defense decision support system based on big data. The system connects real-time sensor to collect streaming data, and uses a data-driven method that considers temporal and spatial factors to forecast water level in the next 6 hours. Thus, it can provide enough time for the authorities to take pertinent flood protection measures such as evacuation. Our predictive model is a hybrid of linear regression and artificial neural network, and can give early warning of potential flood using the forecast results. The system is implemented on Java EE platform, and integrated with Baidu Maps API to provide a user-friendly interface.

Keywords: *Flood Defense; Neural Network; Digital Urban Management; Java EE*

1. Introduction

Many developing countries, such as China, India, and Brazil, urbanization's rapid progress has endangered lots of challenges. Given the complexities and dynamic settings of city, it is very difficult to tackle the challenges. However, smart sensors become cheap and pervasive in cities, these sensors produce a variety of big data in urban spaces (e.g., air quality, water level, rainfall, traffic patterns and geographical data).The big data implies rich knowledge of a city and can help us tackle these challenges when used correctly [1].

In recent years, researchers have done a lot of researches on tackling urban development and management challenges using big data. Zheng Yu et al. used big data to forecast fine- grained air quality [2], diagnose urban noise [3], and estimate urban energy consumption [4]. 2009, F. A. Pugliese et al. used a large scale floating car data system to analyze urban traffic [5]. 2010, He Yongxiu et al. proposed an urban residential load combined forecast model based on data mining techniques and panel data theory [6]. 2015, Guanlin Chen et al. designed an intelligent analysis and mining system for urban lighting information [7].

One of the challenges in urbanization's progress is urban flood. Many big cities in China, such as Beijing, Shanghai, Guangzhou, Hangzhou, just have to deal with flood risk management issues on a regular basis. These are issues that will worsen as climate change effects result in more extreme conditions. Since population density in cities is usually high, occurrences of flood in urban area will not only cause damage on properties, but even threaten people's lives. In the last few years, a large number of projects aim at the development of stronger and "smarter" flood protection systems have been initiated around the world. Most of them aim to solve flood control problems [8]. In this paper, we

ISSN: 1975-4094 IJSH

propose a Big-data-based Urban Flood Defense Decision Support System (BUFDDSS) that connects with real-time sensor to collect streaming data as input, and combine with a water level predictive model to give an early warning for potential flood.

BUFDDSS is developed on the mainstream of the Java EE platform using the MVC pattern. The system employs Spring+Struts2+Hibernate framework and MySQL database. The system also integrates with HighCharts data visualization framework and Baidu Maps API to provide an interactive and friendly user interface.

2. 2. Overview of System

2.1. Framework of the System

The system is composed of three major parts, sensors as external data sources, a server to provide service for real-time query, water level forecast and flood early warning, and clients. Figure 1 presents the framework of BUFDDSS.

