scientific reports

Check for updates

OPEN

From data to action in flood forecasting leveraging graph neural networks and digital twin visualization

Naghmeh Shafiee Roudbari[1,3]✉, Shubham Rajeev Punekar[1,3], Zachary Patterson[2], Ursula Eicker[2] & Charalambos Poullis[1]

Forecasting floods encompasses significant complexity due to the nonlinear nature of hydrological systems, which involve intricate interactions among precipitation, landscapes, river systems, and hydrological networks. Recent efforts in hydrology have aimed at predicting water flow, floods, and quality, yet most methodologies overlook the influence of adjacent areas and lack advanced visualization for water level assessment. Our contribution is two-fold: firstly, we introduce a graph neural network model (LocalFLoodNet) equipped with a graph learning module to capture the interconnections of water systems and the connectivity between stations to predict future water levels. Secondly, we develop a simulation prototype offering visual insights for decision-making in disaster prevention and policy-making. This prototype visualizes predicted water levels and facilitates data analysis using decades of historical information. Focusing on the Greater Montreal Area (GMA), particularly Terrebonne, Quebec, Canada, we apply LocalFLoodNet and prototype to demonstrate a comprehensive method for assessing flood impacts. By utilizing a digital twin of Terrebonne, our simulation tool allows users to interactively modify the landscape and simulate various flood scenarios, thereby providing valuable insights into preventive strategies. This research aims to enhance water level prediction and evaluation of preventive measures, setting a benchmark for similar applications across different geographic areas.

Since 2021, North America has been consistently experiencing devastating flood disasters that leave a trail of destruction in their wake. In a recent event, heavy rainfall and climate change-induced conditions triggered catastrophic flooding in British Columbia, Canada. It resulted in resident displacement, critical infrastructure damage, and a significant economic toll[1]. Communities were inundated, and the cleanup and recovery process were arduous and costly. Similarly, in 2021, the Mississippi River experienced severe flooding[2], impacting several states. This inundation led to the destruction of homes, businesses, and agricultural lands. It disrupted transportation and caused widespread economic losses. The catastrophic floods underscore the importance of flood forecasting and simulation. Communities require timely and accurate warnings to prepare, mitigate, and respond effectively to impending flood events. By simulating potential flood scenarios, we can assess the extent of damage and plan for effective flood control measures. Simulation of barriers along the river sides where inundation is frequent during flooding helps in decision-making for the best distribution of preventative resources in a timely manner. This pivotal task not only bolsters public safety but also underpins resilient urban planning, climate change adaptation, and scientific research, making it an essential component of modern-day disaster preparedness and response efforts.

One of the fundamental difficulties in flood forecasting arises from the intricate network of interconnected hydrological systems. Rivers, streams, tributaries, and drainage networks collectively form a dynamic, interdependent web. Predicting how water will traverse this network, especially in response to extreme weather events, requires sophisticated modelling. The cascading effect of changes in one part of the system can have far-reaching consequences downstream, adding layers of complexity to flood forecasting. Consequently, flood forecasting

[1]Immersive and Creative Technologies Lab, Department Computer Science and Software Engineering, Concordia University, Montreal, Canada. [2]Gina Cody School of Engineering and Computer Science, Concordia University, Montreal, Canada. [3]These authors contributed equally: Naghmeh Shafiee Roudbari and Shubham Rajeev Punekar. ✉email: naghmeh.shafiee@concordia.ca

goes beyond time series forecasting, evolving into a spatiotemporal problem. Graph Neural Networks (GNN) are designed to model complex relationships and spatial dependencies. For flood prediction, using a GNN-based model is necessary because it can accurately represent how hydrological systems are connected and changing over time. This gives us a better understanding of how water moves through water system networks. Their adaptability to changing conditions, reduced need for manual feature engineering, and capacity to integrate diverse data sources make them indispensable for improving prediction accuracy.

Another challenge is the spatial variability of flood events. Floods vary significantly across different regions and are not uniform. The local topography, land use, and geographical features significantly influence flood dynamics. Therefore, forecasting must account for these site-specific conditions, demanding meticulous and often site-specific modelling. This spatial variability can complicate predictions, as what holds true for one area may not apply to another, even within the same watershed. Our digital twin simulation addresses this challenge by creating a dynamic and detailed virtual replica of the real-world environment. Simulating flood scenarios within the digital twin makes it possible to account for these spatial variations and understand how they influence flood dynamics. Different scenarios can reveal the sensitivity of the system to specific variables. For instance, we assess how the placement of concrete barriers or the density of tree trunks influences water flow and how it impacts floods. This sensitivity analysis aids in identifying critical factors and their associated variability. Moreover, visual representations of potential flood scenarios are critical in engaging and educating local communities about flood risks and the importance of being prepared. This visual communication is instrumental in raising awareness and supporting flood mitigation efforts.

New research in hydrological prediction covers a wide range of topics. It helps us understand and predict various water-related events, like streamflow patterns[3], rainfall-runoff modeling[4], drought onset prediction[5], and flood forecasting[6]. However, many of these works often need to adequately consider spatial complexity, which is the cascading influence of water system components on each other. The omission of spatial variables, like localized rainfall variations, terrain differences, and land use changes, can limit the precision and reliability of predictions. Future research should bridge this gap by ensuring a more holistic understanding of hydrological processes to improve predictive capabilities.

In addressing the current challenges in flood forecasting, we have employed a cutting-edge spatiotemporal forecasting approach using graph neural networks. This technique, recently introduced in[7], outperforms traditional methods in predicting water flow. The TransGlow model uses Graph Convolution Recurrent Neural Network (GCRN) blocks to build an encoder-decoder structure. These blocks preserve sequence information and unravel spatial correlations. The transductive model structure is further enhanced with an augmented encoder layer employing attention mechanisms to focus on important elements of the input data. We adapt the TransGlow model to the specific context of Quebec, focusing on water level predictions in Terrebonne. We further compliment the flood forecasting framework with a flood-simulation prototype comprising Terrebonne's digital twin. Hence, our approach not only offers accurate flood forecasting but also improves our understanding of flood variability across different city zones and provides important insights for effective mitigation strategies. Our key contributions are as follows:

1. We present an extendable flood simulation and visualization technique, creating a detailed digital model of the city of Terrebonne, QC for flood simulation. (To ensure that the images shown in the paper are consistent, we focus on a 1.25 km^2 area encompassing Terrebonne, with a central focus on Ile St. Jean, adjacent to the Mille-Iles (Riviere Des) Au Barrage De Terrebonne weather station located at $45°41'35''N 73°38'58''$ W. This is a significant advancement, providing a valuable tool for understanding and predicting flood dynamics specific to this area. Notably, the proposed technique is designed to be reproducible and, more importantly, generalizable to other geographic locations, leveraging open-source data and standard practices to ensure its applicability beyond this initial area of focus. Our approach utilizes historical data and predictions from a graph neural network model to simulate changes in the water levels of nearby water bodies. Furthermore, our method supports interaction, including the simulation of concrete barriers for the purposeful redirection of river flow during flood events. By simulating and visualizing the controlled redirection of floodwaters, our approach facilitates the design of effective mitigation strategies tailored to diverse scenarios in a timely fashion.
2. We introduce a state-of-the-art GCRN-based method, originally developed for predicting hydrological parameters and now extended to local-scale flood prediction in Terrebonne, QC. Extensive experimentation has shown that this adaptation leads to better performance compared to current methods, emphasizing the effectiveness of the approach and its potential to improve flood prediction accuracy in specific local contexts.

In summary, our work contributes to (i) more precise flood forecasting using state-of-the-art deep learning frameworks, and (ii) evaluating mitigation strategies using a digital twin for simulation and visualization. By understanding localized flood patterns, stakeholders can make informed decisions to protect the community and infrastructure, ultimately reducing the impact of floods on the city. Furthermore, the research lays the foundation for similar applications in various geographical areas, extending the potential benefits beyond the city of Terrebonne, QC.

Related work

In this section, we offer a concise overview of related work in the fields of digital twin creation and forecasting methodologies.

