Visual Cascade Analytics of Large-scale Spatiotemporal Data

Zikun Deng, Di Weng, Yuxuan Liang, Jie Bao, Yu Zheng, Tobias Schreck, Mingliang Xu, and Yingcai Wu

Abstract—Many spatiotemporal events can be viewed as contagions. These events implicitly propagate across space and time by following cascading patterns, expanding their influence, and generating event cascades that involve multiple locations. Analyzing such cascading processes presents valuable implications in various urban applications, such as traffic planning and pollution diagnostics. Motivated by the limited capability of the existing approaches in mining and interpreting cascading patterns, we propose a visual analytics system called VisCas. VisCas combines an inference model with interactive visualizations and empowers analysts to infer and interpret the latent cascading patterns in the spatiotemporal context. To develop VisCas, we address three major challenges, 1) generalized pattern inference, 2) implicit influence visualization, and 3) multifaceted cascade analysis. For the first challenge, we adapt the state-of-the-art cascading network inference technique to general urban scenarios, where cascading patterns can be reliably inferred from large-scale spatiotemporal data. For the second and third challenges, we assemble a set of effective visualizations to support location navigation, influence inspection, and cascading exploration, and facilitate the in-depth cascade analysis. We design a novel influence view based on a three-fold optimization strategy for analyzing the implicit influences of the inferred patterns. We demonstrate the capability and effectiveness of VisCas with two case studies conducted on real-world traffic congestion and air pollution datasets with domain experts.

Index Terms—Spatial cascade, pattern mining, spatiotemporal data.

1 Introduction

Many spatiotemporal events, such as air pollution and traffic congestion, can propagate across space and time via cascading patterns [25], [28]. For example, the occurrences of air pollution at location A may result from the pollutants spread from location B, while air pollution at location B, in turn, may depend strongly on whether location C is polluted. Studying such cascading patterns presents valuable implications in helping urban data analysts understand the propagation of spatiotemporal events and develop informed countermeasures to improve urban transportation [28], [70], environment [74], and public infrastructures [29], [38], [51].

Efforts have been devoted to studying the inference [27], [33], [41], [74] and visualization [11] of frequent propagation patterns based on massive spatiotemporal data. However, the frequent propagation patterns are different from cascading patterns because the cascading patterns characterize the strong dependencies among several locations and these patterns may be infrequent. To reveal such spatiotemporal dependencies, Liang et al. [28] proposed CasInf, a novel method that infers the cascading patterns of traffic congestion diffusion in road networks. However, since this method is tailored for periodic traffic data, two limitations are observed: CasInf a) operates on discretized time intervals and is inapplicable to inferring cascading patterns from general spatiotemporal data and b) ignores the duration of time when one location is influencing another although time is a crucial factor involved in the cascade analysis [70], [75]. Furthermore, the inferred cascading patterns can be difficult to comprehend and relate back to the original spatiotemporal context due to their probabilistic and uncertain nature. Effective verification and interpretation of these patterns demand well-designed integration of automated approaches' computational power and analysts' domain knowledge.

These limitations motivate us to develop a visual analytics system that infers the cascading patterns from massive spatiotemporal data and interprets these patterns by visualizing the implicit dependencies among spatiotemporal events. Developing such an approach poses the following three challenges:

Generalized pattern inference. To correctly infer the cascading patterns, time should not be partitioned as the approach used by Liang et al. [28] because this time discretization may break cascades and lead to incomplete results. The proposed method should also incorporate the influence durations to capture the reliability of the inferred patterns.

Implicit influence visualization. The influences among locations cannot be observed directly. Hence, the cascading patterns inferred based on these numerous implicit influences are inherently probabilistic and uncertain. Pattern analysis requires visualizing the influences with a scalable and uncertainty-aware visualization.

- *Z. Deng, D. Weng, and Y. Wu are with State Key Lab of CAD&CG, Zhejiang University, Hangzhou, China and Zhejiang Lab, Hangzhou, China. E-mail: {zikun_rain, dweng, ycwu}@zju.edu.cn. Y. Wu is the corresponding author.*
- *Y. Liang is with School of Computing, National University of Singapore, Singapore. E-mail: yuxliang@outlook.com.*
- *J. Bao and Y. Zheng are with JD Intelligent Cities Research, Beijing, China and JD Intelligent Cities Business Unit, JD Digits, Beijing, China. E-mail: baojie@jd.com, msyuzheng@outlook.com.*
- *T. Schreck is with Graz University of Technology, Austria. E-mail: tobias.schreck@cgv.tugraz.at.*
- *M. Xu is with School of Information Engineering, Zhengzhou University, Zhengzhou, China and Henan Institute of Advanced Technology, Zhengzhou University, Zhengzhou, China. E-mail: iexumingliang@zzu.edu.cn.*

Manuscript received April 19, 2005; revised August 26, 2015.

Multifaceted cascade analysis. Multiple analytical requirements demand that the cascading patterns and cascades of events are appropriately organized in terms of spatial topology and temporal occurrences to facilitate efficient exploration. Designing an analysis interface that comprises flexible interactions and coordinated visualizations to support the requirements constitutes the third challenge.

We propose *VisCas*, a visual analytics system that tightly integrates an automatic mining module and interactive visualizations to address the three challenges. For the first challenge, we adapt *network inference* techniques [15], [28] to general urban domains and infer cascading patterns. The extended method incorporates the influence durations and expands its applicability to general urban scenarios by avoiding time partitioning. For the second challenge, we formulate the uncertainties in pattern inference and design compact and intuitive visualizations for massive implicit influence processes. For the third challenge, we design a set of coordinated visualizations and develop an interactive interface to facilitate location navigation, influence inspection, and cascading exploration. Specifically, a three-fold optimization approach is proposed to provide a legible visual summary of the cascading patterns and their influence processes in a novel table-based visualization. The contributions of this study are summarized as follows.

- ◇ We adapt a network inference technique to cascading pattern mining in general urban domains.
- ◇ We design a table-based influence view with a three-fold optimization to visualize massive implicit influences.
- ◇ We develop a visual analytics system called VisCas by integrating the mining algorithm and a set of effective visualizations to enable analysts to analyze spatial cascades interactively.
- ◇ We evaluate our network inference approach and VisCas with case studies on two different real-world datasets.

2 Related Work

This section reviews previous studies on urban visualization and spatial propagation analysis.

Urban visualization [7], [73] has been widely studied for various urban data, such as human mobility [42], [56], [68], public infrastructure [16], [44], [61], [67], environments [1], [47], [50], and social media [4], [6]. Researchers have also attempted to combine automated algorithms with visualization techniques to obtain valuable knowledge hidden behind urban data [12], [19], [30], [31], [60], [62].

Among these studies, we focus on the part related to visual propagation analysis. Wang et al. [58] proposed the first visualization technique to analyze the individual propagation graphs of traffic congestion. Pi et al. [45] leveraged traffic flow and information theories to analyze the cause and influence of traffic congestion. However, these two studies only analyzed the processes that were individually observed within a short period and may even be outliers. They fail to capture the inherent and reliable relationships of locations. Considering long-term observations, Li et al. [26] developed COPE to extract and visualize the event co-occurrences that frequently occur in location pairs to infer latent influences. However, COPE failed to capture the complex relationships (e.g., network) of multiple locations. Deng et al. [11] developed AirVis to visualize the frequent propagation processes of air pollution based on frequent subgraph mining. Although AirVis and COPE summarized numerous influences, they considered only the frequent occurrences but not the dependencies between locations to disclose probabilistic and uncertain cascades.

Spatiotemporal visualizations can be generally classified into two categories [54], namely, integrated views that compactly display the spatial and temporal information simultaneously [21], [32], [64], [65], and linked views that require the costs of interactions and context switching but are effective in multifaceted analyses [13], [37], [57]. The cascading patterns and cascades in our study are informative in terms of spatial context, uncertainties, and temporal dimensions. Hence, we use multiple linked views to present them thoroughly. We coordinate a set of visualizations to support multifaceted cascade analysis and visually connect them using the colors of locations.

Spatial propagation analysis methods can be categorized into model- and data-driven methods. *Model-driven* methods mainly use mathematical or physical models to simulate the effects on different locations. For example, HYSPLIT [52] is widely used to analyze air pollution propagation [11]. Cell transmission model [10] is proposed to simulate traffic flow and congestion [35], [69]. These methods simplify complex real-world environments by introducing many assumptions [22] when building simulation models.

By comparison, *data-driven* methods start from the observed phenomena and infer the underlying patterns among locations. Liu et al. [33] and Wang et al. [58] constructed the diffusion links between congestion events based on their spatiotemporal connection. Frequent pattern mining has been applied beyond modeling individual links to identify significant propagation pathways [8], [11], [27], [41]. These frequency-based approaches can only uncover the general phenomena but not the inherent relationships between them because they ignored the probabilistic dependencies revealed in the influence estimation.

Two probabilistic methods were proposed recently. Zhu et al. [74] utilized Bayesian network to analyze the causal pathways of air pollution. Liang et al. [28] extended network inference techniques [15] for traffic congestion. However, Liang et al. partitioned the time into slices based on traffic periodicity, which cannot be applied to other scenarios. Cascades can occur on an irregular basis, and spatiotemporal events can occur at any time in a dynamic urban environment. For example, factories may emit air pollutants and pollute the environment during operation. Traffic accidents can lead to congestions over traffic cascading patterns. Moreover, both methods ignored the influence durations [70], [75], which are critical in characterizing influence processes and capturing reliable results (e.g., the chance of randomness is high when the duration is short).

By contrast, our mining algorithm exploits the influence durations to characterize the dependencies and randomness of influences further and is applicable to general scenarios by avoiding the time partitioning. Our visual analytics approach also enables pattern verification and interpretation.

