Large-scale Geospatial Analytics: Problems, Challenges, and Opportunities

Tsz Nam Chan
Hong Kong Baptist University
edisonchan@comp.hkbu.edu.hk

Leong Hou U
University of Macau
ryanlhu@um.edu.mo

Byron Choi
Hong Kong Baptist University
bchoi@comp.hkbu.edu.hk

Jianliang Xu
Hong Kong Baptist University
xujl@comp.hkbu.edu.hk

Reynold Cheng
The University of Hong Kong
Guangdong-Hong Kong-Macau Joint
Laboratory for Smart Cities
ckcheng@cs.hku.hk

ABSTRACT

Geospatial analytics is an important field in many communities, including crime science, transportation science, epidemiology, ecology, and urban planning. However, with the rapid growth of big geospatial data, most of the commonly used geospatial analytic tools are not efficient (or even feasible) to support large-scale datasets. As such, domain experts have raised the concerns about the inefficiency issues for using these tools. In this tutorial, we aim to arouse the attention of database researchers for this important, emerging, database-related, and interdisciplinary topic, which consists of four parts. In the first part, we will discuss different problems and highlight the challenges for two types of geospatial analytic tools, which are (1) hotspot detection and (2) correlation analysis. In the second and third parts, we will specifically discuss two geospatial analytic tools, namely kernel density visualization (the representative hotspot detection method) and K-function (the representative correlation analysis method), respectively, and their variants. In the fourth part, we will highlight the future opportunities for this topic.

CCS CONCEPTS

• **Theory of computation → Computational geometry**; • **Information systems → Geographic information systems**; • **Human-centered computing → Heat maps**.

KEYWORDS

Geospatial analytics, GIS, kernel density visualization, K-function, efficient algorithm and software development

ACM Reference Format:
Tsz Nam Chan, Leong Hou U, Byron Choi, Jianliang Xu, and Reynold Cheng. 2023. Large-scale Geospatial Analytics: Problems, Challenges, and Opportunities. In *Companion of the 2023 International Conference on Management of Data (SIGMOD-Companion '23), June 18–23, 2023, Seattle, WA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3555041.3589401

SIGMOD-Companion '23, June 18–23, 2023, Seattle, WA, USA

ACM ISBN 978-1-4503-9507-6/23/06…$15.00
https://doi.org/10.1145/3555041.3589401

1 INTRODUCTION

Geospatial analytics is an important field in many disciplines. Some representative examples include criminology, transportation science, epidemiology, ecology, and urban planning. Criminologists and transportation scientists [24, 57, 65, 69, 82–84, 95, 97, 102] need to discover crime and traffic accident hotspots, respectively, in different geographical regions. Epidemiologists [39, 41, 42, 46, 54, 55, 58, 80] need to detect disease outbreaks, identify transmission patterns of different diseases, and analyze disease factors. Ecologists [54, 80, 87] need to understand the distribution of environmental incidents (e.g., air pollution). Urban planners [45, 89, 98] need to analyze human mobility in different cities. As such, many off-the-shelf software packages, e.g., QGIS [11], ArcGIS [1], CrimeStat [5], spatstat [14, 19], spNetwork [15], and SANET [13, 73], have been developed to support geospatial analytics.

However, in the big data era, many large-scale location datasets can be collected and analyzed nowadays. For example, the Chicago crime dataset [3] and New York taxi dataset [9] contain 7.68 million and 165 million data points, respectively. Worse yet, many commonly used tools in geospatial analytics (e.g., kernel density visualization (KDV) [32, 57], K-function [33, 74, 106], and spatial clustering [18, 88]) suffer from high time complexity (e.g., $O(n^2)$ time for computing a single K-function, where n denotes the number of data points). Based on the above reasons, these tools cannot be efficiently (or even feasibly) supported by off-the-shelf software packages, which have been also complained by many domain experts [50, 55, 106].