Digital twins

Recent years have seen significant improvements in the use of geospatial data generated from LiDAR scans for landscape modeling[8]. Along with high-resolution topography data, other geospatial data sources, such as roads, building footprints, have been effectively used to create detailed and accurate representations of urban landscapes. The integration of deep learning techniques with photogrammetry and satellite imagery has revolutionized the field of 3D reconstruction. However, these works focus primarily on the visual acuity of the reconstructed models, rather than their physical aspects. For instance, 3D point clouds reconstructed with methods in photogrammetry such as structure-from-motion and multi-view stereo algorithms[9–11] or implicit 3D representations such as neural radiance fields[12,13], while highly useful in visualizing the city do not produce static meshes that can test for physics-based collision with fluids, as required by the problem statement at hand. To model urban features digitally which render with lower visual acuity but provide accurate physical interactions, techniques such as procedural extrusion of the building footprints[14] or deep learning based reconstruction[15] have been proposed to produce models with low level-of-detail.

The simulation of fluids, particularly for applications like flood prediction, has greatly benefited from advances in physics-based computational models. Research in this area has focused on the development of scalable and accurate simulation algorithms capable of handling the complex dynamics of water movement. Various physics-based fluid simulation techniques have been explored in literature, such as Eulerian methods[16], which track the fluid properties of fixed points in space, and Lagrangian methods[17], which track individual particles moving in space and time. Novel methods combining Eulerian and Lagrangian approaches[18] use particles to carry information about fluid motion and a Eulerian grid to solve for pressures and velocities. However, these methods are computationally expensive to be used on consumer hardware, which precludes them from being used in implementations to be accessed by stakeholders at ground zero during flooding, to offer insight for the decision making process. With this objective in mind, we refer to the work of Chentanez and Müller[19] which represents the fluid surface as a two dimensional heightfield grid, where each cell in the grid stores a height value indicating the height of the fluid surface at that point. The simulation works by iteratively updating the height values of the grid cells based on physical principles like conservation of mass and momentum, as well as external forces like gravity and wind. It offers a good balance between computational efficiency and visual fidelity, making it suitable for real-time applications as well as high-quality rendering. This approach is particularly suited for large-scale simulations like rivers where vertical detail—such as formation of waves and splashing of water—is less important than the overall motion and surface appearance.

Our area of research involves the integration of a detailed 3D city model with appropriate fluid simulation techniques to enhance flood prediction and management. Past attempts focused on the aesthetics of the water simulation[20], such as including foam, or simulating floods using amalgamated geometries of geospatial features[21]. In contrast, we propose a tool for immediate disaster planning and mitigation, focusing on next-day(s) predictions and actionable outcomes. We implement the simulation and visualization application using the Unreal Engine as the foundational framework, augmented with supporting libraries for the procedural generation of urban features such as roads and buildings, and the fluid height field simulation. This interdisciplinary approach allows for more accurate simulations of how water interacts with urban infrastructure, aiding in the development of more effective flood defense and urban planning strategies.

Forecasting

Time series forecasting is a fundamental task in various domains, from economics to meteorology, and it has witnessed a continuous evolution in methodology over the years. Traditional time series forecasting methods, particularly those rooted in statistical techniques, have historically assumed linearity within the data. These methods include Vector Auto Regressive models[22], Autoregressive Integrated Moving Average (ARIMA)[23], and exponential smoothing[24]. The assumption of linearity simplifies the forecasting process but is often at odds with the intricate, nonlinear patterns frequently present in real-world time series data. However, machine learning has shown remarkable aptitude in uncovering and modeling non-linearity within time series data, significantly enhancing forecasting accuracy across various domains. In their work,[25] introduced a wavelet-based model for stream flow prediction.[26] explored a Support Vector Machine (SVM) approach for water quality prediction, while[27] utilized a Multilayer Perceptron (MLP) and Radial Basis Function (RBF) model for the same task. Their findings demonstrated that these models exhibited strong predictive performance compared to conventional baseline methods. While machine learning methods show proficiency in learning non-linearity in time series data, the ever-increasing complexity of modern datasets necessitates exploring deep learning approaches.

Recurrent Neural Networks (RNNs) with internal memory[28] have emerged as a compelling choice for time-series forecasting. Specifically, models like Long Short-Term Memory (LSTM)[29] and Gated Recurrent Unit (GRU) models[30] have proven proficiency at mitigating issues such as vanishing gradients while effectively capturing long-term temporal dependencies. The integration of attention mechanisms[31] has extended the horizons of time series forecasting. Transformer-based methods have made notable progress in Time Series Forecasting (TSF) applications, spanning domains like hydrometric forecasting[32], air quality prediction[33], and energy forecasting[34]. These models have showcased their effectiveness in modeling complex temporal relationships.

In the context of spatiotemporal forecasting, the landscape of TSF has witnessed an evolution driven by innovative approaches and their intersection with the spatial domain. Initially, to capture spatial dependencies within time series data, Convolutional Neural Networks (CNNs) emerged as a solution, treating TSF data as a time-space matrix[35]. However, CNN models exhibit limitations in their applicability, particularly when dealing with non-grid-like structures and the representation of data in Euclidean space.

Subsequently, the spotlight turned to GNNs, which offered a robust framework for expressing intricate relationships within unstructured data via a graph-based data structure. While graph-based methods found extensive

use in various spatiotemporal applications, including renewable energies[36], traffic[37], and electricity forecasting[38], their utilization in predicting hydrological-related parameters and water resources remained relatively restricted.

Recent studies have bridged this gap by combining GNNs with Recurrent Neural Networks (RNNs), exploring the interplay of spatial and temporal changes[39] and achieving promising results. The advent of Encoder–Decoder architectures, renowned for their efficacy in processing sequential data[40], has further propelled innovations in TSF. Researchers have adapted this architecture to TSF, employing models such as GCRN[41], attention-based mechanisms[42], and Transformer-based architectures[43].

In this context, we leverage a novel transduction architecture[7] that uses the attention mechanism to enhance the hidden state, facilitating improved information flow and context preservation. This method harnesses the strengths of RNN models in capturing temporal information and GNNs in unveiling spatial relationships. To support the Graph Convolutional operation, LocalFLoodNet uses a self-learning graph module that can autonomously discover the implicit connections within the data. This is invaluable since sometimes the actual network structure between different water stations is not known in advance or evolves over time.

Problem formulation

The input to the prediction network can be represented as a two-dimensional matrix, denoted as X. This matrix serves as the fundamental input data structure to capture the spatiotemporal dynamics of water levels, where each row in the matrix represents a different water station, and each column corresponds to a specific timestamp covering T time steps (from $t - T + 1$ to t). Thus, each element of the matrix, represented as $X[i][j]$, signifies the water level at a certain water station (indexed by i) for a particular day (since our data frequency is daily) (indexed by j), allowing the prediction network to learn and model the complex relationships and patterns associated with water levels at different stations over time.

$$X = \begin{bmatrix} x_1^{t-T+1} & \dots & x_1^{t-1} & x_1^t \\ x_2^{t-T+1} & \dots & x_2^{t-1} & x_2^t \\ \dots & \dots & \dots & \dots \\ x_n^{t-T+1} & \dots & x_n^{t-1} & x_n^t \end{bmatrix}$$

The objective is to find a function f that captures the connections between water systems. The function f maps the spatiotemporal matrix X to a connectivity matrix A, representing the relationships between different water stations, $A = f(X)$. Given the connectivity matrix A, the current spatiotemporal matrix X, and a specified number of future time steps h, the goal is to find a function g that predicts the spatiotemporal matrix X^h for the future h time steps. Hence, function g takes the current connectivity matrix, the current spatiotemporal matrix, and the number of future time steps as inputs and produces the predicted spatiotemporal matrix for the future, i.e. $X^h = g(A, X, h)$.

The overall objective is to find functions f and g that enable the network to understand the connections between water systems from the current spatiotemporal data and use this understanding to forecast water levels at multiple stations for future time steps.

Methodology

This section details the theoretical approach utilized in this work for predicting the water level, and is illustrated in Fig. 1. An extended explanation can be found in the Supplementary Material, Section "Graph Neural Networks".