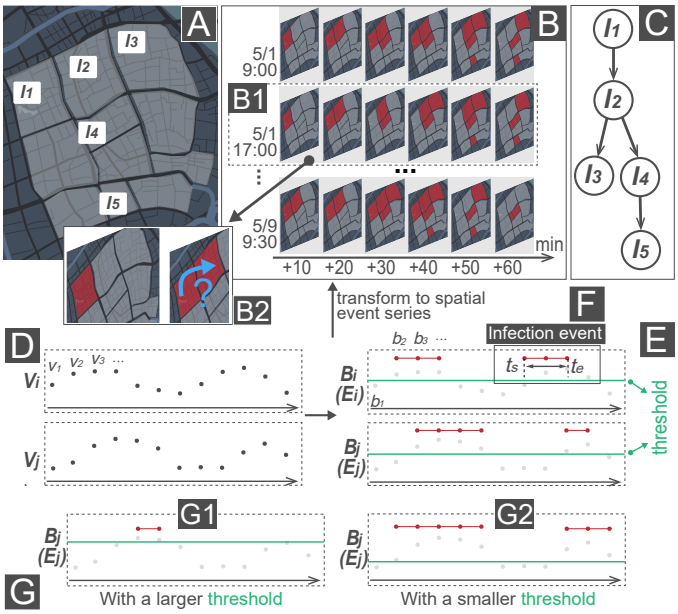

Fig. 1: Relevant concepts for spatial cascade analyses. (A) Multiple geographic locations. (B) The numerous spatial cascades occurring in (A). (C) The cascading pattern inferred from (B). (D) Spatial time series. (E) Spatial event time series. (F) An infection event. (G) The different results of infection event extraction with different thresholds.

3 Background and Concepts

Rapid urbanization has contributed to many notorious urban problems, such as traffic congestion and air pollution [72]. These urban problems generally constitute numerous spatiotemporal events propagating over space and time via cascading patterns. Fig. 1C shows a cascading pattern example that involves five locations $l_1, ..., l_5$ in Fig. 1A. Each propagation process is called a cascade (Fig. 1B). Investigating these cascades and cascading patterns presents valuable implications in many urban applications, such as vulnerability identification and spatiotemporal prediction. For example, the cascading patterns extracted from traffic congestion events can enable local authorities to take effective countermeasures, such as improving the road conditions at the patterns' roots and pathways.

Extracting and analyzing such patterns remain challenging. Most urban sensors generate a series of records that comprise locations, timestamps, and readings. Recovering implicit spatiotemporal influences from these records is difficult. For example, Fig. 1B2 illustrates the sensor records collected at two consecutive timestamps by denoting the abnormal locations (l_1 on the left and l_1, l_2 on the right) with red. The implicit influence indicated with the blue arrow between l_1 and l_2 must be inferred from the discrete records. Moreover, cascading patterns must be characterized from these influences to support domain analyses and insight retrieval. Hence, we propose a visual analytics approach that combines an inference algorithm with interactive visualizations to facilitate extraction, verification, and interpretation of cascading patterns.

We draw an intuitive analogy between the propagation of spatiotemporal events and the spread of viruses and introduce the following relevant concepts:

- A *spatial contagion* is defined as an event that occurs in geo-space and can expand its influence to cause urban deterioration, such as a traffic congestion event.
- A *spatial time series* $A_l = \langle a_1, a_2, ... \rangle$ is a chronologically ordered set of values detected at location l (Fig. 1D). Each value $a_t \in A$ is collected at time t.
- A *spatial event time series* $B_l = \langle b_1, b_2, ... \rangle$ is a chronologically ordered set of Boolean values (i.e., $b_t \in \{0, 1\}$) that indicate whether an event has occurred at location l at time t. Fig. 1E shows two spatial event time series where the red dots indicate the occurrences of events (i.e., $b_t = 1$). A spatial event time series B_l can be obtained directly from sensors or a spatial time series $A_l = \langle a_t \rangle$ by $b_t = P(a_t)$ based on a predicate $P : \mathbb{R} \to \{0, 1\}$. For example, the predicate P_θ we use in Fig. 1E determines event occurrences based on whether the value of a_t exceeds a given threshold θ as follows:

$$P_\theta(a_t) = \begin{cases} 1, & \text{if } a_t > \theta \\ 0, & \text{otherwise} \end{cases}$$

  The value of θ is denoted by the green lines in Fig. 1E. The threshold θ or the predicate P itself is determined by actual applications in practice (two examples are provided in Sec. 4.1). Fig. 1G shows how event detection varies with different thresholds. Fewer events are detected with a larger θ (Fig. 1G1), eventually producing the patterns that capture the cascades of extreme events (e.g., hazardous air pollution events).
- An *infection event series* $E_l = \langle e_{l;t_1,t_2}, e_{l;t_3,t_4}, ... \rangle$ can be extracted from a spatial event time series $B_l = \langle b_t \rangle$ to characterize continuous events at location l. An *infection event* $e_{l;t_s,t_e} \in E_l$ indicates that the spatial contagion event continuously occurs from time t_s to t_e at location l (Fig. 1F). We have $\wedge_{t=t_s}^{t_e} b_t = 1$ while $b_{t_s-1} = 0$ and $b_{t_e+1} = 0$ for each $e_{l;t_s,t_e} \in E_l$. We denote the start time of $e_l \in E_l$ as $t_s(e_l)$ and the end time as $t_e(e_l)$ for a compact description.
- An *infection* or *influence* describes a process where the spatial contagion is propagated from one location u to another v (the blue arrow in Fig. 1B2), that is, v is infected by u. An infection reveals the implicit causal relationship between two infection events occurring at locations u and v.
- A *cascade* reveals how a spatial contagion spreads over the locations and infect them (Fig. 1B1). Cascades can be considered directed graph data where vertices represent infection events and edges represent influences.
- A *cascading pattern* or *network* $\mathbb{G} = (V, S)$ is a directed graph extracted from a sheer volume of cascades to describe the spreading pattern of spatial contagions (Fig. 1C). Each vertex $v \in V$ represents a location l, and each directed edge $(v_i, v_j) \in S$ indicates that v_j *depends* on v_i, i.e., if v_i is infected, then v_j will likely be infected also. Such a relationship is called *dependency*.

This study aims to identify the underlying cascading network $\mathbb{G}$ effectively and facilitate the reliable verification and interpretation of this network with visual analytics given a set of locations and their associated spatial time series $\{A_{l_1}, A_{l_2}, ...\}$.

4 MODEL

This section presents the mining framework for inferring cascading patterns in general spatiotemporal domains.

4.1 Infection Event Extraction

First of all, infection events are extracted from the given spatial time series $\{A_{l_1}, A_{l_2}, ...\}$ by converting the series to spatial event time series $\{B_{l_1}, B_{l_2}, ...\}$ with application-specific predicates and identifying the start and end times of continuous events. For example, fine particulate air pollution events can be determined based on the spatial time series of $PM_{2.5}$ (i.e., the density of fine particles with a diameter of $2.5\,\mu m$ or less) with the predicate P_θ. According to WHO's air quality guidelines [43], the threshold θ in this predicate is set to $75\,\mu g\,m^{-3}$ to find the extreme air pollution events where the value of $PM_{2.5}$ fails to even meet the lowest interim air quality targets for particulate matter. Similarly, we extract traffic congestion events based on the spatial time series of traffic speed with a predicate that determines whether the speed is lower than $20\ km/h$, which is a threshold commonly seen in traffic congestion studies [23], [34], [40]. In this way, an infection event series E_l can be obtained for each location l.

4.2 Cascading Network Inference

This section describes the approach we adopt to infer the cascading network $\mathbb{G}$ based on the infection event series.

4.2.1 Congestion Cascading Network Inference

This subsection briefly summarizes the core idea of the previous method by Liang et al. [28], which is the foundation of our approach. This method is designed to infer traffic congestion cascading patterns and comprised of two major parts, namely, influence modeling and network extraction.

Influence Modeling. To extract traffic congestion events, the previous method exploits the periodicity of traffic congestion by partitioning the time into a set of slices $C = \{c_1, c_2, ...\}$ (Fig. 2A1) and identifies at most one event e_u for each location u in every slice (Fig. 2B). Liang et al. use the following monotonic exponential model to model the influence likelihood between two events e_u and e_v occurring in the same slice:

$$f(e_v|e_u) \propto \exp(-\alpha(d_{ts}(e_u, e_v) + \lambda d_s(u, v))),$$

where α is the transmission rate, $d_{ts}(e_u, e_v)$ denotes the start time difference $t_s(e_v) - t_s(e_u)$ that is always non-negative, $d_s(u, v)$ denotes the spatial distance between u and v, and λ is the trade-off parameter between two terms.

The previous method also considers environmental influence. For example, congested traffic may be caused by a crowded parking lot rather than nearby roads. To incorporate such influence, the previous method estimates an environmental factor ϵ_g for each spatial grid g with supervised learning and derives a weighted likelihood for a pair of roads u and v in a slice c: $w_c(u, v) = \epsilon_g^{-1} f(e_v|e_u)$, where $e_u, e_v \in c$, $v \in g$.

Network Extraction. Liang et al. use a greedy algorithm to extract the cascading network $\mathbb{G} = (V, S)$. Starting from an empty network, this algorithm evaluates every pair of roads that have not been added to the network in all slices and determines the pair (p, q) that maximizes

$$\sum_{c \in C} \log(1 + \frac{w_c(p, q)}{\sum_{i \in V \cup p; i \neq q} w_c(i, q)}).$$

(p, q) is subsequently inserted into $\mathbb{G}$. Repeat this process until the number of edges reaches a given threshold k.

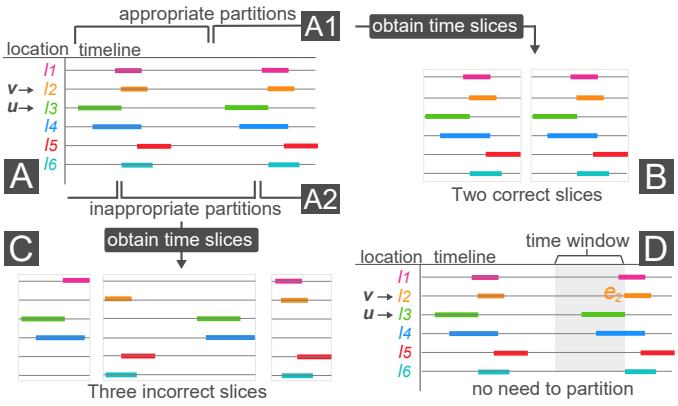

Fig. 2: (A) Illustrative infection event series. (B) Two correct slices extracted based on (A1) the appropriate partitions. (C) Three incorrect slices extracted based on (A2) the inappropriate partitions. (D) Our time window strategy that does not need time partitioning.