As such, efficient algorithm and software development for these geospatial analytic tools is **an important, emerging, database-related, and interdisciplinary topic.** Although many tutorials that are related to spatial/spatiotemporal databases and data visualization have been given in the database community [38, 43, 63, 70, 72, 85, 90, 103, 113], none of these tutorials has focused on improving the efficiency for these geospatial analytic tools. Therefore, we propose this tutorial in order to arouse the attention of database researchers and practitioners for understanding different problems, challenges, and opportunities of this important topic. In particular, we will also discuss the state-of-the-art solutions for two commonly used tools, namely kernel density visualization (KDV) and K-function, in order to provide insights for tackling these geospatial analytic problems.

Target audience: In this tutorial, we mainly target the SIGMOD attendees who are interested in conducting research for spatial/spatiotemporal databases and data analytics or interested in incorporating latest technologies into software. The audience needs to understand some basic database concepts, e.g., indexing. However, this tutorial is self-contained, which does not require prior knowledge of geographic information systems and data visualization.

Related work from authors: We have extensively conducted research on improving the efficiency of different geospatial analytic tools in recent years, including kernel density visualization (KDV) [25, 26, 31, 32, 34], network kernel density visualization (NKDV) [30], spatiotemporal kernel density visualization (STKDV) [27], and network K-function [33]. Moreover, we have developed the python software packages, LIBKDV [29] and PyNKDV [35], and the web-based demonstration system, KDV-Explorer [28]. Furthermore, two online hotspot visualization systems (based on our research studies), namely Hong Kong COVID-19 hotspot map [6] and Macau COVID-19 hotspot map [8], have been deployed for monitoring COVID-19 hotspots in Hong Kong and Macau, respectively.

2 TUTORIAL OUTLINE

The tutorial lasts for **1.5 hours**, which consists of four parts. In the first part **(30 minutes)**, we will have a comprehensive overview of different geospatial analytic tools, which are supported by famous software packages (e.g., ArcGIS and QGIS). In the second part **(25 minutes)** and third part **(15 minutes)**, we will review the state-of-the-art solutions for two commonly used tools, namely kernel density visualization (KDV) and K-function, respectively. Moreover, we will also discuss other variants of KDV and K-function in these two parts. In the fourth part **(20 minutes)**, we will discuss the future opportunities of this topic.

2.1 Overview of Geospatial Analytics

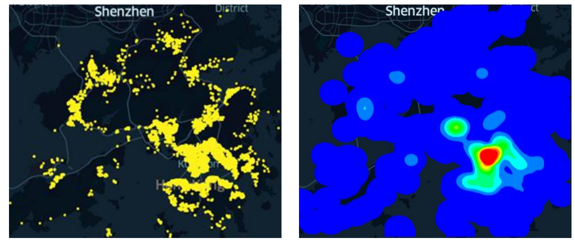
In the first part of this tutorial, we will focus on two application types of geospatial analytics, namely hotspot detection and correlation analysis, which are widely used by domain experts to analyze their location datasets. For each application type, we will discuss all famous tools in Table 1 by (1) illustrating the backgrounds of them (e.g., formulating them as spatial query processing problems), (2) providing a hands-on demonstration, using QGIS/ArcGIS, for showing how these tools can be used to analyze patterns in the Hong Kong COVID-19 dataset [7], (3) comparing the pros and cons for these tools, and (4) highlighting the challenges (i.e., the inefficiency issues) for using these tools.

Table 1: Different types of geospatial analytic tools.

Application type	Geospatial analytic tool	References
Hotspot detection	Kernel density visualization (KDV)	[44, 83, 95]
	Inverse distance weighting (IDW)	[16, 61, 104]
	Kriging	[92, 101, 112]
Correlation analysis	K-function	[22, 64, 108]
	Moran's I	[37, 60, 93]
	Getis-Ord General G	[17, 59, 62]

As an example, we provide the backgrounds of KDV (one of the hotspot detection methods) and K-function (one of the correlation analysis methods) in this section.

Background of KDV: To discover hotspots in a location dataset, domain experts need to generate a KDV-based heatmap. Figure 1 shows an example for discovering hotspots in the Hong Kong COVID-19 dataset. Note that the red region is the COVID-19 hotspot in Hong Kong.