Architectural enhancement: attention-augmented encoder–decoder

The encoder–decoder design, inspired by RNNs and their variations, addresses the challenges of capturing and processing sequential information in a wide array of applications, from natural language processing, where it excels in language translation tasks[44], to time series forecasting, where it adeptly handles the intricacies of temporal patterns[45]. At its core, the encoder–decoder structure is a two-step process strategically tailored to handle

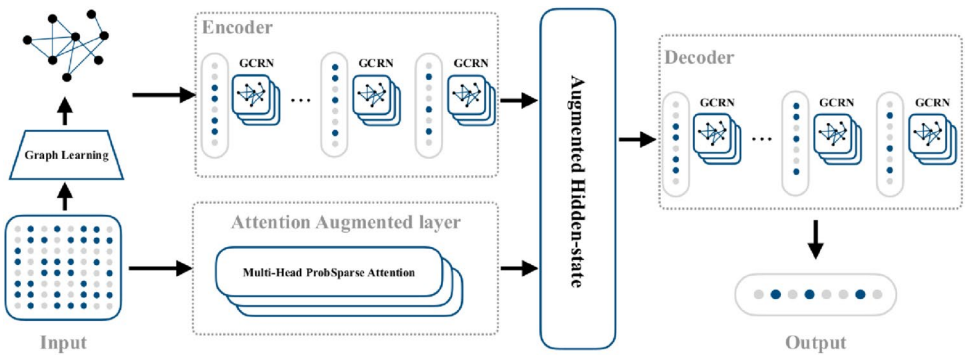

Figure 1. Network architecture. The input is a spatiotemporal matrix. LocalFLoodNet learns the graph and predicts using a GCRN-based encoder–decoder; an attention layer is added to augment the decoder input.

sequences of variable lengths. The encoder, functioning as the initial stage, systematically processes the input sequence, distilling its salient features into a fixed-size representation known as the context vector.

This condensed context becomes a comprehensive summary of the input, capturing its essential characteristics. The vanilla encoder-decoder architecture processes an input sequence $X = (x_1, x_2, ..., x_T)$ with an encoder to create a context vector. The encoder processes each element of the input sequence sequentially, producing hidden states h_t for each time step t, i.e., $h_t = Encoder(x_t, h_{t-1}), C = h_T$ where C is the context vector and the function *Encoder* can be any recurrent or non-recurrent layer, such as an LSTM or GRU. In this paper, the foundational component is GCRN. It captures the information from the input sequence and creates a hidden state for each time step. Then, the decoder generates an output sequence using this context vector. For each time step in the output sequence, the decoder generates an output y_t and updates its hidden state s_t, i.e., $s_t = Decoder(y_{t-1}, s_{t-1}), s_0 = C$ where the function *Decoder* is similar to the encoder's function but typically has a different set of parameters.

In order to tackle challenges related to information loss and compression, LocalFLoodNet refines the traditional encoder-decoder architecture by introducing an augmented attention layer. This supplementary attention layer dynamically focuses on elements within the input sequence and calculates an attention vector that emphasizes the significance of different segments of the input sequence. The resultant attention vector *Attention* is subsequently merged with the final hidden state obtained from the encoder, $C = Concat[h_T, Attention]$. This fusion produces an enriched context vector tailored for the decoder. By incorporating this augmented attention mechanism, the model adapts more flexibly to varying degrees of importance within the input sequence. This modification enhances the model's capacity to reduce information loss and improves its ability to generate contextually rich output sequences in sequence-to-sequence tasks.

The attention mechanism plays an important role in sequence-to-sequence tasks by allowing the model to focus on different parts of the input sequence when generating each element of the output sequence. However, the standard attention mechanism has a quadratic computational complexity in terms of sequence length. This is primarily because, for each element in the output sequence, the model computes an attention score for every component of the input sequence. LocalFLoodNet adopts ProbSparse attention, as proposed by[46]. In this version of attention, a subset of K queries is selectively chosen based on specific measure M:

$$\hat{Q} = M(Q, K)$$

$$Attention(Q, K, V) = Softmax\left(\frac{\hat{Q}K^T}{\sqrt{d}}\right)V,$$

where Q,K, and V denote query, key, and value, respectively, and d is the input dimension. The probability distribution M decides the relevance of each token in the sequence relative to the current token. Tokens with higher importance have a higher chance of being incorporated into the sparse query matrix, while tokens with lower significance have a lower probability of inclusion. This approach ensures that the attention mechanism selectively focuses on relevant tokens, contributing to the overall efficiency of the LocalFLoodNet.

Experiments

The experiments in this work focus on Terrebonne, Quebec, Canada, situated in the North Shore region of the Greater Montreal Area. Terrebonne's climate features warm summers and cold winters with variable weather, including heavy rainfall and rapid snow melt periods. While the research is rooted in Terrebonne, its proposed approach transcends geographic constraints, offering generalization potential for application in other cities. We leverage daily water level data from 8 monitoring stations situated within the Terrebonne vicinity, across rivers, streams, and lakes, enhancing the model understanding of interconnected water bodies. Sourced from Environment and Climate Change Canada[47] spanning 21 years (2000–2021), the spatial distribution of the monitoring stations and water level fluctuations is visualized in Fig. 2a,b demonstrates temporal variability of Terrebonne station positioned at the geographic coordinates of (45.69306, − 73.64944) for the year 2016. To address missing values, the Historical Average is employed, supplemented by Gaussian smoothing, to remove anomalies. This filtering technique mitigates sudden data fluctuations without compromising underlying patterns. By enhancing data reliability, LocalFLoodNet is better equipped to derive meaningful insights from the water level data.

For a fair comparison, we use the same training, validation, and test sets for all the methods and allocate 70% of the data (2000–2015) for training, 10% (2015–2017) for model validation, and the remaining 20% (2017–2021) for final testing. To verify the effectiveness of the LocalFLoodNet, we ran the experiments on six baseline and state-of-the-art models explained below. Table 1 shows the experiment results for different examined periods over three evaluation metrics, including mean absolute error (MAE), mean absolute percentage error (MAPE) and root mean square error (RMSE). The smaller these errors, the higher the prediction accuracy (Fig. 2).

We compare a number of baseline and state-of-the-art methods for spatiotemporal forecasting. Baseline methods, such as the Historical Average (HA) method, provide simplistic yet valuable benchmarks by predicting future values based on historical averages, serving as a foundational reference point for evaluating more complex techniques. Meanwhile, state-of-the-art methods represent the forefront of spatiotemporal forecasting research, incorporating advanced algorithms and architectures to capture intricate spatial and temporal dependencies within the data. These methods include the Informer model[46], which is designed for time series prediction, integrating a novel Transformer-based architecture with a multi-level feature fusion mechanism to efficiently capture long-term dependencies and spatial-temporal patterns, thereby improving forecasting accuracy in diverse time series datasets, GTS method[49] which is highlighted for its scalability, making it well-suited for large datasets, especially in traffic forecasting, utilizing the GCRN block structure and a graph learning approach, Dynamic Causal Graph Convolutional Network (DCGCN)[50] which uses graph convolutional networks based on generated stepwise dynamic causal graphs to make predictions and Spatial-Temporal Attention Wavenet (STAWnet)

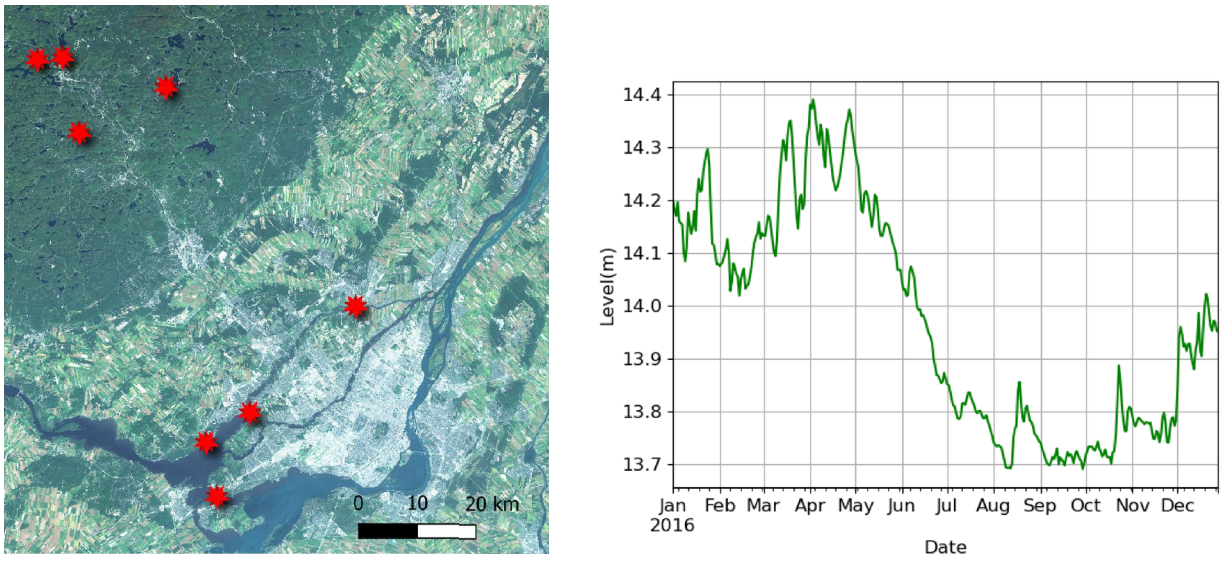

Figure 2. (**a**) Location of the stations. Map data from Ressources naturelles du Québec[48] visualized in QGIS v.3.34.0 (**b**) Water level variation at Terrebonne station throughout 2016.