4.2.2 Generalized Cascading Network Inference

Following Liang et al., we also use a similar monotonic exponential model for characterizing influence likelihood. However, two limitations of Liang et al.'s approach must be addressed for generalized pattern inference.

The first limitation lies in data partitioning. Liang et al.'s approach requires the temporal partitioning of traffic congestion events first because traffic congestion occurs periodically. Applying this partitioning strategy to the generalized pattern inference has the following drawbacks: (a) many datasets, such as air quality data spanning across months, are not periodic; (b) improper partitioning may break long cascades (Fig. 2A2) and lead to incorrect cascading patterns; and (c) extracting only one infection event for each location in every time slice prohibits fine-grained cascading pattern analysis. To address this limitation, we allow multiple infection events to be detected for each location as described in Sec. 4.1 and introduce a time window parameter t_w. As illustrated in Fig. 2D, we only search in a time window of t_w before the occurrence of the infection event e_2 for the events that potentially cause e_2. Only the latest event in each location will be searched.

The second limitation is that their approach does not consider the durations of infection events, which are critical in the cascade analysis [70], [75]. We address this limitation by integrating the durations as an external factor ϵ_d. Specifically, we denote the duration of an infection event e as $t_d(e) = t_e(e) - t_s(e)$ and the normalized frequency of the infection events occurring at location u as $f_u \in [0, 1]$. For any two infection events e_u and e_v occurring at locations u and v, we have $\epsilon_d(e_u, e_v) = {f_u}^{t_d(e_u)} {f_v}^{t_d(e_v)}$. Longer infection durations will lead to lower external factors, thereby re-

sulting in higher influence likelihood. However, long event durations are required to maintain reliable inference for the events occurring at frequently infected locations.

Therefore, we derive a new weighted likelihood model that estimates the influence of infection event e_u occurring at location u on e_v occurring at v as follows:

$$w(e_u, e_v) = \epsilon_d(e_u, e_v)^{-1} f(e_v|e_u).$$

We then utilize the greedy approach proposed by Liang et al. to generate the cascading network $\mathbb{G}$ by iteratively adding the pair of locations p and q that maximizes

$$\sum_{e_q \in E_q} \log(1 + \frac{w(R_p(e_q), e_q)}{\sum_{i \in V \cup p; i \neq q} w(R_i(e_q), e_q)}),$$

where $R_p(e_q)$ denotes the latest event occurring at location p in the time window t_w before e_q. The algorithm pseudocode and time complexity estimation of network extraction are presented in Appendix A.

4.3 Cascade Retrieving

Based on the extracted cascading network $\mathbb{G} = (V, S)$, we further describe the procedure for obtaining the cascades of infection events that reveal how spatial contagions spread.

We denote the events that are caused by event e_l as $C(e_l)$, which can be computed as follows. For each edge $(u, v) \in S$ and each event $e_u \in E_u$, we find the latest event $e_v \in E_v$ that occurs in the time window t_w after the occurrence of e_u and add e_v to $C(e_u)$. We can then reconstruct a graph with C and obtain each connected component as a cascade of events.

5 User Requirements

We collaborated closely with four domain experts (Experts A, B, C, and D denoted EA, EB, EC, and ED, respectively) committed to urban science for many years. EA and EB have decades of experience in utilizing data-driven approaches to study various urban problems. EC is a Ph.D. candidate working on urban issues with machine learning, including network inference. ED has been engaged in combining environmental science and geoscience research for the past decade. We follow an iterative user-centered design process [39] in this study. We iteratively refine user requirements as well as a prototype system with bi-weekly expert interviews and literature review. Finally, we compile seven requirements in three aspects as follows.

N. Location navigation helps specify the area where the experts are going to analyze spatial cascades.

R1 **Provide location overviews.** First, the experts need to identify and select interesting locations like Liang et al. did in their study [28]. They pay attention to the spatial context and temporal distribution of the locations' infection events. The system should allow access to the spatiotemporal overview for each location.

R2 **Recommend potential locations.** The experts can have no specific prior knowledge for a target area to initialize the analysis. The system should recommend several potential locations, where strong infection dependencies and valuable patterns may be revealed. In particular, the experts believe that the temporal co-occurrences of infection events can enhance this recommendation.

I. Influence inspection assists the experts in analyzing the uncertain and implicit influences among locations for verifying and interpreting cascading patterns.

R3 **Summarize various cascades.** A cascading network comprises massive cascades of infection events. Each cascade involves multiple location influences due to the same contagion. Summarizing these cascades prior to the influence analysis is strongly required to learn how contagions propagate; for example, *which locations are all infected by contagions frequently?*

R4 **Display the uncertainties of influences.** An infection event has many possible upstream infection events. Although the algorithm has estimated the most likely upstream, the system should reveal other possibilities to understand the reliability and uncertainty of the inference result, such as *how does the possibility of the inferred upstream differ from that of others?*

R5 **Characterize influencing processes.** The experts can understand how a downstream event depends on its upstream and what happened between them by examining location influences. The following questions may be asked: *Does the downstream get easily infected when its upstream is infected? Are the infection situations similar?* Therefore, the system should visually depict influencing processes.

E. Cascading exploration helps the experts explore the cascades of cascading patterns and unfold them in detail.

R6 **Investigate the temporal characteristics of cascades.** Cascades can occur many times and lead to different degrees of infections. The experts aim to obtain the temporal distribution of cascades (*When do the cascades usually happen? Winter? Rush hour?*) and the cascading effects on the involved locations (*Does the cascades last long?*). Thus, temporal visualizations for cascades should be implemented in the system.

R7 **Unfold individual cascades.** The experts tend to access detailed infections and obtain convincing results by unfolding the cascades. They must understand how the contagion spreads over the locations involved in a cascade to recover the reasons behind urban deterioration.

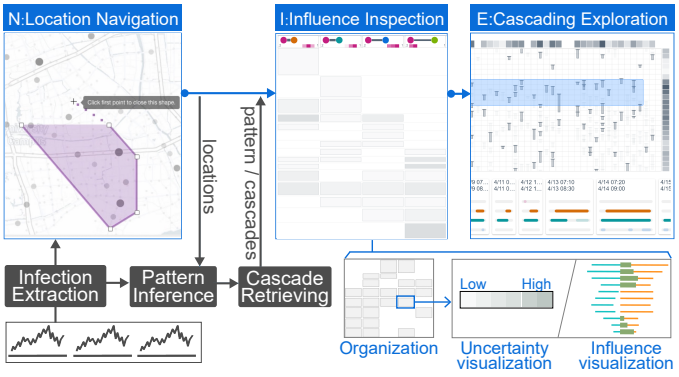

Fig. 3: Workflow and system architecture. The blue parts are the user requirements in three aspects and their corresponding visualizations. The black parts show the model.

Questions may be asked, for instance, *when did A get infected? Were B and C infected after A got infected? How long was the delay? How severe was the cascade?*

6 VisCas

We develop VisCas to satisfy the seven requirements. Fig. 3 presents the system overview. The system is refined and improved iteratively based on the user requirements. We fully consider the experts' suggestions collected from interviews during the iteration. As a result, the system adopts many representations familiar to the experts, such as tables and timelines. We also carefully explore design spaces and justify our designs by following the design principles from the visualization community.

VisCas comprises three views, namely, spatial, influence, and cascading views. First, the *spatial view* (Fig. 4A) offers an effective and interactive environment for navigating locations (**N: R1 and R2**). Users can explore and select interesting locations based on their knowledge and then run the inference algorithm. The inferred cascading pattern is also presented in the spatial view. Second, the *influence view* (Fig. 4B) helps inspect the influencing and cascading processes among locations (**I: R3-R5**). This view depicts the numerous cascades and influences in the pattern. Users can verify the pattern through uncertainty visualizations and interpret the pattern by analyzing the influencing processes. Finally, the *cascading view* (Fig. 4C) supports exploring numerous cascades and drilling down to individual ones (**E: R6 and R7**). The temporal characteristics of the cascades are revealed by a folded timeline. Users can further check the details of the cascades by brushing the timeline.

6.1 Spatial View

The spatial view (Fig. 4A) comprises a map (left) and a location projection (right) to assist users in navigating locations (**N: Location navigation**).

6.1.1 Map

Maps (Fig. 4A4) are widely used in urban analysis scenarios [59]. We use a level-of-detail mechanism for the map. Each location is represented by a circle, and the size and opacity of the circles encode the number of events. Zooming in on the map reveals the spatiotemporal overviews (Figs. 8B2 and 4A6) for the locations (**R1**). Inspired by the widely used radial layout for temporal data [32], [46], we encode the temporal distribution of events with a radial heatmap around the circle for each location. The opacity of each sector encodes the number of events aggregated from the corresponding timespan. Different aggregation strategies can be applied. For example, for traffic congestion events, each sector represents an hour of every day, resulting in 24 sectors; and for air pollution events, 12 sectors correspond to 12 months of a year.

The map also depicts the cascading patterns inferred by the model as spatial networks (Fig. 4A4). Each location is assigned with a unique color that is consistent throughout the system.

6.1.2 Location projection

The projection (Fig. 4A1) reveals interesting location subsets when users have no exact prior knowledge (**R2**).

If 1) a pair of locations are close to each other and 2) the infection events in these two locations frequently co-occur, then these two locations are highly likely infected simultaneously. The first condition can be easily checked on the map. We design the projection based on a two-level distance measurement to capture the event co-occurrences between two locations, such as, m and n.