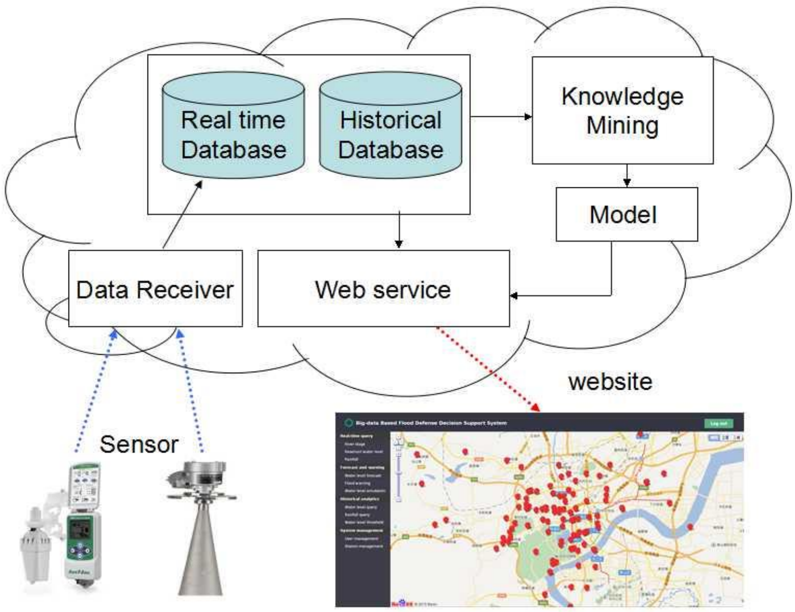

Figure 1. The Framework of BUFDDSS

The sensors include water level sensors and rain sensors, these sensors could send real time data back to the server side. The server side has a data receiver module to handle the data sent back by the sensors. Theses sensors collect data every 5 minutes, so volume of the database will become large in a few years, query for the latest data will not respond in a reasonable time, thus we use a separate database to store real time data and historical data. The real time database stores the latest 24 hours data. The model to forecast water level is trained with the historical data using machine learning technique. The model uses the real time streaming data as input to predict water levels in the next 6 hours. Clients could query these predicted water results through web service. The web service module also provides analytics of historical data.

2.2. Website Architecture

The website provides the user interface of BUFDDSS. Figure 2 presents the architecture of the website. There are four major parts: system management, real-time query, historical analytics, and forecast and warning.

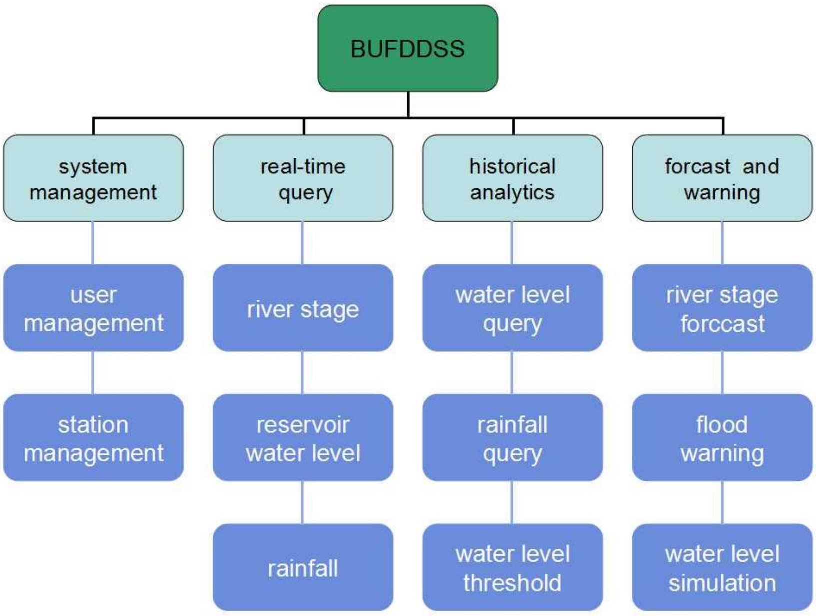

Figure 2. The Architecture of Website

(1) System management, which includes user management and station management. Administrator of the system could add or delete user and manage user account permissions through user management module, and add or delete sensor stations via the station management module.

(2) Real-time query, which provides query for real-time river stage, reservoir water level, and rainfall.

(3) Historical analytics, this module provides descriptive analytics for historical data, such as maximum, minimum value observed in past years, variance and distribution of water levels in a year, frequency of water level threshold exceeded at a given station in past years.

(4) Forecast and warning, in this module user could query forecast water level produced by the predictive mode. If the forecast results exceed threshold at a station, there will be a warning sign displayed on the website, thus decision makers could make a decision on how to deal with it. There is an interactive simulation on water level under given rainfall in this module. Since rainfall is the direct factor on water level, when rainfall is given, water level could be computed with the predictive model. Decision maker could input rainfall in the next few hours, and then website will display the computed result water level under the input rainfall.

3. System Implementation

BUFDDSS is implemented on the mainstream Java EE platform, integrated with SSH (Spring+Struts2+Hibernate) framework, and applied the MVC pattern. The system integrates with Baidu Maps API to provide a user-friendly interface, and uses a data visualization library (HighCharts) to provide visual presentation of data. In the following, we will give a detailed description of some key modules in the system.