		Horizon		
Method	Metric	3 days	6 days	9 days
HA	MAE	0.28	0.28	0.28
	MAPE	9.56	9.56	9.56
	RMSE	0.42	0.42	0.42
Informer	MAE	0.35	0.35	0.35
	MAPE	0.10	0.10	0.10
	RMSE	0.53	0.53	0.53
GTS	MAE	0.34	0.33	0.33
	MAPE	0.12	0.13	0.14
	RMSE	0.50	0.49	0.49
DCGCN	MAE	0.16	0.25	0.51
	MAPE	0.06	0.11	0.83
	RMSE	0.22	0.32	1.81
STAWnet	MAE	0.08	0.12	0.15
	MAPE	0.03	0.04	0.04
	RMSE	0.12	0.18	0.23
LocalFLoodNet	MAE	0.05	0.09	0.13
	MAPE	0.01	0.02	0.03
	RMSE	0.09	0.16	0.21

Table 1. Experimental results of baseline methods conducted on water flow data for different future time horizons e.g. 3, 6, 9 days. The smaller the errors, the higher the prediction accuracy.

method[51] which uses temporal convolution to handle long-time sequences and self-attention networks to capture dynamic spatial dependencies between different nodes.

In analyzing the performance comparison based on the table of results, several trends emerge across different spatiotemporal forecasting models. The HA method does not take into account any trends, seasonality, or other patterns in the data, making it less accurate compared to the rest of the methods. Informer and GTS exhibit similar performance levels, while DCGCN surpasses both methods in shorter-term predictions. However, its performance diminishes when forecasting over longer periods, indicating the advantages of correlation-based spatial dependencies over causality-based ones in this context. STAWnet consistently demonstrates strong performance across all evaluated metrics, showcasing the effectiveness of its attention mechanism, particularly for longer forecasting horizons. Notably, LocalFLoodNet, which combines graph convolution with an RNN-based

encoder-decoder and augmented attention layer, achieves the most promising results among the analyzed models. This underscores its capability to effectively capture complex spatiotemporal patterns and surpass other models in this comparative assessment.

Simulation and visualization

We have developed an immersive, interactive application utilizing Unreal Engine to simulate and visualize fluvial floods (river floods) and pluvial or flash floods on a digital representation of the desired geographical region.

We selected Unreal Engine[52] as our simulation and visualization platform due to its robust physics engine, extensive support for asset creation, and comprehensive fluid simulation libraries. Additionally, it has the ability to execute complex simulations on consumer-grade hardware. The optimization capabilities of Unreal Engine, such as the conversion of dynamic elements into static meshes to minimize draw calls, significantly improve rendering performance. This feature is particularly beneficial for rendering large-scale, detailed urban environments. Moreover, Unreal Engine's flexibility facilitates the seamless integration of new data, ensuring that our digital twins accurately mirror the changing urban landscapes and support swift development cycles.

Another point that merits consideration is the purpose which is to be served by the reconstructed digital twin of an urban landscape. Only certain features of an urban city digital twin pertain to accurate flood simulation and visualization and the decision-making strategies for mitigating the adverse effects, namely the buildings, roads, and urban tree canopy. We distill these features and focus our efforts on reconstructing a representative digital twin of the city, which is qualified in scope for the purpose of flood simulation. We implement a procedural approach to reconstructing buildings, roads, and trees in the urban landscape, leveraging the procedural tools provided in Unreal Engine by utilizing curated geospatial data.

The application facilitates various forms of interaction based on the specific viewing perspective, aerial and ground. The rise in the level of the water bodies can be specified to simulate flooding to varying extents. Our application incorporates functionality to add or remove obstructions flexibly at custom scales and orientations, enabling users to manipulate the water flow and providing valuable insight for flood mitigation strategies. Furthermore, we introduce the capability to simulate inland flooding by enabling users to specify water sources within the urban terrain. Our simulation and visualization technique offers a comprehensive platform for understanding and addressing flood scenarios, encompassing both natural and human-made factors for effective decision-making in flood risk management.

Digital twin modeling

In subsequent sections, we provide a detailed account of the methodology employed in the development of digital twins, beginning with the establishment of a digital twin for Terrebonne. This initial digital twin lays the groundwork for subsequent analysis, simulations, and decision-making processes. Our development process commences with the collection and integration of geospatial data, encompassing a diverse array of spatial information such as topography, environmental elements, and infrastructure. Section "Data-source Pre-visualizations" in the Supplementary Material depicts renders of the data used. Our primary objective is to formulate a method that is efficient and adaptable to different geographical contexts with minimal adjustments, thereby affirming the generalizability, scalability, and broader application of our methods. Furthermore, to ensure the robustness and generalizability of our approach, we give precedence to utilizing data from open repositories. These sources are publicly available and typically conform to established standards and guidelines, facilitating broader accessibility and compatibility.

After data preprocessing in QGIS [an open-source Geographic Information System (GIS)], we import it into Unreal Engine through a landscaping plugin tool. This transition marked a pivotal phase in our methodology, allowing for the integration of geospatial data into a dynamic, interactive 3D modeling environment. The use of Unreal Engine enables the realistic and detailed reconstruction of urban landscapes, significantly enhancing our ability to simulate and analyze the potential impacts of changing water levels within these environments. Figure S2f shows renders of the various stages during the reconstruction of the digital twin in Unreal Engine.

Modeling the terrain

For the specific task of acquiring high-quality elevation data, we utilize a High-Resolution Digital Elevation Surface Model (HRDEM), which provides detailed topographical information. The HRDEM data utilized in this work is obtained from the Open Data Portal of Canada[53]. The HRDEM dataset is a comprehensive source of terrain data that encompasses a variety of models, including both DSM (Digital Surface Model) and DTM (Digital Terrain Model) among other terrain-related data types. The DSM is distinct in its approach to elevation data, as it captures elevations that account for both natural and artificial features on the earth's surface, such as buildings, vegetation, and infrastructure. Conversely, the DTM focuses exclusively on the ground surface, offering elevation data that considers only the natural undulations and contours of the terrain, absent of any buildings, vegetation, or man-made structures. This delineation between the DSM and DTM models within the HRDEM dataset is critical, as we require a terrain model that isolates the natural ground surface to compute the levels of flooding accurately. The HRDEM uses the Universal Transverse Mercator (UTM) coordinate system based on the North American Datum of 1983 (NAD83), specifically its Canadian Spatial Reference System (CSRS) variant. Each 1-meter resolution dataset encompasses a 10×10 km^2 area, making it highly focused and detailed for localized studies. To ensure high-precision geo-referencing, all datasets are projected into the EPSG:2959 coordinate system. EPSG:2959 is a specific implementation of the UTM system optimized for the region, providing an additional layer of accuracy. By standardizing the coordinate system, it simplifies data integration and sharing among different GIS applications. Figure S1g shows a render of the Digital Terrain Model (DTM) from the HRDEM dataset for our test area.