First, we introduce an *infection-level* co-occurrence distance function for $e_m \in E_m$ and $e_n \in E_n$ as follows:

$$D_i(e_m, e_n) = \frac{|d_{ts}(e_m, e_n)| + |d_{te}(e_m, e_n)|}{t_d(e_m) + t_d(e_n)}.$$

$d_{ts}(e_m, e_n)$ and $d_{te}(e_m, e_n)$ denote the differences between the start times and end times, respectively, and $t_d(e) = t_e(e) - t_s(e)$ denotes the duration of e. Two co-occurring infection events should start and end at similar timestamps (the numerator of $D_i(e_m, e_n)$) and these two events should both last for a long time (the denominator of $D_i(e_m, e_n)$).

Second, we introduce a *location-level* distance function between locations m and n based on the infection-level distance. Specifically, we utilize a classic framework for time series called dynamic time warping (DTW) [2], [3], [17], [24]. DTW can automatically pair the items in two sequences and compute the similarity of sequences by accumulating the distances between the paired items. In our scenario, we feed E_m and E_n into DTW and use the infection-level co-occurrence distance $D_i(e_m, e_n)$ as the distance function for the paired events. Hence, DTW will produce the location-level distance, which measures the degree of event co-occurrences at locations m and n.

Finally, we feed a matrix that comprises the distances between all pairs of locations into t-SNE and obtain projected 2D coordinates for each location. The coordinates are subsequently plotted as a scatterplot. The numbers of events are encoded with the opacity of scattered points.

6.2 Influence View

The influence view (Fig. 4B) helps users verify and interpret a cascading pattern by inspecting its uncertain and implicit influences (**I: Influence inspection**). We describe the influence view from the two aspects: the organization of cascades and the visualization of location influences.

6.2.1 Organization of cascades

The influence view adopts a table-based layout to organize and summarize cascades in an occlusion-free manner (**R3**), as illustrated in Fig. 5.

Table-based Layout. Each column represents an edge in the cascading network. For example, the last column in Fig. 5B represents edge 5 in Fig. 5A. Each row represents a group of cascades comprising the same topology. For example, the first row in Fig. 5B represents the cascade group in Fig. 5a. The height of each row encodes the number of cascades. The cell $cell_{i,j}$ at the i-th row (cascade group) and j-th column (edge) comprises the influences associated with the corresponding cascade group and the edge. For example, the top-right cell in Fig. 5B comprises the

Fig. 4: The system interface of VisCas. (A) The spatial view enables users to navigate and select locations and analyze cascading patterns. (B) The influence view organizes numerous influences and assists users in analyzing their uncertainties and dependencies. (C) The cascading view presents massive cascades for exploration.

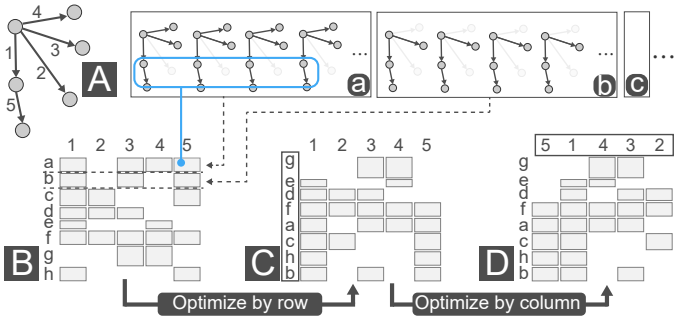

Fig. 5: Organization of cascades and layout optimization. (A) An illustrative cascading pattern and (a, b) its two cascade groups. (B) Our initial table-based layout for organizing cascade groups. (C, D) The legible layouts optimized by the row and column orders in (B), respectively.

influences associated with the part of cascading networks enclosed inside the blue rectangle in Fig. 5a. We visualize the uncertainties and influencing processes of its associated influences in a cell (Sec. 6.2.2). Each column header shows 1) the upstream (left) and downstream (right) locations of the edge and 2) the likelihood distribution of all influences associated with the edge (Fig. 4B4).

Improvement. The tabular representation can organize numerous cascades and influences in an unobstructed manner, but some patterns may become illegible if the layout of the representation is not optimized. For example, frequently co-occurring edges 1 and 5 can be easily seen in Fig. 5D but unclear with a scattered layout shown in Fig. 5B. Achieving a legible layout requires reordering the rows and columns.

We can naturally formulate this problem of finding the optimized order (path) as the *Hamiltonian path* problem [48] because the order of the rows or columns is equivalent to a path that visits each row or column exactly once. The Hamiltonian path problem is to identify the path with the minimum cost (the sum of path weights) that visits each vertex exactly once in a weighted graph. The weighted graph in our scenario is a complete graph where each vertex denotes a row or column. The weight between a pair of rows or columns indicates the cost of placing them side by side. The optimal Hamiltonian path represents the optimal order of the rows or columns. Thus, the key problem is to define the weights between the rows or columns.

Optimizing the row order. Adjacent rows should share many cells and have similar heights. We first formulate each row i as a set, where each item j indicates $cell_{i,j}$ exists. Then, we obtain the weight between each pair of the row sets by computing the Jaccard distance multiplied by the height difference between these two rows. After optimizing the row order, the appearance becomes continuous in the vertical direction but visual discontinuities are still observed in the horizontal direction (Fig. 5C).

Optimizing the column order. Similarly, adjacent columns should share many cells. We first formulate each column j as a vector with the dimension as the number of rows, where the i-th item is the height of the i-th row. If the column j has no cell in the i-th row, then the i-th item of the vector is zero. We then use the weighted Jaccard distance [9], [20] (an

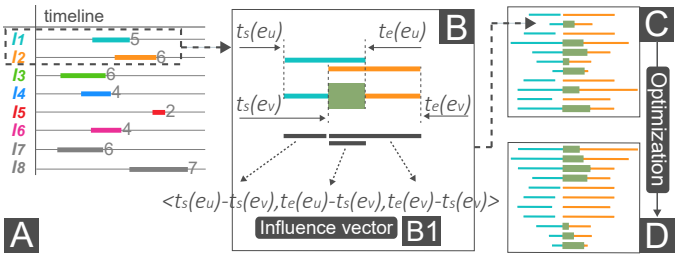

Fig. 6: Visualization of location influences. (A) A cascade instance contains an inferred influence from u to v. (B) The influence instance is visualized with an influence glyph. (B1) An influence vector portrays the glyph. (C) Stacked influence glyphs visualize multiple influences. (D) The order of the influence glyphs in (C) has been optimized.

extended version of the Jaccard distance on positive vectors) between two vectors as the weight between their columns. Finally, we obtain a legible layout (Fig. 5D) where both the row and column orders are optimized.

In this way, users can easily summarize the cascades and influences. The details of the weight functions above are available in Appendix B.

6.2.2 Visualization of location influences

Within each cell, we summarize the uncertainties of the influences comprised by the cell (**R4**) and portray these influencing processes (**R5**). The uncertainty is displayed by default. Users can switch between the uncertainty and influence visualizations.

Uncertainty Visualization. A cascading network is summarized from long-term observations. Thus, for a specific infection event e_v, other possibilities e_ms ($m \neq u$) with a larger likelihood $f(e_v|e_m)$ than the inferred $f(e_v|e_u)$ may exist. Uncertainty visualization should reflect how much the inferred upstream e_u exceeds e_ms.

According to the experts' suggestions, those e_ms with a larger likelihood and longer duration are denoted more likely causes. For the influences comprised by the same cell, we derive an uncertainty metric by dividing the number of the more likely causes by the number of influences. The metric is then encoded with the cell luminance. For example, most cells in Fig. 4B5, except those at the bottom, are bright, indicating that most of the influences are certain.

Influence Visualization. We visualize every *single influence* with a timeline-based *influence glyph* (Fig. 6B), where the two infection events are aligned along the timeline. We place these two events on the same horizontal position and make the overlapping part bold for highlight, thereby improving the scalability and readability. Colors are assigned according to the corresponding locations, and the overlapping part mixes both. Subsequently, we visualize *multiple influences* by stacking the multiple glyphs and aligning them according to the downstream events' start times (Fig. 6C).

However, the visually intermittent appearance prevents users from analyzing these influences, for example, grasping the distribution of start time differences. Improving the readability is also an order optimization problem like that in the influence view. We address this issue based on the Hamiltonian path problem again. Here, the weights are Euclidean distances between *influence vectors* (e.g., $\langle t_s(e_u) - t_s(e_v), t_e(e_u) - t_s(e_v), t_e(e_v) - t_s(e_v)\rangle$ in Fig. 6B1). Such vectors properly portray the appearance of the influence glyph in the horizontal direction. We obtain a clear appearance in the end (Fig. 6D).

6.2.3 Solving Hamiltonian path problem

Our designs use layout optimizations based on the Hamiltonian path problem in three parts, the orders of 1) stacked multiple influence glyphs and 2) the column and 3) the row of the influence view. However, solving the Hamiltonian path problem within a time of fluid interaction is difficult. The time complexity of its dynamic programming solution is $O(2^{|V|}|V|^2)$, where $|V|$ is the number of vertices. The run time becomes prohibited when $|V|$ is larger than 30. Various heuristic algorithms, such as Genetic and Ant Colony algorithms, have been used to approximate the optimal solution to similar problems [18], [49], [53]. We must sacrifice some accuracy while maintaining the overall visual quality of the layouts to obtain a satisfactory solution for dozens of vertices in a short time; for example, 300 ms. We finally apply a local search algorithm [55] to iteratively conduct the item swaps that obtain the maximum gain in each iteration.

6.2.4 Justification

We justify the two main visual designs in the influence view.

Organization of cascades. We also create two alternatives to organize cascades based on the two commonly used graph visualizations [14], namely, adjacency matrix and node-link diagram [71]. Each cascade can be visualized as a node-link diagram and squashed by placing the nodes on the same horizontal position (Fig. 7A). Compactly stacking them can handle many cascades. However, the limited visual channels for the links hinder the effective encodings of its associated information. Each cascade can also be represented as an adjacency matrix (Fig. 7B). This encoding is even worse than that in Fig. 7A. The matrix-based representation is space inefficient because of its sparsity. Finally, we adopt the table-based visualization (Fig. 7C), a scalable and clutter-free manner, as our final design. The influences in each cell can be unfolded by embedding effective visualizations.