3.1. Database Design

The system uses MySQL 5.5 database and there are nine major tables designed. These tables include basic station information (st_stbprp_b), river stage historical data (st_river_r0), river stage real time data (st_river_rt), reservoir water level historical data (st_rsvr_r0), reservoir water level real time data (st_rsvr_rt), rainfall historical data (st_pptn_r0), rainfall real time data (st_pptn_rt), warning threshold for river and reservoir (st_warn), and user account (tbl_user).

The relationships of these tables are as follows: the user in tbl_user is for validation when a user logs in the system. st_stbprp_b stores basic information of all water level and rainfall sensors. Every sensor has a unique id which is stored in the stcd column in st_stbprp_b. st_stbprp_b also stores longitude and latitude of all sensors. Each row in st_river_r0, st_rsvr_r0 and st_pptn_r0 stores a record produced by a sensor. The three tables have stcd and time as its composite primary key. The real time tables have same fields as the corresponding historical table, except they just store the latest 24 hours sensors generated data.

3.2. Geographical Display with Baidu Maps

In the current implemented system, there are 113 water level sensors and 128 rainfall sensors, so it's hard to query data by type the id of a sensor. Thus we overlay the sensor with a marker on Baidu Maps, and the user could query a sensor by hover or click on the correspondent marker. The Javascript code of using Baidu Maps is as follows.

```
<script type="text/javascript" src="http://api.map.baidu.com/api?v=1.2"></script>
var map;
$(document).ready(function() {
    map = new BMap.Map("container"); // create a map instance
    map.disableDoubleClickZoom();  // disable zoom by double click
    map.enableScrollWheelZoom();  // enable zoom by scroll mouse wheel
    map.addControl(new BMap.NavigationControl({//add navigation controller on map
        type : BMAP_NAVIGATION_CONTROL_LARGE
    }));
    map.addControl(new BMap.MapTypeControl()); // add map type controller on map
    map.centerAndZoom('Hangzhou');  // set Hangzhou as map center
});
```

The javascript code of adding marker of sensor on the map is as follows.

```
var stationMarkers = new Array();  // ceate an array to store station markers
  // jQuery getJSON function will get the latest 24 hours data from server,
// and display the station marker on map
$.getJSON('../station/getAll',  function(data) {
    for( var i=0; i<data.count; i++) {
        var lng = data.list[i].lng;
        var lat = data.list[i].lat;
 // create and initialize the marker
        var marker = new BMap.Marker(new BMap.Point(lng, lat));
        marker.div = document.createElement("div");
        marker.div.style.position = "absolute";
        marker.div.style.width = "auto";
        marker.div.style.height = "auto";
        marker.div.innerHTML="<h3>" + data.list[i].id + "</h3>";
// display latest 24 hours data
        var infoWindow = new BMap.InfoWindow($("#canvas")[0]);
// display infoWindow when mouse over the marker
```



```
        marker.addEventListener("mouseover", function(){
            this.openInfoWindow(infoWindow);});
// close infoWindow when mouse move out the marker
        marker.addEventListener("mouseout", function(){
         this.closeInfoWindow();
        });
        map.addOverlay( marker );    // add the marker on map
    }
});
```

After a map object is created, the web page will get all the stations information from the server via AJAX, and the latest 24 hours water levels and rainfall data. Every station will be presented as a marker on the map, when the user hover mouse on a marker, there will pop up an info window to display the latest 24 hours data. The interface is shown in Figure 3.