Following the preprocessing of the DTM to isolate the target area, we integrated the processed data into Unreal Engine to construct a landscape, utilizing the elevation data as a heightmap. To achieve a realistic terrain representation, we employed a sophisticated material strategy that leverages geospatial land use data for the informed distribution of various terrain materials. This approach enables the accurate depiction of different land covers: grass in open spaces and parks, mud in areas designated for buildings, and a vegetation template for tree-rich zones, including urban canopies and forests. Figure S2a illustrates the visual accuracy achieved by aligning land use data with the precise application of materials based on terrain characteristics derived from elevation data, establishing a foundation for subsequent fluid simulation tasks.

Modeling urban infrastructure

Following the completion of the surface terrain modeling, we focused on the modeling of urban infrastructure elements, namely buildings and roads.

Buildings

For the integration of building-specific information, we obtained multi-polygonal geospatial vector data from the municipal urban planning authority[54]. This data, characterized by high spatial accuracy, delineated the geographical extents of all man-made structures within the jurisdiction of the city. Figure S1b shows a render of this multi-polygon vector data of building footprints in QGIS. Utilizing this vector data, static 3D representations of buildings were incorporated into the digital twin environment. To generate the buildings within the digital twin of the city, the data was imported into Unreal Engine via a landscaping plugin and a standard procedural modeling algorithm was employed by extending a wall mesh along the spline of the building footprint to construct the exterior facets of these buildings. This resulted in topologically closed, watertight 3D models that are representative of their actual physical counterparts in terms of water collision that will be tested for flood simulation. Given our primary focus on flood simulation scenarios, the vertical elevation or rooftop height of individual buildings was standardized, as it did not influence the hydrodynamic computations. This procedural generation algorithm ensured that the buildings adhered to collision boundaries essential for water simulation, allowing for a meaningful output that accurately reflected the bounds of the real structures.

After the reconstruction of the 3D models of the buildings themselves, a notable challenge we encountered involved the accurate placement of these structures on uneven terrain. This issue was critical, as buildings needed to be positioned in a manner that ensured water would correctly interact with their walls during simulations. Specifically, it was imperative to avoid scenarios where water could seep through gaps resulting from the terrain's irregularities beneath the buildings. To address this, we programmatically adjusted the buildings' placement to ensure that their walls extended sufficiently into the ground, thereby eliminating any potential spaces through which water could erroneously pass. This alignment was essential for achieving realistic simulation results, particularly in flood scenarios. Figure S3 shows the renders before and after aligning the building structures on the landscape, in perspective and orthographic views. The line traces shown in the images show the guiding points used to place the buildings. The line traces are red to indicate no intersection with the landscape underneath, and they are green after computing the points of intersection. It should be noted that the figure shows a sub-sampled region of the map for the purpose of illustrating the method of aligning procedurally generated buildings with the map. Figure S2b shows the landscape with building structures after their correct placement from a high viewpoint.

Roads

For road infrastructure, multi-line string geospatial vector data was also acquired from the city's database[54]. Preliminary steps involved the conversion of these multi-line strings to geospline entities. These were imported into Unreal Engine using the landscaping plugin as georeferenced splines. Subsequent adjustments were made to the splines' elevational attributes to ensure they were co-planar with the pre-modeled terrain. A procedural algorithm was then applied to transform these geosplines into three-dimensional road models with proper texture and material attributes, by procedurally applying a road section mesh constructed to the city proportions along these splines to reconstruct a representative digital model of the roads. Note that appropriate manual adjustments were made in cases of roads placed for bridges. Figure S1c shows a render of the vector geospatial data in QGIS and S2c shows the landscape with roads in the digital twin model.

Modeling environmental elements

We dedicated the final step to the modelling of environmental components, specifically the tree canopy and bodies of water.

Trees

The data utilized for canopy mapping was obtained from the Donnees Quebec database[55]. In this data, the canopy is demarcated by the ground-level projection of arboreal crowns, encompassing leaves, branches, and trunks, and is observable via aerial imagery. All vegetation exceeding a height of 2 m was included in the data. The canopy mapping was executed through advanced deep learning techniques, leveraging variables calculated at a 1-m spatial resolution from raw airborne LiDAR data gathered over the period 2010–2020. Figure S1f shows a render of the vector data representing the tree canopy for our test area. Since the data for tree canopy is represented using multi-polygon geometry and we do not have the locations of the tree trunks themselves, a procedural foliage spawning method was used to create an digital modeling of the urban vegetation. The trees were placed at randomly sampled points to ensure they cover the regions of canopy retrieved from the canopy dataset with appropriate density. We ensured that in modeling the trees, the resistance offered by the trees to the flood water was accurate rather than achieving faithful physical placement of trees with respect to the their locations in the real world. Figure S2d shows a render of the placed foliage in the digital twin reconstructed in Unreal Engine.

Modeling water bodies

This flood simulation model uses geospatial data sourced from Geofabrik[56]. The dataset comprises two main types of vector geospatial information: water bounds represented by polygons and waterways represented by strings. The waterways form the basis for generating waterbodies in the simulation. The model integrates historical or predicted data to dynamically simulate water volume by specifying the height of water bodies. The simulation aims to accurately represent the behavior of water systems over time, providing a valuable tool for applications such as water resource management and flood prediction. Figure S1e shows a render of the two types of vector data representing the water for our test area.

To model the water bodies accurately in our project, we utilized waterways geospatial vector data represented as multiline strings. In our heightfield fluid simulation (Supplementary Material, Section "Heightfield Fluid Simulation"), we introduce a collection of water sources in the shape of customizable elliptical cylinders to represent a volume of water within each water body (river/stream). This cylinder's dimensions are adjustable along its major and minor axes, allowing for the precise modeling of water volumes, and the height is computed from the specified water level, thus closely mirroring the actual water volume present in the geospatially defined water bodies.

To ensure that the water bodies have the correct volume of water, we have implemented the following series of steps, which produce a close approximation of the actual water volume presented in the water bodies, given the water level.

1. We first augmented the geospatial multiline string vector data for waterways by adding any unrepresented streams, identified visually by dashed lines in Fig. S1 showing a QGIS render.
2. We then imported the geospatial multiline string vectors into Unreal Engine with the landscaping plugin so that they are georeferenced in the same coordinate system as the landscape (and the other features as well), the corresponding objects in Unreal Engine are splines. Figure S5a shows the splines representing water bodies loaded up in Unreal Engine.
3. Each point in a spline has location coordinates and a vector representing the normal to the direction along the spline at the point. For each spline point in the water body (rivers/streams) splines, we computed the intersection of two line traces projected along the normals to the spline at the point in clock-wise and anti-clockwise directions at a configurable height above the river bed, which ensures that our model approximates the physical dimensions of the actual water in the river/stream. Figure S5b shows the line traces in red, and the points of intersections of these traces with the landscape along the contours of the river, marked in red as well.
4. For every point in the spline, we used the length of the spline segment from the current point to the next point halved and the distance between the line trace intersections to compute the scale of the elliptical surface of the water and placed it at the midpoint of the line trace intersections, orienting it correctly using the direction along the spline at the point. The height of each water cylinder is computed so that the actual water surface is aligned with the specified water level. Figure S5c and S5d show the spawning of the cylindrical water sources and the eventual surface of the water body itself.

For inland water flooding, we have implemented the ability for the user to specify cylindrical sources of water, with configurable height of the water cylinder to simulate flash floods due to rains.

Comparative visualization

Figure S6 presents a comparative analysis of our Unreal Engine reconstructed digital twin of an urban environment against the representation provided by Google Earth. While the Google Earth digital twin does present a more aesthetically pleasing representation due to the use of satellite images for textures, as seen from figures S6a and S6b, it does reveal a limitation in utility of Google Earth's approach, notably its reliance on low-polygon mesh approximations for urban features. This simplification compromises the accuracy necessary for flood simulation scenarios, as exemplified by Fig. S6d, which highlight Google Earth's inadequacy in measuring the depth of water bodies with the necessary precision for hydrological analysis.

In contrast, Figs. S6c,e illustrate the enhanced detail and dynamic water body simulation of our digital twin model within Unreal Engine. This model includes accurate terrain modeling and comprehensive geospatial elements like roads, buildings, and canopies that influence water flow, showcasing our model's superior fidelity and precision. The detailed urban features and accurately simulated water dynamics of our digital twin are essential for conducting dependable flood scenario simulations. This precision enables more accurate impact assessments and supports the formulation of effective mitigation strategies.