Influence visualization. Influences are essentially multi-dimensional data. We explore commonly used visualizations, such as parallel coordinates plot, multiple coordinated views, dimension reduction, and glyph-based methods, for this kind of data. Only the proposed timeline-based glyph (Fig. 6E) can intuitively and compactly present the temporal feature of an influencing process. Moreover, the experts can immediately understand the glyph design because of their familiarity with timelines.

6.3 Cascading View

The cascading view (Fig. 4C) consists of a *temporal chart* (top) and a list of *cascading cards* (bottom) to allow users to explore

Fig. 7: (A, B) Two alternatives and (C) our final design for organizing groups of cascades with different topologies.

the numerous cascades of cascading patterns (**E: Cascading exploration**).

Temporal Chart. We visualize many cascades along a long timeline and expose their temporal characteristics to users (**R6**). As illustrated in Fig. 4C3, the temporal chart folds the timeline from top to bottom and left to right. Each cascade is represented as a bar. The positions of its upper and lower borders encode the start and end times of the cascade, respectively. The bars' occurrences are aggregated in both horizontal and vertical directions as the heatmaps above and to the right of the chart, respectively.

Cascading Card. Each card presents an individual cascade in detail (**R7**) and mainly constitutes multiple timelines. Each one indicates how a specific contagion infects an involved location and is colored according to the location. The card header displays the start and end times of the cascade. The cascading card design is borrowed from the experts' hand-draft of cascades (Fig. 2).

6.4 User Interactions

The following useful interactions are implemented.

Interactive Mining. VisCas enables the interactive customization of the model inputs and obtains inference results in real time. Users can specify locations by drawing polygons in the geographic map or using a lasso tool in the projection view. The number of edges k in the cascading pattern and the time window t_w can also be tuned via the sliders in Fig. 4A7.

Filtering and Selecting. Users can focus on the cascades of their interests by filtering and selecting. They can click the column headers in the influence view. The cascades without the clicked columns (edges) are filtered from the influence and cascading views. Users can also click rows to select cascade groups. For example, the header of $e3$ and the first row are clicked in Fig. 4B. Besides, users can brush the temporal chart (Fig. 4C4), and these brushed cascades (i.e., bars) are shown with cascading cards (Fig. 4C2).

Switching. We design the uncertainty and influence visualizations in the influence view to reveal location influences from the two aspects. Users must know uncertainties (**R4**) before the influence analysis (**R5**). Thus, the uncertainty visualizations are displayed by default. When a row is selected (e.g., the first row in Fig. 4B) or a floating button of a column is clicked (e.g., $e4$ in Fig. 8C), the involved cells will be switched to the influence visualizations.

6.5 Implementation

VisCas is a web-based application with a frontend and a backend. The frontend is written in Vue.js and TypeScript, runs in modern web browsers, and enables users to interact with the system and conduct in-depth cascade analyses. The backend is implemented in GO (https://golang.org/). It integrates 1) a data mining module to infer cascading networks given input parameters (Sec. 4.2) and retrieve cascades (Sec. 4.3) and 2) a layout optimization procedure to produce readable layouts in the influence view (Sec. 6.2.3).

7 Case Studies

We demonstrate the effectiveness and capability of VisCas with two case studies and four interviews with the four experts. We introduced VisCas, including the workflow, visual encodings, and user interactions, to the experts to help them familiarize with the system, before the case studies. Thereafter, the experts utilized our system to analyze the spatial cascades extracted from two different datasets, namely, traffic congestion and air pollution datasets, in an attempt to gain insights into urban planning. Finally, we interviewed the experts individually and obtained feedback.

7.1 Cascade Analysis of Traffic Congestion

The experts aimed to identify traffic bottlenecks or congestion development patterns in Hangzhou and subsequently formulate policies for improved transportation, such as infrastructure upgrade and intelligent traffic control.

Dataset and Processing. We retrieved a congestion dataset from taxi trajectories in Hangzhou following prior studies [33], [45], [66]. Taxi trajectories from local authorities recorded the trajectories of 8,816 taxis between March and April 2016. Low-level roads, such as village roads, were filtered and resulted in 944 major road segments. We computed the average speed of the taxis passing through each road for every 10 minutes as the travel speed, which constitutes a set of spatial time series. A road segment is considered congested if the travel speed did not exceed $20\ \mathrm{km/h}$ (see Sec. 4.1). Finally, 944 time series data of congestion events were extracted and fed into VisCas.

Location Navigation. The experts started with the projection view (Fig. 4A1) given that they had no preference for regions. Locations close to one other in the projection view were likely influenced by the same contagion. Few dark and many bright dots in each location cluster indicated diverse degrees of congestions. EC explained that "*congestions were often caused and spread by some locations in a bad traffic situation.*" The experts selected the most abnormal location cluster (Fig. 4A2) with the Lasso tool based on their interest after examining the projection view. A group of geographically adjacent locations in the map view (enclosed in Fig. 4A3) caught the attention of the experts. Two of these locations were heavily congested, and one was a part of an arterial road.

The experts ran the network inference model in the selected region. The time window t_w was set to 1 hour considering the backward propagation speed of traffic flows. The inferred cascading pattern and an overview of the involved locations were shown in Fig. 4A4. The experts noticed that the cyan and orange locations were more congested because their inner circles were larger than those of the two other locations (Fig. 4A6). The following findings are drawn from the analysis of the 24-hour congestion distribution: 1) for the cyan and orange locations, severe congestions (darker sectors) tended to occur in the morning; 2) for the blue and cyan locations and 3) for the purple and orange locations, the trends in the afternoon were similar. EB concluded that the temporal distributions of these locations were similar in certain periods, thereby indicating the presence of many congestion co-occurrences and potential dependencies. He also praised the spatial view saying, "*it helped me quickly identify interesting areas.*"

Influence Inspection. The experts analyzed influences with the influence view (Fig. 4B) to reveal the inherent

relationships among locations. For convenience, we denoted the four columns (each representing an edge) from left to right as $e1$-$e4$. First, the longer cells in $e3$ than others (Fig. 4B5) indicated that the influences from the cyan location to the orange location occurred frequently. Therefore, the cyan location could be a significant traffic bottleneck. Moreover, $e1$-$e4$ rarely co-occurred (Fig. 4B6). EA suggested that this might be because $e1$-$e4$ represented different traffic demands (e.g., the different demands in the morning and evening rush hours); thus, the traffic deterioration at these locations could be caused by different contagions.

The experts then explored the influences between the top two congested locations., that is, the cyan and orange locations linked by $e3$. EA observed that the overall likelihoods of $e3$ were lower than that of others (Fig. 4B4). Nevertheless, the cells in $e3$ were mostly bright, thereby indicating that their inference results were generally reliable and deserved further analysis.

The experts examined the influence processes with the influence glyphs (Figs. 4B1-4B3), which were made readable with the proposed layout optimization. They categorized these glyphs into three modes based on the relationships between the infection durations in upstream DU and those in downstream DD and explained these modes accordingly. 1) $DU \simeq DD$ (Fig. 4B2): The downstream events largely depended on the upstream events, which was the most common mode. "*The varied response times might be due to the different traffic wave speeds in the backward propagation*" [10]; 2) $DU \gg DD$ (Fig. 4B3): Other events might also contribute to the downstream events; 3) $DU \ll DD$ (Fig. 4B1): The traffic volume was small, or drivers chose other routes to avoid the upstream congestions.

Cascading Exploration. The cascades only containing $e3$ were selected by clicking the first row in Fig. 4B. The heatmap on the right side of Fig. 4C1 suggested that these cascades mainly occurred in the morning. These morning cascades were further brushed and selected by the experts (Fig. 4C4). The cascading cards in Fig. 4C2 showed that most upstream congestions influenced the downstream locations within half an hour. EB commented that "*such information can facilitate intelligent routing for the traffic demands in the morning*." Besides, the cascades containing $e1$ mainly occurred in the afternoon (Fig. 4C5). This temporal difference in the cascades' occurrences further confirmed that traffic deterioration was due to different traffic demands.

Conclusion. The traffic infrastructure development after 2016, when our data were collected, validates the aforementioned patterns. 1) The Wenyi Tunnel that passes through the blue and cyan locations started operations in October 2018. 2) Metro Line 10, which is currently under construction, stops at both the cyan and orange locations; 3) Metro Line 2, which was opened in 2017, connects the purple and blue locations (where passengers can interchange to Line 10) and goes downtown. This case showed that our system could help reveal and interpret the traffic cascades concealed in congestion observations.

7.2 Cascade Analysis of Air Pollution

Before 2016, the heavy industry (e.g., coal-fired heating) in northern China spread pollution to the Yangtze Delta

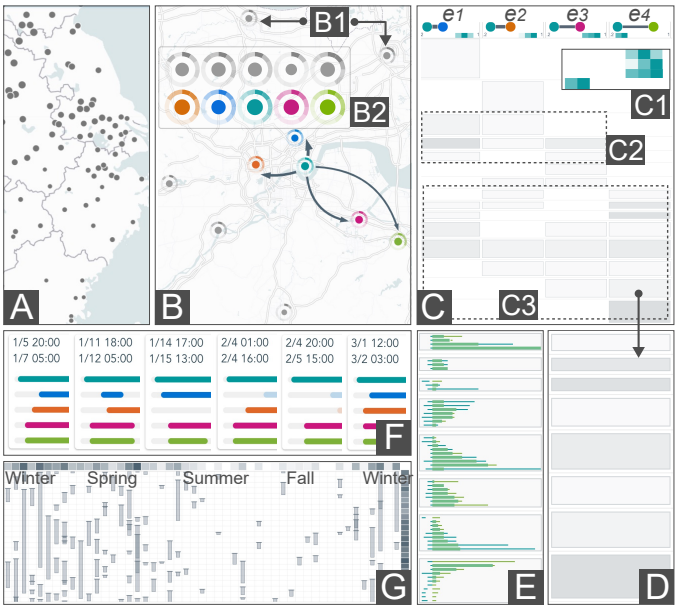

Fig. 8: Cascade analysis of air pollution. (A) The spatial distribution of the locations in eastern China. (B) The locations in Hangzhou and the inferred cascading pattern. (C) The influences in the pattern in (B). (D) The uncertainty and (E) influence visualizations for the influences of $e4$. (F) The cascading cards of the cascades involving $e3$ and $e4$. (G) The temporal chart of the pattern in (B).

in southern China, where Hangzhou is located [36]. Two actions were taken to improve the situation: 1) in 2016, Hangzhou enforced strict air pollution control policies prior to the G20 summit and achieved significant progress in improving air quality; 2) since 2018, local authorities in northern China have started to reform the industry, e.g., by using gas instead of burning coal. Thus, the experts aimed to determine the major pollution source that led to air pollution in Hangzhou during 2018.