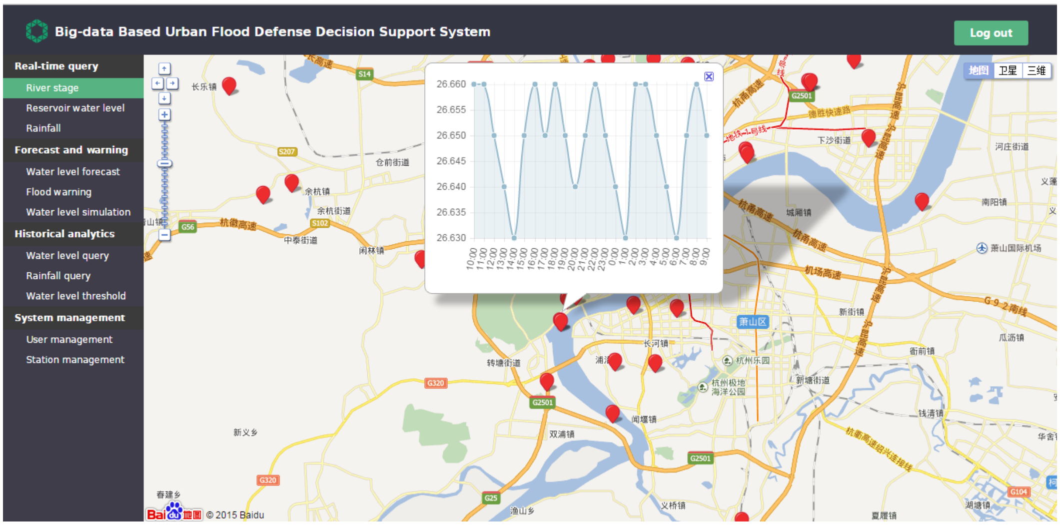

Figure 3. The User Interface of Real-time Query

3.3. Visual Analytics

To date, we have collected water level and rainfall sensors data in 2011-2014. These data are almost 5GB. There is rich knowledge implied in those massive data, but just query the original data from the database is hard to find a useful pattern. So we need a better presentation of those data. A good visualization of original data can reveal knowledge intuitively so as to help in decision making. The visualization of data may also suggest the correlation or causality between different factors. So we provide a visual presentation of historical data.

User can select the sensor which he interested from Baidu Maps, and then a page for inquiring that sensor will show. Users can type more detailed parameters in that page to get a visual descriptive representation of data generated by that sensor in past years. A sensor generates data every 5 minutes. There would be more than 100000 in a year. When a user inquiries data by the year, it’s impractical to show all of the data on a single page. To address this issue, the system will aggregate those data by the day, and then respond with a reasonable volume of data back to the client. If user wants to inquiry data in more fine-grained, such as water level series in a day, server will respond with original data, thus provide more information to users. Distribution of water level in that time range query by user and frequency of water level exceeds the warning threshold will display on that correspondent page. Figure 4 presents the visual analytic page for a river stage sensor.

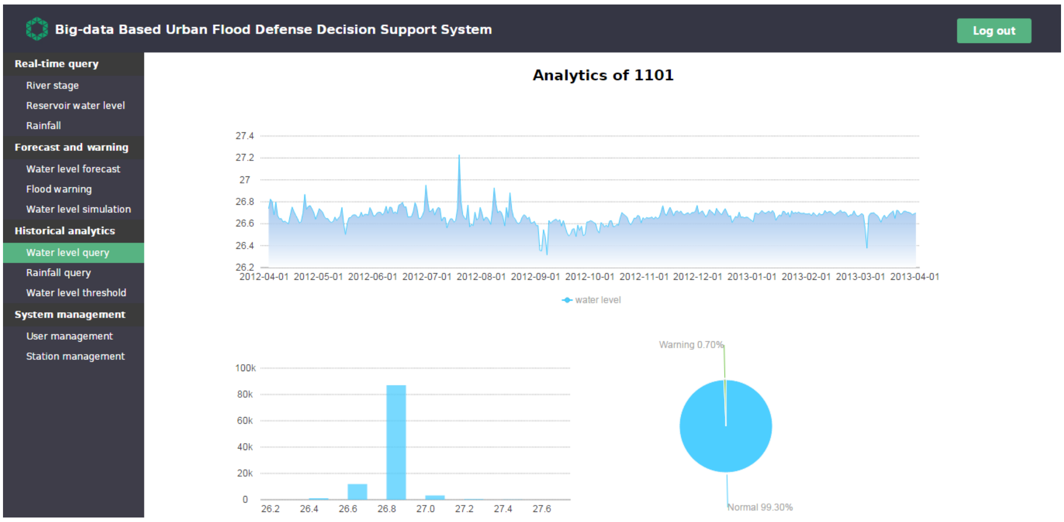

Figure 4. The Historical Analytics of a Station

3.4. Framework of the Predictive Model

There are a large number of predictive models of flood proposed and real time forecast of river stage. River stage is not fit in a linear regression model very well, so most of those models applied artificial neural network [9], support vector machine [10] and dynamic Bayesian network [11] or a hybrid of these techniques [12].