Viewing and interaction

Viewing perspectives

There are two distinct viewing perspectives for observation: the aerial perspective, commonly referred to as the birds-eye view, and the terrestrial perspective, commonly referred to as the ground-view. Utilizing an aerial perspective, the camera is positioned at a significant altitude, affording a comprehensive vantage point capable of surveying up to an area of 1.25 km^2. The camera elevation can be flexibly controlled within the altitude limits of 35 m up to 500 m (with reference to the sea level). From this perspective, given the aim of comprehending the impact of the flood on a broader region, the digital twin encompasses all intricate geospatial entities such as vegetation, buildings, roads, and so forth, as described in the previous section.

In contrast, the ground-view perspective employs a third-person camera that tracks a user-controlled character positioned at the point of interest on the ground. From this perspective, the primary aim is to examine the impact of water flow at a lower geographical scale. This view allows the user to gauge the extent of flooding in comparison to the scale of urban landmarks, such as the roads and buildings in the vicinity. To provide real-world

context, both viewing modes feature an onscreen overlay displaying the user location in WGS84 reference system coordinates and altitude in meters, thereby enabling users to accurately reference their position in the real world. The aerial and ground views are depicted in Fig. S7a,b, respectively. Note that in Fig. S7a, a green marker shows the location of the first person character that will be possessed on switching to ground view.

Interaction modes
Users can toggle between this navigation mode and an interaction mode, which allows them to navigate freely in either aerial or ground views or to add obstructions or water sources and specify the rise of water levels in various bodies. In this interaction mode, users have the ability to engage with the terrain by strategically positioning concrete water barriers in order to redirect the movement of water. Users also have the ability to add sources of water inland to simulate flash flood conditions. In Fig. 3a shows how barriers and inland sources of water can be set up, and Fig. 3b shows the simulation.

Overview map
We have implemented an orthographic projection of the map without the tree canopy, which allows real-time tracking of the water level as the simulation occurs. Figure S7c shows the overview map, which resembles a floodplain map.

Simulation experiments

This section outlines the experiments conducted using our simulation and visualization application, showcasing its qualitative performance by thoroughly exploring a range of scenarios in a detailed, comprehensive manner.

Simulation of the rise in water level
In our simulation, we modeled a scenario where the water level was raised by 200cm from the riverbed's baseline. Figure 4 illustrates the evolution of the water simulation across three different perspectives (aerial, ground, and overview) captured at successive moments in time.

Simulations with varying rises in water levels
In this experiment, we systematically investigated the impact of different rises in water levels on our simulation's performance and outcomes. We initiated the simulation with a water level set at 100 cm above the riverbed's baseline and incrementally increased this level by 25 cm for each subsequent experiment, continuing up to a maximum of 300 cm. This chosen range, from 100 to 300 cm, was designed to span a spectrum of scenarios: the lower threshold of 100 cm closely approximates conditions of regular river flow, where water bodies do not flood (refer to Fig. S8a), while the upper limit of 300 cm is representative of extreme flooding situations (refer to Fig. S8i). Through this approach, we aimed to comprehensively test the simulation's behavior under varying degrees of water elevation, from normal to critical flooding conditions. The overview map was utilized to illustrate the floodplain regions for each scenario, providing a visual representation of the extent and impact of flooding across the different water level simulations. (refer to Fig. S8)

Simulation with obstructions
We ran the simulation for a water level of 200 cm (above the base level in the river bed), and tested two scenarios, viewed from both the perspectives and monitored in the overview map. In the first scenario, we ran the simulation without creating any barriers obstructing the flooding water flow, resulting in puddling of water over the streets and close to the houses, as exhibited in figures S9a, S9c and S9e. In the second scenario, barriers were added to the map by carefully considering the expected flow of the water as observed from the previous scenario. Figure S9b,d,f show the migitated circumstances, thereby demonstrating the utility of this platform in providing insight into the mitigation strategy design for floods.

(a) Interaction mode in the application allowing users to add in-land water sources and obstructive barriers prior to simulation.

(b) Simulation triggered by the user, illustrating water flow based on placed sources and collision with obstructive barriers.

Figure 3. Interaction mode: adding obstructions and inland sources of water for simulation.

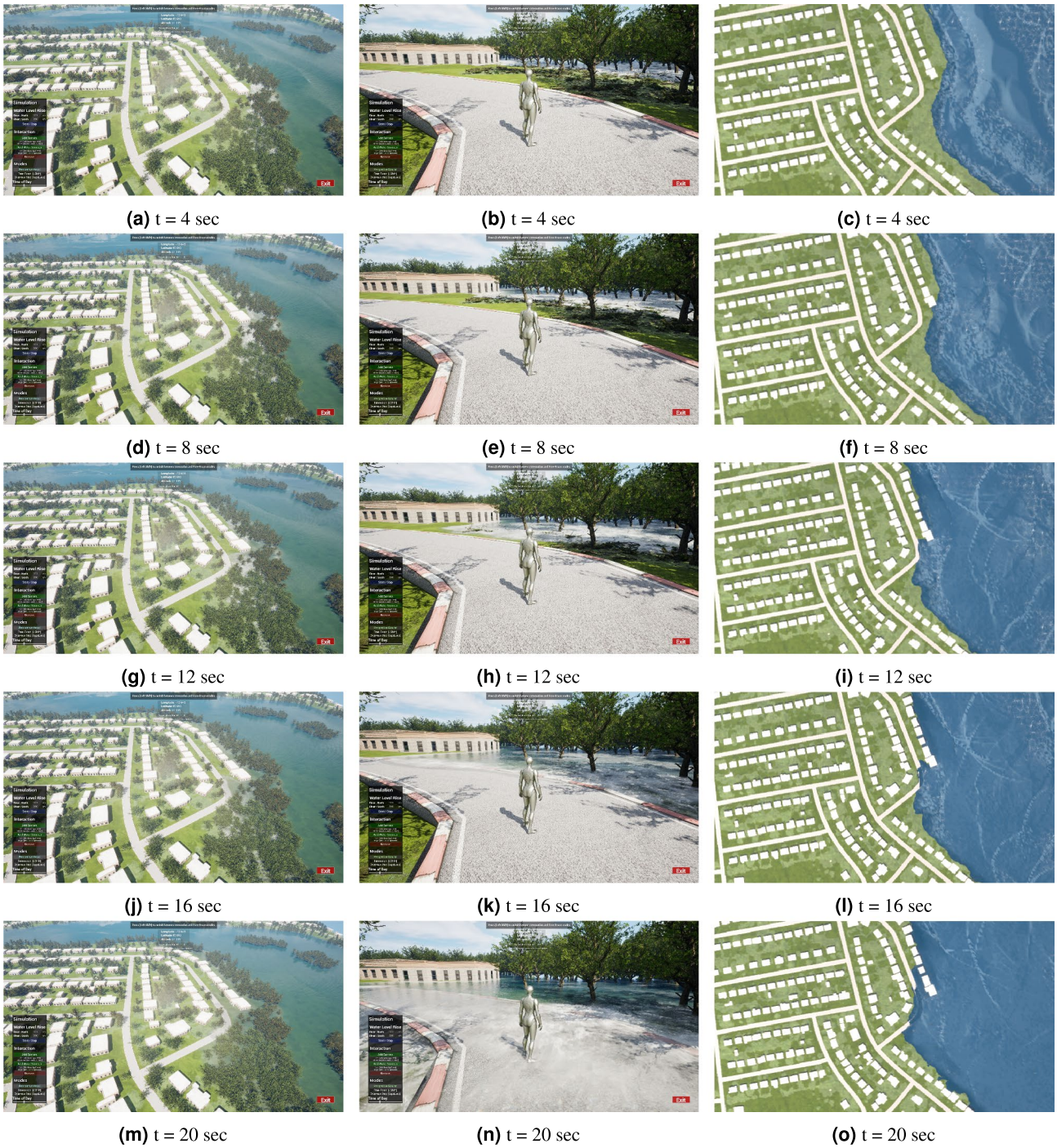

Figure 4. Simulation of water starting at level = 200 cm in river. Three columns show different perspectives of the simulation as it develops over time. (Column1: Aerial mode, Column2: Ground mode, Column3: Overview map). The simulation time is sped up for convenience to reach a stable state (no water motion) in reasonable time.