Dataset and Processing. An air pollution event is detected if the $PM_{2.5}$ concentration is larger than $75\,\mu g\,m^{-3}$ (Sec. 4.1). Our dataset comprises the hourly readings of the $PM_{2.5}$ concentration from 482 major air monitoring stations in China during 2018. 482 infection event series of air pollution were extracted and then fed into VisCas.

Location Navigation. The experts first zoomed into eastern China. The decrease of dot sizes from north to south indicated that pollution in the north is more severe than that in the south (Fig. 8A). ED pointed out that this spatial distribution was contributed by "*the different industrial structures and climates between north and south and the pollutants propagating from north to south.*" The experts then obtained the event distributions over 12 months in 2018 for the locations in Hangzhou by zooming in (Fig. 8B2). The presence of dark sectors suggested that the air pollution events mainly occurred from winter to spring, especially in January. ED explained that "*such a seasonal characteristic might be because high humidity facilitates the formation of $PM_{2.5}$ via nucleation, and there is little rain to clean air pollution.*"

Examining the spatiotemporal summary alone cannot justify the cause of air pollution. Hence, the experts set t_w to 6 hours based on their domain knowledge and obtained a

star-shaped cascading pattern (Fig. 8B) This pattern showed that air pollution first occurred at the center of Hangzhou and then spread out. Therefore, the main cause of air pollution in Hangzhou was likely local generation rather than propagation from the north; otherwise, the network would begin with two locations in the north (Fig. 8B1).

Influence Inspection. The experts analyzed the influences comprised in the pattern in the influence view to understand the pattern thoroughly (Fig. 8C). The four edges tended to co-occur, such as $e1$ and $e2$-3 (Fig. 8C2) and $e4$ and $e1$-3 (Fig. 8C3). The experts explained that, due to the dynamic wind field, the central location's pollution usually influenced many other locations but not always all of the locations were. ED hypothesized that "*air deterioration in this region might be caused by the diffusion processes started from the cyan (central) location*." EC also commented that "*massive cascades and influences were well summarized by VisCas*."

Experts also noticed that the influences of $e4$ had lower likelihoods (Fig. 8C1) than those of $e1$-$e3$ (Fig. 8C1) Moreover, the presence of some dark cells in $e4$ indicated high uncertainties (Fig. 8D). Thus, the experts considered $e4$ unreliable. Nonetheless, the severe overlapping in Fig. 8E showed that the infection situations in the two locations involved in $e4$ were highly similar and thus had strong dependencies. Such an inconsistency confused the experts.

EA believed that the long distance between the cyan and green locations resulted in the low likelihoods. The experts also speculated that the pollutants of the cyan location were first transported to the purple, the geographically central location of $e4$, and then to the green. To figure out this situation, the experts accessed the cascades infecting both of the purple and green locations by selecting $e3$ and $e4$. They skimmed over the cascading cards and found many cascades showing that the purple and green locations got infected simultaneously (Fig. 8F). EA suggested that the sensors might not have sufficient time granularity to record the processes where the purple and green locations got infected successively, and thus their dependency relationship was missed. ED concluded that the cyan location's pollutants could spread to the areas around Hangzhou, including the distant southeast region. EC added, "*the uncertainty visualizations enable analyzing cascades and influences confidently*."

Cascading Exploration. In the temporal chart, the experts found that the cascades of the pattern mostly occurred in winter and spring (Fig. 8G). The cascades during these periods lasted an exceptionally long time, indicated by the bars' lengths. EC spoke highly of VisCas, stating that "*VisCas allows analyzing the cascades occurring on an irregular basis that cannot be handled before*." EB commented that the effects of these cascades were severe and required urgent attention. Effective policies were also recommended to alleviate pollution in Hangzhou: 1) control the local emissions, particularly during winter and spring; 2) derive early warning strategies based on the pollution events at the central location.

Conclusion. The experts learned through investigation that the center location was a key factor contributing to the pollution around Hangzhou. The largest industrial park in Hangzhou and the largest logistics park in Zhejiang Province were located at the center location. This case demonstrated that the system could help identify the pollution sources and diffusion processes that harmed the environment. Our visualizations were also proven effective in alleviating the limitation of data granularity.

7.3 Expert Interview

We conducted informal one-on-one interviews with the experts and summarized their feedback after the case studies.

Visual Design. All experts confirmed that VisCas could present massive cascades with the well-designed and coordinated visualizations. EA and EB commented that "*the cascades can be intuitively presented through the timeline-based influence visualizations and cascading cards*," and "*the three major views support the workflow of multifaced cascade analyses*."

Usability. All experts praised the usability of VisCas. The combination of the automated algorithm and visualizations facilitates the discovery of the inherent relationships among locations. ED mentioned, "*Finding the air pollution source originally is a time-consuming and labor-intensive task*." Moreover, the cascading pattern in the first case study revealed the real traffic deterioration at that time. EA claimed that "*feeding the latest traffic data can understand and improve the present city*."

Suggestions. The experts also made some suggestions for VisCas. Although the system effectively solves the general spatial cascade analyses, domain-specific details need to be presented to improve reasoning when applied to a specific domain. For example, EB commented that "*it is suggested to display the vehicles' trajectories as a supplement*" when analyzing traffic congestion. Additional attributes were also suggested to be included to capture the latent influence further. "*Other relevant attributes such as PM_{10} can be considered to model air pollution events*," said EA.

8 Discussion

This section discusses the implications, lessons learned, generalizability, and limitations of the proposed system.

Implications. Our work has many implications on numerous applications from various domains. First, VisCas helps experts identify spatial cascades and study inherent relationships among locations to analyze urban deterioration with big data, which avoids time-consuming on-site investigations. For example, the traffic cascading pattern discovered in the first case study can improve local authorities' understanding of complex traffic congestion and assist them in informed policy-making. Second, VisCas supports the analysis of complex cascades based only on Boolean events. This can be particularly useful in real-world applications because rich and complete data are sometimes unavailable for many reasons, such as privacy issues and limited monitoring capabilities. Third, VisCas is applicable to large-scale urban data because 1) the proposed network inference algorithm (Appendix A) is adequately fast to operate on a large dataset in real time and 2) the proposed visualizations can accommodate numerous cascades because of the compact glyph design.

Lessons Learned. We learn two design lessons through this design study. First, optimization algorithms can improve the readability of visual designs. We formulate the design of the influence view as the Hamiltonian path problem to obtain readable visual representations (Fig. 5D). Generating visualizations with automatic optimization effectively addresses the design issues, such as visual clutters

and occlusions caused by massive data. Second, interactive mining can enhance exploratory analyses. Without useful interactions, analysts may suffer from the tedious trial-and-error process when searching for the correct area to obtain meaningful patterns. By contrast, interactive mining provides analysts with fluent exploratory analyses, enabling a human-in-the-loop workflow. Moreover, heavy precalculation with different parameters is unnecessary due to the on-demand mining process.

Generalizability. The case studies for traffic congestion and air pollution have demonstrated that our approach can be easily generalized to a number of scenarios. First, the proposed network inference technique is application-independent and can be applied to any spatiotemporal events. Second, the proposed system is flexible because it is developed by abstracting the user requirements from multiple domains. Domain-specific views based on application scenarios can be added to improve users' understanding of cascades in a different context and further improve the usability of our approach.

Limitations. Our work has three limitations. The first one lies in influence modeling. We only consider a single attribute in this study. Modeling with multiple attributes [63] or even cross-domain spatiotemporal events [38] can potentially improve the performance of the proposed model. We plan to incorporate multiple attributes and extend the model to analyze cross-domain cascades in the future.

The second limitation is that the non-trivial designs may be too complicated for average users to understand. VisCas adopts novel designs to support challenging analytical tasks. Although the experts can easily understand these visual designs, government officials may experience difficulty in gaining insights due to their lack of expertise in data analysis We will study how to better convey insights gained from visual analytics in the future [5].

The third limitation lies in the specification of the spatial and temporal constraints. For the spatial constraints, if users perform an inappropriate area selection, then important locations can be missed in the cascading network. This issue can potentially be addressed with automatic space division and area recommendation, and the model can be further improved to adjust selection boundaries adaptively. For the temporal constraints, small time windows may lead to the loss of some important events and locations. To avoid this issue, analysts can specify a slightly large window because the monotonic exponential model will assign low weights to the events that have occurred for a long time.

9 Conclusion

In this work, we systematically study the cascades of spatial contagions via visual analytics. We first adapt a network inference model to infer the cascading network of spatial contagions for general urban scenarios. We then combine this model with well-designed visualizations to develop a novel visual analytics system called VisCas. VisCas supports an analytical workflow for multi-faceted spatial cascade analyses, including location navigation, influence inspection, and cascading exploration. We demonstrate the effectiveness of our approach with two case studies conducted on real-world datasets and also discuss the valuable implications and high generalizability of our approach.

Acknowledgment

We thank Editor and all reviewers for their constructive comments. We also thank Huachang Yu for his contribution to the system development. The work was supported by National Natural Science Foundation of China (62072400, 61822701), Zhejiang Provincial Natural Science Foundation (LR18F020001), and the 100 Talents Program of Zhejiang University.