Our model is a hybrid of linear regression and artificial neural network, which considers local temporal and global spatial factors. The predictive model consisting of two components: a temporal predictor and a spatial predictor. The temporal predictor predicts the water level of a station in terms of the data about the station, which is water level of the past few hours. The temporal predictor is based on a linear regression, which models the water level regression process. The spatial predictor considers all sensor stations current state to predict a station's future water level. A station's water level is influenced not only by its past water level, but also has a spatial correlation with other stations, a station at downstream will have a correlation with stations in upstream. The spatial predictor is based on artificial neural network (ANN), modeling the spatial correlation and predicting water level from other stations' points of view.

(1) Temporal Predictor.

The temporal predictor models the trend of water level of a station based on its past few hours of data. Intuitively, the current status has different degrees of impact to different future time intervals. So we train different models to different time intervals, as shown in Figure 5. Each blue broken arrow showed in Figure 5 denotes a temporal predictor. Over the next six hours, we train a model for each hour. A linear regression is employed to model the local temporal change of water level.

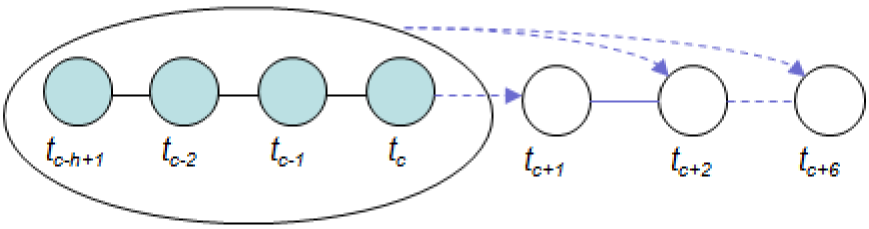

Figure 5. Illustration of the Temporal Predictor

(2) Spatial Predictor.

A station's water level is also depends on its upstream stations and rainfall at current. Rainfall is a direct factor for the water level, and if a station's upstream has a rising on water level, the station's water level will rise in future. To model the impact of rainfall and correlations with other stations, we devise a spatial predictor which predicts the water level of a location based on rainfall sensors' data and other water level stations' status, as shown in Figure 6.

In the implementation, we predict the deviation between the water level of current time t_c and the future time interval t_{c+w}, because the distribution space of deviation is much narrower than the original water level. Since every station's water level has a different range, to avoid large range value dominates the model, we need to rescale the range of water level to scale in [0, 1]. The formula is given in equation 1:

$$s' = \frac{s - \min(s)}{\max(s) - \min(s)} \quad (1)$$

The artificial neural network applied in the system is the widely-used Back-propagation (BP) neural network with one hidden layer in the experiments for its simplicity and generality. We set a sigmoid function φ(x) for the hidden and output layer. The formula is defined in equation 2:

$$\Delta S = \varphi\left(\sum_r w_r \varphi\left(\sum_p f_p w_{pq} + b_q\right) + b'_n\right) + b'' \quad (2)$$

Where f_p the feature scaling function for input; b'_n and b'' are the biases associated with neuron in different layers; w_{pq}, w'_{qr} and w_r denote the weight associated with the input of different layers.