Simulation with real-world data

Historical and predicted data

There are two primary data sources for weather analysis: historical data and predictive data. Historical data refers to data that has been recorded over a period of time using sensors installed at the weather station. On the other hand, the predictive data is derived from utilizing the model outlined in Section "Methodology" to conduct weather forecasts for a specified time frame, such as the subsequent day, the following three days, and so forth. Based on the aforementioned factors, the application utilizes simulations to replicate the projected water levels, allowing users to analyze potential flood events and make informed decisions.

In this simulation experiment, using the historical dataset, we retrieved records and accessed the water level data for well-documented spring flooding events in 2017. Our model predicted the anticipated water levels by choosing the same time period. Figure 5a presents a demarcation map of the floodplain that occurred during the spring floods of 2017 and 2019 (provided by the government of Quebec[57]). Figure 5b illustrates the simulated water levels based on observed data, while Fig. 5c displays the simulated water levels using predicted data. Both figures represent the same time period and geographical area, providing a comparative analysis between the observed and predicted water level scenarios.

Our flood simulation and visualization platform performs by considering terrain variations and incorporating all modeled urban features. This comprehensive approach ensures that simulations of water inundation generated from both historical Fig. 5b and predictive Fig. 5c data offer precise depictions of floodplain extents on par with the demarcated floodplain Fig. 5a. Moreover, the platform's capability to simulate scenarios with obstructions, in conjunction with predicted water level rises, provides a robust framework for testing and implementing effective flood mitigation strategies.

Limitations and scope for future works

Our research has laid a solid foundation for predicting water levels with spatiotemporal analysis of historical data and simulating and visualizing urban flooding scenarios. There remains potential for future enhancements. Currently, our simulation does not account for stream flow dynamics; integrating it could capture the variability in water velocity in flooding regions, allowing for more precise temporal modeling. Another aspect not yet modeled is the city drainage and sewage infrastructure. Incorporating these systems into our simulation will provide a more comprehensive view of water management within urban environments, potentially offering insights into how these systems influence flood dynamics and mitigation strategies. One of the inherent limitations of the heightfield fluid simulation technique is its difficulty in accurately modeling water flow under structures like bridges. Research into identifying structures such as bridges in geospatial data could potentially exclude these from the ground map estimation for heightfield fluid simulation.

Future work could also include evaluating different flood mitigation measures beyond the obstructions presented in this study. This could involve comparing various hard and green infrastructure solutions, as relying solely on barriers is not a resilient solution for urban areas. Assessing the feasibility of integrating Best Management Practices (BMPs) into our modeling approach could provide additional insights and enhance urban flood management strategies. The prediction network does not currently account for terrain elevation differences between stations, which is an influential factor that could enhance the prediction accuracy. Future work could integrate terrain elevation information into the prediction network to address this limitation.

Conclusion

In conclusion, our research has demonstrated the substantial potential of advanced simulation tools in enhancing urban flood management and policy development. Through the innovative integration of digital twins, dynamic obstruction modeling and predictive analytics, we have unlocked new avenues for detailed and proactive disaster response planning. The capability to introduce obstructions dynamically within the simulation plays a pivotal role in the context of policy formulation, enabling users to explore various scenarios effectively. Given the constrained timeframe usually associated with flood warnings, the integration of a graph neural network facilitates precise forecasts up to nine days ahead. By considering available resources, including workforce, physical materials, and man hours, this simulation tool significantly enhances decision-making processes at a detailed level. The added feature of navigating through the simulation environment offers a tangible sense of anticipated flood water levels and the potential impact of obstructions. This immersive experience aids in prioritizing and, if necessary, triaging responses to flood events, thereby optimizing resource allocation and response strategies.

(a) QC Govt - Spring Floods 2017/2019 **(b)** Flood simulation based on historical data **(c)** Flood simulation based on our prediction

Figure 5. Simulation with real-world data for spring floods of 2017.

Data availibility

The datasets generated and analyzed during the current study are available from the corresponding author on reasonable request.

Received: 24 February 2024; Accepted: 29 July 2024
Published online: 10 August 2024

References

1. B.c. floods caused at least \$450 m in damage. https://www.cbc.ca/news/canada/british-columbia/bc-flood-damage-1.6280393 (accessed 30 Oct 2023).
2. There were flash floods, strong winds and at least two deaths in mississippi. https://www.nytimes.com/2021/08/31/us/hurricane-ida-mississippi.html (accessed 2 Oct 2023).
3. Gore, J. A. & Banning, J. Discharge measurements and streamflow analysis. In *Methods in Stream Ecology*, vol. 1, 49–70 (Elsevier, 2017).
4. Jaiswal, R., Ali, S. & Bharti, B. Comparative evaluation of conceptual and physical rainfall-runoff models. *Appl. Water Sci.* **10**, 1–14 (2020).
5. Xu, L., Chen, N., Zhang, X. & Chen, Z. An evaluation of statistical, NMME and hybrid models for drought prediction in China. *J. Hydrol.* **566**, 235–249 (2018).
6. Dtissibe, F. Y., Ari, A. A. A., Titouna, C., Thiare, O. & Gueroui, A. M. Flood forecasting based on an artificial neural network scheme. *Nat. Hazards* **104**, 1211–1237 (2020).
7. Roudbari, N. S., Poullis, C., Patterson, Z. & Eicker, U. Transglow: Attention-augmented transduction model based on graph neural networks for water flow forecasting. 2312.05961 (2023).
8. Gamba, P. & Houshmand, B. Digital surface models and building extraction: A comparison of ifsar and lidar data. *Geosci. Remote Sens. IEEE Trans.* **38**, 1959–1968. https://doi.org/10.1109/36.851777 (2000).
9. Schönberger, J. L. & Frahm, J.-M. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)* (2016).
10. Schönberger, J. L., Zheng, E., Pollefeys, M. & Frahm, J.-M. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)* (2016).
11. Massimiliano Pepe, L. F. & Crocetto, N. Use of SFM-MVS approach to nadir and oblique images generated throught aerial cameras to build 2.5d map and 3d models in urban areas. *Geocarto Int.* **37**, 120–141. https://doi.org/10.1080/10106049.2019.1700558 (2022).
12. Mildenhall, B. *et al.* Nerf: Representing scenes as neural radiance fields for view synthesis. In *ECCV* (2020).
13. Xiangli, Y. *et al.* Bungeenerf: Progressive neural radiance field for extreme multi-scale scene rendering. In *European Conference on Computer Vision*, 106–122 (Springer, 2022).
14. Ledoux, H. & Meijers, M. Topologically consistent 3d city models obtained by extrusion. *Int. J. Geograph. Inf. Sci.* **25**, 557–574. https://doi.org/10.1080/13658811003623277 (2011).
15. Yu, D., Ji, S., Liu, J. & Wei, S. Automatic 3d building reconstruction from multi-view aerial images with deep learning. *ISPRS J. Photogramm. Remote Sens.* **171**, 155–170. https://doi.org/10.1016/j.isprsjprs.2020.11.011 (2021).
16. Chentanez, N., Müller, M. Real-time. & eulerian water simulation using a restricted tall cell grid. In *ACM SIGGRAPH,. Papers. SIGGRAPH'*, vol. 11, 2011. https://doi.org/10.1145/1964921.1964977 (Association for Computing Machinery, 2011).
17. Müller, M., Charypar, D. & Gross, M. Particle-based fluid simulation for interactive applications. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 154–159 (Citeseer, 2003).
18. Macklin, M. & Müller, M. Position based fluids. *ACM Trans. Graph.*https://doi.org/10.1145/2461912.2461984 *(2013).*
19. Chentanez, N. & Müller, M. Real-time simulation of large bodies of water with small scale details. In *Symposium on Computer Animation*, 197–206 (2010).
20. Cornel, D. *et al.* Interactive visualization of flood and heavy rain simulations. *Comput. Graph. Forum* **38**, 25–39. https://doi.org/10.1111/cgf.13669 (2019).
21. Kumar, K., Ledoux, H. & Stoter, J. Dynamic 3d visualization of floods: Case of the netherlands. *Int. Arch. Photogramm. Remote Sens. Spat. Inf. Sci.* **XLII-4/W10**, 83–87. https://doi.org/10.5194/isprs-archives-XLII-4-W10-83-2018 (2018).
22. Zivot, E. & Wang, J. Vector autoregressive models for multivariate time series. In *Modeling Financial Time Series with S-PLUS*, 385–429 (2006).
23. Ho, S.-L., Xie, M. & Goh, T. N. A comparative study of neural network and box-jenkins arima modeling in time series prediction. *Comput. Ind. Eng.* **42**, 371–375 (2002).
24. Taylor, J. W. Short-term electricity demand forecasting using double seasonal exponential smoothing. *J. Oper. Res. Soc.* **54**, 799–805 (2003).
25. Makwana, J. J. & Tiwari, M. K. Intermittent streamflow forecasting and extreme event modelling using wavelet-based artificial neural networks. *Water Resour. Manag.* **28**, 4857–4873 (2014).
26. Tan, G., Yan, J., Gao, C. & Yang, S. Prediction of water quality time series data based on least squares support vector machine. *Proc. Eng.* **31**, 1194–1199 (2012).
27. Niroobakhsh, M., Musavi-Jahromi, S., Manshouri, M. & Sedghi, H. Prediction of water quality parameter in jajrood river basin: application of multi-layer perceptron (MLP) perceptron and radial basis function networks of artificial neural networks (ANNS). *Afr. J. Agric. Res.* **7**, 4131–4139 (2012).
28. Medsker, L. R. & Jain, L. Recurrent neural networks. *Des. Appl.* **5**, 2 (2001).
29. Graves, A. & Graves, A. Long short-term memory. In *Supervised Sequence Labelling with Recurrent Neural Networks* 37–45 (2012).
30. Dey, R. & Salem, F. M. Gate-variants of gated recurrent unit (GRU) neural networks. In *2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS)*, 1597–1600 (IEEE, 2017).
31. Vaswani, A. *et al.* Attention is all you need. *Adv. Neural Inf. Process. Syst.* **30** (2017).
32. Wei, X., Wang, G., Schmalz, B., Hagan, D. F. T. & Duan, Z. Evaluate transformer model and self-attention mechanism in the yangtze river basin runoff prediction. *J. Hydrol. Reg. Stud.* **47**, 101438 (2023).
33. Méndez, M., Montero, C. & Núñez, M. Using deep transformer based models to predict ozone levels. In *Asian Conference on Intelligent Information and Database Systems*, 169–182 (Springer, 2022).
34. Wang, C. *et al.* A transformer-based method of multienergy load forecasting in integrated energy system. *IEEE Trans. Smart Grid* **13**, 2703–2714 (2022).
35. Ma, X. *et al.* Learning traffic as images: A deep convolutional neural network for large-scale transportation network speed prediction. *Sensors* **17**, 818 (2017).
36. Khodayar, M. & Wang, J. Spatio-temporal graph deep neural network for short-term wind speed forecasting. *IEEE Trans. Sustain. Energy* **10**, 670–681 (2018).
37. Zheng, C., Fan, X., Wang, C. & Qi, J. Gman: A graph multi-attention network for traffic prediction. *Proc. AAAI Conf. Artif. Intell.* **34**, 1234–1241 (2020).