References

[1] P. Accorsi, N. Lalande, M. Fabrègue, A. Braud, P. Poncelet, A. Sallaberry, S. Bringay, M. Teisseire, F. Cernesson, and F. L. Ber. HydroQual: Visual analysis of river water quality. In *Proc. of IEEE VAST*, pages 123–132, 2014.

[2] N. V. Andrienko, G. L. Andrienko, J. M. C. Garcia, and D. Scarlatti. Analysis of flight variability: a systematic approach. *IEEE TVCG*, 25(1):54–64, 2019.

[3] D. J. Berndt and J. Clifford. Using dynamic time warping to find patterns in time series. In *Proc. of AAAI Workshop*, pages 359–370, 1994.

[4] J. Chae, D. Thom, H. Bosch, Y. Jang, R. Maciejewski, D. S. Ebert, and T. Ertl. Spatiotemporal social media analytics for abnormal event detection and examination using seasonal-trend decomposition. In *Proc. of IEEE VAST*, pages 143–152, 2012.

[5] S. Chen, J. Li, G. L. Andrienko, N. V. Andrienko, Y. Wang, P. H. Nguyen, and C. Turkay. Supporting story synthesis: Bridging the gap between visual analytics and storytelling. *IEEE TVCG.*, 26(7):2499–2516, 2020.

[6] S. Chen, X. Yuan, Z. Wang, C. Guo, J. Liang, Z. Wang, X. L. Zhang, and J. Zhang. Interactive visual discovering of movement patterns from sparsely sampled geo-tagged social media data. *IEEE TVCG*, 22(1):270–279, 2016.

[7] W. Chen, F. Guo, and F. Wang. A survey of traffic data visualization. *IEEE TITS*, 16(6):2970–2984, 2015.

[8] Z. Chen, Y. Yang, L. Huang, E. Wang, and D. Li. Discovering urban traffic congestion propagation patterns with taxi trajectory data. *IEEE Access*, 6:69481–69491, 2018.

[9] F. Chierichetti, R. Kumar, S. Pandey, and S. Vassilvitskii. Finding the jaccard median. In *Proc. of SODA*, pages 293–311, 2010.

[10] C. F. Daganzo. The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research Part B: Methodological*, 28(4):269–287, 1994.

[11] Z. Deng, D. Weng, J. Chen, R. Liu, Z. Wang, J. Bao, Y. Zheng, and Y. Wu. AirVis: Visual analytics of air pollution propagation. *IEEE TVCG*, 26(1):800–810, 2020.

[12] H. Doraiswamy, N. Ferreira, T. Damoulas, J. Freire, and C. T. Silva. Using topological analysis to support event-guided exploration in urban data. *IEEE TVCG*, 20(12):2634–2643, 2014.

[13] N. Ferreira, J. Poco, H. T. Vo, J. Freire, and C. T. Silva. Visual exploration of big spatio-temporal urban data: A study of new york city taxi trips. *IEEE TVCG*, 19(12):2149–2158, 2013.

[14] M. Ghoniem, J. Fekete, and P. Castagliola. A comparison of the readability of graphs using node-link and matrix-based representations. In *Proc. of IEEE Symp. on Information Visualization*, pages 17–24, 2004.

[15] M. Gomez-Rodriguez, J. Leskovec, and A. Krause. Inferring networks of diffusion and influence. *ACM TKDD*, 5(4):21:1–21:37, 2012.

[16] S. Goodwin, J. Dykes, S. Jones, I. Dillingham, G. Dove, A. Duffy, A. Kachkaev, A. Slingsby, and J. Wood. Creative user-centered visualization design for energy analysts and modelers. *IEEE TVCG*, 19(12):2516–2525, 2013.

[17] S. Guo, Z. Jin, D. Gotz, F. Du, H. Zha, and N. Cao. Visual progression analysis of event sequence data. *IEEE TVCG*, 25(1):417–426, 2019.

[18] A. H. Halim and I. Ismail. Combinatorial optimization: comparison of heuristic algorithms in travelling salesman problem. *Archives of Computational Methods in Engineering*, 26(2):367–380, 2019.

[19] Z. Huang, Y. Zhao, W. Chen, S. Gao, K. Yu, W. Xu, M. Tang, M. Zhu, and M. Xu. A natural-language-based visual query approach of uncertain human trajectories. *IEEE TVCG*, 26(1):1256–1266, 2020.

[20] S. Ioffe. Improved consistent sampling, weighted minhash and L1 sketching. In *Proc. of ICDM*, pages 246–255, 2010.

[21] T. Kapler and W. Wright. Geotime information visualization. *Information Visualization*, 4(2):136–146, 2005.

[22] C. D. Laird, L. T. Biegler, B. G. van Bloemen Waanders, and R. A. Bartlett. Contamination source determination for water networks. *Journal of Water Resources Planning and Management*, 131(2):125–134, 2005.

[23] C. Lee, Y. Kim, S. Jin, D. Kim, R. Maciejewski, D. S. Ebert, and S. Ko. A visual analytics system for exploring, monitoring, and forecasting road traffic congestion. *IEEE TVCG*, 26(11):3133–3146, 2020.

[24] T. Lee and H. Shen. Visualization and exploration of temporal trend relationships in multivariate time-varying data. *IEEE TVCG*, 15(6):1359–1366, 2009.

[25] J. Leskovec, M. McGlohon, C. Faloutsos, N. S. Glance, and M. Hurst. Patterns of cascading behavior in large blog graphs. In *Proc. of SIAM SDM07*, pages 551–556, 2007.

[26] J. Li, S. Chen, K. Zhang, G. L. Andrienko, and N. V. Andrienko. COPE: Interactive exploration of co-occurrence patterns in spatial time series. *IEEE TVCG*, 25(8):2554–2567, 2019.

[27] X. Li, Y. Cheng, G. Cong, and L. Chen. Discovering pollution sources and propagation patterns in urban area. In *Proc. of ACM SIGKDD*, pages 1863–1872, 2017.

[28] Y. Liang, Z. Jiang, and Y. Zheng. Inferring traffic cascading patterns. In *Proc. of ACM SIGSPATIAL*, pages 2:1–2:10, 2017.

[29] R. G. Little. Managing the risk of cascading failure in complex urban infrastructures. In *Disrupted Cities*, pages 39–52. 2010.

[30] D. Liu, D. Weng, Y. Li, J. Bao, Y. Zheng, H. Qu, and Y. Wu. SmartAdP: Visual analytics of large-scale taxi trajectories for selecting billboard locations. *IEEE TVCG*, 23(1):1–10, 2017.

[31] D. Liu, P. Xu, and L. Ren. TPFlow: Progressive partition and multidimensional pattern extraction for large-scale spatio-temporal data analysis. *IEEE TVCG*, 25(1):1–11, 2019.

[32] H. Liu, Y. Gao, L. Lu, S. Liu, H. Qu, and L. M. Ni. Visual analysis of route diversity. In *Proc. of IEEE VAST*, pages 171–180, 2011.

[33] W. Liu, Y. Zheng, S. Chawla, J. Yuan, and X. Xie. Discovering spatio-temporal causal interactions in traffic data streams. In *Proc. of ACM SIGKDD*, pages 1010–1018, 2011.

[34] J. Long, Z. Gao, H. Ren, and A. Lian. Urban traffic congestion propagation and bottleneck identification. *Science in China Series F: Information Sciences*, 51(7):948, 2008.

[35] J. Long, Z. Gao, X. Zhao, A. Lian, and P. Orenstein. Urban traffic jam simulation based on the cell transmission model. *Networks and Spatial Economics*, 11(1):43–64, 2011.

[36] T. Ma, F. Duan, K. He, Y. Qin, D. Tong, G. Geng, X. Liu, H. Li, S. Yang, S. Ye, et al. Air pollution characteristics and their relationship with emissions and meteorology in the yangtze river delta region during 2014–2016. *Journal of Environmental Sciences*, 83:8–20, 2019.

[37] A. Malik, R. Maciejewski, N. Elmqvist, Y. Jang, D. S. Ebert, and W. Huang. A correlative analysis process in a visual analytics environment. In *Proc. of IEEE VAST*, pages 33–42, 2012.

[38] S. Mittelstädt, X. Wang, T. Eaglin, D. Thom, D. A. Keim, W. J. Tolone, and W. Ribarsky. An integrated in-situ approach to impacts from natural disasters on critical infrastructures. In *Proc. of HICSS*, pages 1118–1127, 2015.

[39] T. Munzner. A nested process model for visualization design and validation. *IEEE TVCG.*, 15(6):921–928, 2009.

[40] W.-S. Ng, L. Schipper, and Y. Chen. China motorization trends: New directions for crowded cities. *Journal of Transport and Land Use*, 3(3):5–25, 2010.

[41] H. Nguyen, W. Liu, and F. Chen. Discovering congestion propagation patterns in spatio-temporal traffic data. *IEEE TBD*, 3(2):169–180, 2017.

[42] B. Ni, Q. Shen, J. Xu, and H. Qu. Spatio-temporal flow maps for visualizing movement and contact patterns. *Visusal Informatics*, 1(1):57–64, 2017.

[43] W. H. Organization. *Air quality guidelines: global update 2005: particulate matter, ozone, nitrogen dioxide, and sulfur dioxide*. World Health Organization, 2006.

[44] C. Palomo, Z. Guo, C. T. Silva, and J. Freire. Visually exploring transportation schedules. *IEEE TVCG*, 22(1):170–179, 2016.

[45] M. Pi, H. Yeon, H. Son, and Y. Jang. Visual cause analytics for traffic congestion. *IEEE TVCG*, pages 1–1, 2019.

[46] J. Pu, S. Liu, Y. Ding, H. Qu, and L. M. Ni. T-Watcher: A new visual analytic system for effective traffic surveillance. In *Proc. of IEEE MDM*, pages 127–136, 2013.