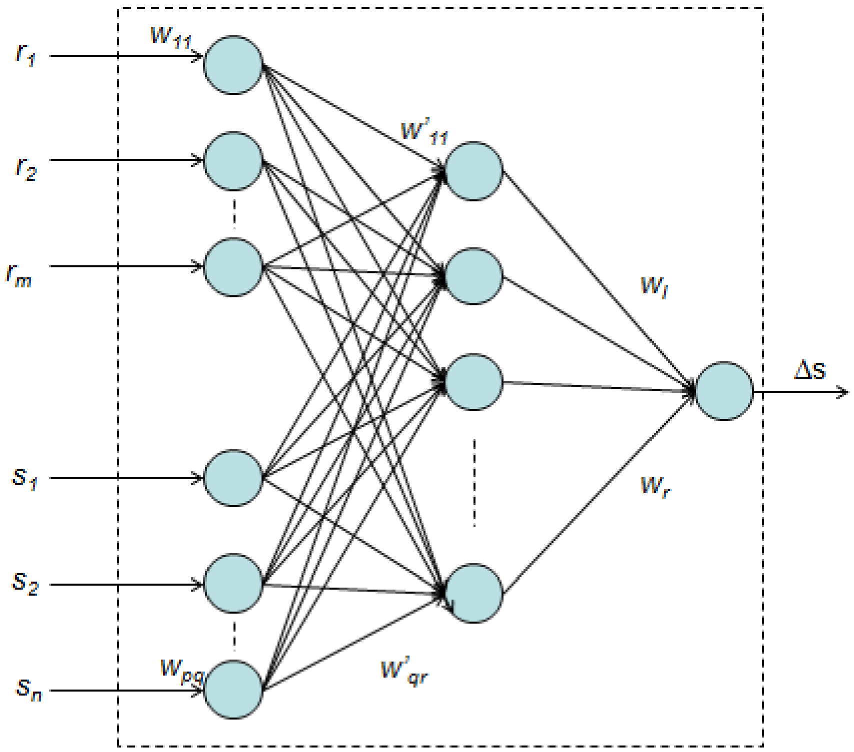

Figure 6. Illustration of the Spatial Predictor

3.5. Forecast and Warning

Using the predictive model and the real time data sent back by sensors, we can forecast the water level in the next 6 hours. Combine the forecast result with the warning threshold stored in database, we can easily compute whether there is a potential flood in the next 6 hours, thus give an early warning of flood to reduce the damage caused by flood. Figure 7 shows the prediction of our method at the next one hour against the ground truth of station 1101 in Hangzhou from March 3, 2012 to March 3, 2013.

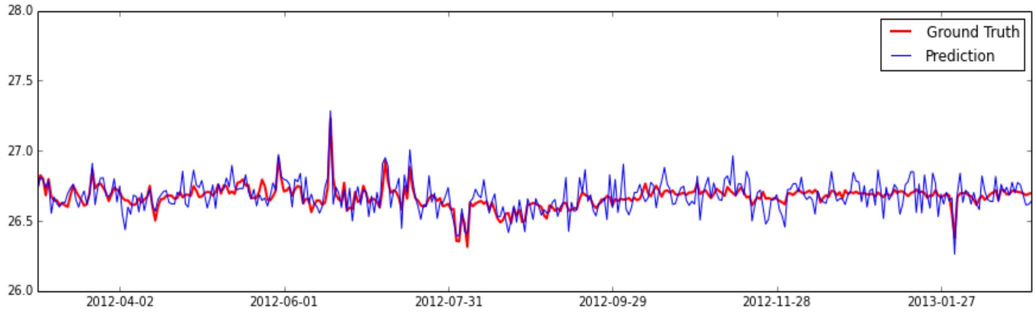

Figure 7. 1-hour River Stage Prediction of Station 1101 in Hangzhou

For the next 1-6 hours, we measure the prediction of each hour Yi against its ground truth Ti, calculating the RMSE and Efficiency Index (equation 3 and equation 4). As a result, we generate a measurement for the six time intervals respectively at each station.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(T_i - Y_i)^2} \qquad (3)$$

$$EI = 1 - \frac{\sum_{i=1}^{N}(T_i - Y_i)^2}{\sum_{i=1}^{N}(T_i - T_{avg})^2} \qquad (4)$$

Table 1 shows the corresponding performance statistical parameters for station 1101 in Hangzhou. It found that the statistical performance of the one hour prediction is better in term of EI. In terms of RMSE and EI, the shorter the forecasting time interval, the better the predictive model performs.