38. Wu, Z. *et al.* Connecting the dots: Multivariate time series forecasting with graph neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 753–763 (2020).
39. Geng, X., He, X., Xu, L. & Yu, J. Graph correlated attention recurrent neural network for multivariate time series forecasting. *Inf. Sci.* **606**, 126–142 (2022).
40. Makin, J. G., Moses, D. A. & Chang, E. F. Machine translation of cortical activity to text with an encoder–decoder framework. *Nat. Neurosci.* **23**, 575–582 (2020).
41. Huang, Y., Weng, Y., Yu, S. & Chen, X. Diffusion convolutional recurrent neural network with rank influence learning for traffic forecasting. In *2019 18th IEEE International Conference on Trust, Security and Privacy in Computing and Communications/3th IEEE International Conference on Big Data Science and Engineering (TrustCom/BigDataSE)*, 678–685 (IEEE, 2019).
42. Zhou, H. *et al.* Short-term photovoltaic power forecasting based on long short term memory neural network and attention mechanism. *IEEE Access* **7**, 78063–78074 (2019).
43. Wu, S. *et al.* Adversarial sparse transformer for time series forecasting. *Adv. Neural Inf. Process. Syst.* **33**, 17105–17115 (2020).
44. Sutskever, I., Vinyals, O. & Le, Q. V. Sequence to sequence learning with neural networks. *Adv. Neural Inf. Process. Syst.* **27** (2014).
45. He, Z., Chow, C.-Y. & Zhang, J.-D. Stann: A spatio-temporal attentive neural network for traffic prediction. *IEEE Access* **7**, 4795–4806 (2018).
46. Zhou, H. *et al.* Informer: Beyond efficient transformer for long sequence time-series forecasting. *Proc. AAAI Conf. Artif. Intell.* **35**, 11106–11115 (2021).
47. Environment & Canada, C. C. Hydrometric statistics data (2023).
48. de l'Énergie et des Ressources naturelles du Québec, M. Mosaïque landsat. https://www.donneesquebec.ca/recherche/dataset/mosaique-satellites. Jeu de données, Données Québec, publié le 6 décembre 2017, mis ï jour le 6 décembre 2023. Consulté le 27 mai 2024. Contenu soumis ï la licence CC-BY 4.0
49. Shang, C., Chen, J. & Bi, J. Discrete graph structure learning for forecasting multiple time series. arXiv preprint arXiv:2101.06861 (2021).
50. Lin, J. *et al.* Dynamic causal graph convolutional network for traffic prediction. arXiv preprint arXiv:2306.07019 (2023).
51. Tian, C. & Chan, W. K. Spatial-temporal attention wavenet: A deep learning framework for traffic prediction considering spatial-temporal dependencies. *IET Intell. Transp. Syst.* **15**, 549–561 (2021).
52. Epic Games. Unreal engine.
53. Canada, N. R. High resolution digital elevation model (hrdem)—canelevation series—product specifications. https://open.canada.ca/data/en/dataset/957782bf-847c-4644-a757-e383c0057995 (2019).
54. Ville de Terrebonne. Terrebonne urban planning database (2023) (accessed 01 Jun 2023).
55. Québec, D. Cartographie de la canopée de la rmr de montréal en format vectoriel. https://www.donneesquebec.ca/recherche/dataset/canopee-des-six-rmr-du-quebec/resource/95bc3a15-bad9-46b4-b4f2-8000a391d770 (2023) (accessed 01 Nov 2023).
56. Geofabrik GmbH. Openstreetmap data extracts for quebec. https://download.geofabrik.de/north-america/canada/quebec.html (accessed 01 Nov 2023).
57. ArcGIS Web Application—cehq.gouv.qc.ca. https://www.cehq.gouv.qc.ca/zones-inond/ZIS-20190715/index.html (accessed 20 Feb 2024).

Acknowledgements

C.P. This work is financially supported by the Mathematics of Information Technology and Complex Systems' Accelerate programme under grant agreement IT29301 and the Natural Sciences and Engineering Research Council of Canada Grants RGPIN-2021-03479 (NSERC DG). U.E. This research was undertaken, in part, thanks to funding from the Canada Excellence Research Chairs Program. Z.P. This research was undertaken, in part, based on support from the Gina Cody School of Engineering of Concordia University FRS. This work would not have been possible without the invaluable insights and support provided by Rémi Asselin, Directeur, Direction des technologies de l'information, Ville de Terrebonne; Philippe Hamel, Chef de section, sécurité organisationnelle et réseautique, Ville de Terrebonne; Anh Phuong Tran, Architecte de solutions, Ville de Terrebonne; and Sacha Leprêtre, CTO at Presagis Inc.

Author contributions

N.S., S.R., Z.P., U.E., and C.P. wrote the main manuscript text. N.S. implemented the experiments and analyzed the results for the prediction network. S.R. implemented the Simulations and prepared the visualizations. All authors reviewed the manuscript.N.S. and S.R. contributed equally to this work.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1038/s41598-024-68857-y.

Correspondence and requests for materials should be addressed to N.S.R.