[47] H. Qu, W. Chan, A. Xu, K. Chung, A. K. Lau, and P. Guo. Visual analysis of the air pollution problem in hong kong. *IEEE TVCG*, 13(6):1408–1415, 2007.

[48] M. S. Rahman and M. Kaykobad. On hamiltonian cycles and hamiltonian paths. *Inf. Process. Lett.*, 94(1):37–41, 2005.

[49] C. Rego, D. Gamboa, F. Glover, and C. Osterman. Traveling salesman problem heuristics: Leading methods, implementations and latest advances. *European Journal of Operational Research*, 211(3):427–441, 2011.

[50] Q. Shen, W. Zeng, Y. Ye, S. M. Arisona, S. Schubiger, R. Burkhard, and H. Qu. StreetVizor: Visual exploration of human-scale urban forms based on street views. *IEEE TVCG*, 24(1):1004–1013, 2018.

[51] R. Sitzenfrei, M. Mair, M. Möderl, and W. Rauch. Cascade vulnerability for risk analysis of water infrastructure. *Water science and technology : a journal of the International Association on Water Pollution Research*, 64(9):1885–1891, 2011.

[52] A. Stein, R. R. Draxler, G. D. Rolph, B. J. Stunder, M. Cohen, and F. Ngan. Noaa's hysplit atmospheric transport and dispersion modeling system. *Bulletin of the American Meteorological Society*, 96(12):2059–2077, 2015.

[53] T. Stützle, A. Grün, S. Linke, and M. Rüttger. A comparison of nature inspired heuristics on the traveling salesman problem. In *Proc. of Parallel Problem Solving from Nature*, volume 1917, pages 661–670, 2000.

[54] G. Sun, R. Liang, H. Qu, and Y. Wu. Embedding spatio-temporal information into maps by route-zooming. *IEEE TVCG*, 23(5):1506–1519, 2017.

[55] R. J. M. Vaessens, E. H. L. Aarts, and J. K. Lenstra. Job shop scheduling by local search. *INFORMS Journal on Computing*, 8(3):302–317, 1996.

[56] T. von Landesberger, F. Brodkorb, P. Roskosch, N. V. Andrienko, G. L. Andrienko, and A. Kerren. MobilityGraphs: Visual analysis of mass mobility dynamics via spatio-temporal graphs and clustering. *IEEE TVCG*, 22(1):11–20, 2016.

[57] H. Wang, Y. Lu, S. T. Shutters, M. Steptoe, F. Wang, S. Landis, and R. Maciejewski. A visual analytics framework for spatiotemporal trade network analysis. *IEEE TVCG*, 25(1):331–341, 2019.

[58] Z. Wang, M. Lu, X. Yuan, J. Zhang, and H. van de Wetering. Visual traffic jam analysis based on trajectory data. *IEEE TVCG*, 19(12):2159–2168, 2013.

[59] D. Weng, R. Chen, Z. Deng, F. Wu, J. Chen, and Y. Wu. SRVis: Towards better spatial integration in ranking visualization. *IEEE TVCG*, 25(1):459–469, 2019.

[60] D. Weng, R. Chen, J. Zhang, J. Bao, Y. Zheng, and Y. Wu. Pareto-optimal transit route planning with multi-objective monte-carlo tree search. *IEEE TIST*, 22(2):1185–1195, 2021.

[61] D. Weng, C. Zheng, Z. Deng, M. Ma, J. Bao, Y. Zheng, M. Xu, and Y. Wu. Towards better bus networks: A visual analytics approach. *IEEE TVCG*, 27(2):817–827, 2021.

[62] D. Weng, H. Zhu, J. Bao, Y. Zheng, and Y. Wu. Homefinder Revisited: Finding ideal homes with reachability-centric multi-criteria decision making. In *Proc. of ACM SIGCHI*, page 247, 2018.

[63] J. Wu, Z. Guo, Z. Wang, Q. Xu, and Y. Wu. Visual analytics of multivariate event sequence data in racquet sports. In *Proc. of IEEE VAST*, pages 36–47, 2020.

[64] Y. Wu, X. Xie, J. Wang, D. Deng, H. Liang, H. Zhang, S. Cheng, and W. Chen. ForVizor: Visualizing spatio-temporal team formations in soccer. *IEEE TVCG*, 25(1):65–75, 2019.

[65] X. Xie, J. Wang, H. Liang, D. Deng, S. Cheng, H. Zhang, W. Chen, and Y. Wu. PassVizor: Toward better understanding of the dynamics of soccer passes. *IEEE TVCG*, 27(2):1322–1331, 2021.

[66] H. Xiong, A. Vahedian, X. Zhou, Y. Li, and J. Luo. Predicting traffic congestion propagation patterns: A propagation graph approach. In *Proc. of ACM SIGSPATIAL*, pages 60–69, 2018.

[67] W. Zeng, C. Fu, S. M. Arisona, A. Erath, and H. Qu. Visualizing mobility of public transportation system. *IEEE TVCG*, 20(12):1833–1842, 2014.

[68] W. Zeng, Q. Shen, Y. Jiang, and A. Telea. Route-aware edge bundling for visualizing origin-destination trails in urban traffic. *CGF*, 38(3):581–593, 2019.

[69] A. Zhang and Z. Gao. Effect of atis information under incident-based congestion propagation. *Procedia-Social and Behavioral Sciences*, 43:628–637, 2012.

[70] H. Zhang and A. J. Khattak. Analysis of cascading incident event durations on urban freeways. *Transportation Research Record*, 2178(1):30–39, 2010.
[71] Y. Zhao, H. Jiang, Q. Chen, Y. Qin, H. Xie, Y. Wu, S. Liu, Z. Zhou, J. Xia, and F. Zhou. Preserving minority structures in graph sampling. *IEEE TVCG*, 27(2):1698–1708, 2021.
[72] Y. Zheng, L. Capra, O. Wolfson, and H. Yang. Urban Computing: Concepts, methodologies, and applications. *ACM TIST*, 5(3):38:1–38:55, 2014.
[73] Y. Zheng, W. Wu, Y. Chen, H. Qu, and L. M. Ni. Visual analytics in urban computing: An overview. *IEEE TBD*, 2(3):276–296, 2016.
[74] J. Y. Zhu, C. Zhang, H. Zhang, S. Zhi, V. O. K. Li, J. Han, and Y. Zheng. pg-Causality: Identifying spatiotemporal causal pathways for air pollutants with urban big data. *IEEE TBD*, 4(4):571–585, 2018.
[75] E. Zio and G. Sansavini. Component criticality in failure cascade processes of network systems. *Risk Analysis: An International Journal*, 31(8):1196–1210, 2011.

Zikun Deng received his B.S. degree in Transportation Engineering from Sun Yat-Sen University in 2016. He is currently pursuing the doctoral degree with the State Key Lab of CAD&CG, Zhejiang University. His research interests mainly include spatiotemporal data mining, visualization, and urban visual analytics. For more information, please visit https://zkdeng.org.

Di Weng received his B.S. degree in Computer Science from Taishan Honored College, Shandong University in 2016. He is currently pursuing the doctoral degree with the State Key Lab of CAD&CG, Zhejiang University. His research interests mainly include the data mining, visualization, and visual analytics of large-scale urban data. For more information, please visit https://dweng.org.

Yuxuan Liang is now a PhD student at School of Computing, National University of Singapore. He has published over twenty papers in refereed journals and conferences, such as TKDE, KDD, NeurIPS, IJCAI, AAAI, UbiComp and SIGSPATIAL. His research interests lie in urban computing and graph machine learning.

Dr. Jie Bao got his Ph.D degree in Computer Science from University of Minnesota at Twin Cities in 2014. He worked as a researcher in Urban Computing Group at MSR Asia from 2014 to 2017. He currently leads the Data Platform Division in JD Urban Computing Business Unit. His research interests include: Spatio-temporal Data Management/Mining, Urban Computing, and Location-based Services.

Dr. Yu Zheng is a Vice President and Chief Data Scientist at JD Tech., passionate about using big data and AI technology to tackle urban challenges. His research interests include big data analytics, spatio-temporal data mining, machine learning, and artificial intelligence. He also leads the JD Urban Computing Business Unit as the president and serves as the director of the JD Intelligent City Research. Before joining JD, he was a senior research manager at Microsoft Research. Zheng is also a Chair Professor at Shanghai Jiao Tong University, an Adjunct Professor at Hong Kong University of Science and Technology.

Dr. Tobias Schreck is a Professor and head of the Institute of Computer Graphics and Knowledge Visualization at Graz University of Technology. He previously was Assistant Professor with University of Konstanz, Germany, and Postdoc Fellow with Technische Universität Darmstadt. He obtained a PhD in Computer Science in 2006 from the University of Konstanz. Tobias Schreck works in the areas of Visual Analytics, Information Visualization, and Applied 3D Object Retrieval.

Dr. Mingliang Xu is a professor in the School of Information Engineering of Zhengzhou University, China, and currently is the director of CIISR (Center for Interdisciplinary Information Science Research) and the vice General Secretary of ACM SIGAI China. He received his Ph.D. degree in computer science and technology from the State Key Lab of CAD&CG at Zhejiang University, Hangzhou, China. His current research interests include computer graphics and artificial intelligence. He has authored more than 80 journal and conference papers in these areas, including ACM TOG, ACM TIST, IEEE TPAMI, IEEE TIP, IEEE TCYB, IEEE TCSVT, IEEE TAC, IEEE TCIAIG, ACM SIGGRAPH (Asia), ACM MM, IJCAI, etc.

Dr. Yingcai Wu is a Professor at the State Key Lab of CAD&CG, Zhejiang University. His main research interests are in information visualization and visual analytics, with focuses on urban computing, sports science, immersive visualization, and narrative visualization. He received his Ph.D. degree in Computer Science from the Hong Kong University of Science and Technology. Prior to his current position, Dr. Wu was a postdoctoral researcher in the University of California, Davis from 2010 to 2012, a researcher in Microsoft Research Asia from 2012 to 2015, and a ZJU100 Young Professor at Zhejiang University from 2015 to 2020. For more information, please visit http://www.ycwu.org.