Table 1. Performance Statistical Parameters for Station 1101 in Hangzhou

	RMSE (m)	EI
t+1	0.0408	0.819
t+2	0.0513	0.773
t+3	0.0575	0.712
t+4	0.0629	0.658
t+5	0.0814	0.592
t+6	0.0873	0.536

4. Conclusions

We implemented a big-data-based urban flood defense decision support system as shown in the paper, which is a complex system using machine learning and data visualization methods. We focus on monitoring, analysis and mining of the water level and rainfall sensor data. We trained a model which can predict the water level in the next 6 hours, and give more information to the decision maker to defense the potential flood. In the future, we will continue to expand data sources and introduce more sophisticated model in the system, enhancing the applied value of flood defense in the digital urban management. Since our technology is general and only the system is only deployed and tested in Hangzhou, we will deploy the system in other cities to test its effectiveness.

Acknowledgements

This work is supported by the 2013 Construction Research Project of Zhejiang Provence, China (No. 18) and the Public Welfare Technology Project of Zhejiang Province, China (No. 2014C33060).

References

[1] Y. Zheng, L. Capra, O. Wolfson and H. Yang, “Urban Computing: Concepts, Methodologies, and Applications”, ACM Trans. Intelligent Systems and Technology, vol. 5, no. 3, **(2014)**, pp.28-55.
[2] Y. Zheng, F. Liu and H. P. Hsieh, “U-Air: When Urban Air Quality Inference Meets Big Data”, Proceedings of KDD, **(2013)**.
[3] Y. Zheng, T. Liu, Y. Wang, Y. Zhu and E. Chang, “Diagnosing New York City’s noises with ubiquitous data”, Proceedings of the 16th International Conference on Ubiquitous Computing, **(2014)**.
[4] J. Shang, Y. Zheng, W. Tong, E. Chang and Y. Yu, “Inferring Gas Consumption and Pollution Emission of Vehicles throughout City”, Proceeding of the KDD, **(2014)**.
[5] F. A. Pugliese, G. Montesano and G. Valenti, “Urban Traffic Analysis From a Large Scale Floating Car Data System”, WIT Transactions on the Built Environment, vol. 107, **(2009)**, pp. 3-11.
[6] Y. X. He, Y. J. Wang, T. Luo, A. Y. He and J. Wang, “Urban Residential Load Combined Forecast Model Based on Data Mining Techniques and Panel Data Theory”, Journal of Computational Information Systems, vol. 6, **(2010)**, pp.1801-1808.
[7] G. L. Chen, E. P. Wang, X. X. Sun and Y. Q. Tang, “An Intelligent Analysis and Mining System for Urban Lighting Information”, International Journal of Smart Home, vol. 9, no. 3, **(2015)**, pp.253-262.
[8] B. E. Pengel, V. V. Krzhizhanovskaya, N. B. Melnikova, G. S. Shirshov, A. R. Koelewijin, A. L. Pyayt and I. I. Mokhov, “Flood Early Warning System: Sensors and Internet”, Proceedings of the 5th International Conference on Flood Management, **(2011)**;Tokyo, Japan.
[9] L. H. Feng and J. Lu, “The practical research on flood forecasting based on artificial neural networks”, Expert Systems with Applications, vol. 37, **(2010)**, pp. 2974-2977.
[10] P. S. Yu, S. T. Chen and I. F. Chang, “Support vector regression for real time flood stage forecasting”, Journal of Hydrology, vol. 328, **(2006)**, pp. 704-716.
[11] R. Krzysztofowicz, “Bayesian system for probabilistic river stage forecasting”, Journal of Hydrology, vol. 268, **(2002)**, pp.16-40.
[12] Y. Chidthong, H. Tanaka and S. Supharatid, “Developing a hybrid multi-model for peak flood forecasting”, Hydrological Process, vol. 23, **(2009)**, pp.1725-1738.

Authors

Taimeng Yang
Born in 1989, postgraduate, yangtm2008@163.com.
His main research interests include data mining, big data processing and smart city.

Guanlin Chen
Born in 1978, Ph.D., associate professor, chenguanlin@zucc.edu.cn. His main research interests include computer networks, E-government and information security.

Xinxin Sun
Born in 1973, M.E, associate professor, sunxx@zjweu.edu.cn. Her main research interests include computer networks, software engineering.