(a) Hong Kong COVID-19 cases (b) Heatmap

Figure 1: A heatmap (based on KDV) for the Hong Kong COVID-19 dataset (yellow points in (a)), where we use the red color (in (b)) to denote the high-density (hotspot) region.

In Definition 1, we formally define the problem for generating KDV (cf. Figure 1).

Definition 1. *(KDV [32]) Given a location dataset* $P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_n\}$ *with n spatial data points and a geographical region with $X \times Y$ pixels, we need to color each pixel $\mathbf{q}$ based on the kernel density value $\mathcal{F}_P(\mathbf{q})$ (cf. Equation 1).*

$$\mathcal{F}_P(\mathbf{q}) = \sum_{\mathbf{p} \in P} w \cdot \mathcal{K}(\mathbf{q}, \mathbf{p}) \quad (1)$$

where w and $\mathcal{K}(\mathbf{q}, \mathbf{p})$ denote the normalization constant and kernel function, respectively. Some representative kernel functions are shown in Table 2.

Table 2: Some representative kernel functions, where *dist* and b denote the Euclidean distance and the bandwidth parameter, respectively.

Kernel	$\mathcal{K}(\mathbf{q}, \mathbf{p})$	References
Uniform	$\begin{cases} \frac{1}{b} & \text{if } dist(\mathbf{q}, \mathbf{p}) \leq b \\ 0 & \text{otherwise} \end{cases}$	[99]
Epanechnikov	$\begin{cases} 1 - \frac{1}{b^2} dist(\mathbf{q}, \mathbf{p})^2 & \text{if } dist(\mathbf{q}, \mathbf{p}) \leq b \\ 0 & \text{otherwise} \end{cases}$	[41, 57]
Quartic	$\begin{cases} \left(1 - \frac{1}{b^2} dist(\mathbf{q}, \mathbf{p})^2\right)^2 & \text{if } dist(\mathbf{q}, \mathbf{p}) \leq b \\ 0 & \text{otherwise} \end{cases}$	[23, 68]
Gaussian	$\exp\left(-\frac{1}{b^2} dist(\mathbf{q}, \mathbf{p})^2\right)$	[69, 95]

As a remark, this tutorial will also cover the backgrounds of other tools in hotspot detection (i.e., IDW and Kriging in Table 1).

Background of K-function: Although many hotspot detection methods (e.g., KDV) can identify hotspots in a location dataset, these approaches cannot determine the meaningfulness/significance of these hotspots. For example, we can obtain some "hotspot regions" in a randomly generated location dataset, which are not meaningful. To tackle this issue, domain experts adopt the correlation analysis (cf. Table 1) to analyze whether a location dataset exhibits the cluster property (or is merely random). Here, we formally define the K-function (cf. Definition 2) and the K-function plot (cf. Definition 3), which can be used to analyze the cluster property of a location dataset.

Definition 2. *(K-function [19]) Given a location dataset $P = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_n\}$ with n spatial data points and the spatial threshold s, the K-function for this dataset is:*

$$K_P(s) = \sum_{\mathbf{p}_i \in P} \sum_{\mathbf{p}_j \in P} \mathbb{I}(dist(\mathbf{p}_i, \mathbf{p}_j) \leq s) \quad (2)$$

where $\mathbb{I}$ denotes the indicator function.

$$\mathbb{I}(x) = \begin{cases} 1 & \text{if } x \text{ is true.} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

Definition 3. *(K-function plot [19]) Given a location dataset P, L randomly generated datasets (with the same size n), $R_1, R_2, ..., R_L$, and D spatial thresholds, $s_1, s_2, ..., s_D$, generating a K-function plot involves computing $K_P(s_d)$ (cf. Equation 2), $\mathcal{L}(s_d)$ (cf. Equation 4), and $\mathcal{U}(s_d)$ (cf. Equation 5) for each spatial threshold s_d ($1 \leq d \leq D$).*

$$\mathcal{L}(s_d) = \min(K_{R_1}(s_d), K_{R_2}(s_d), ..., K_{R_L}(s_d)) \quad (4)$$

$$\mathcal{U}(s_d) = \max(K_{R_1}(s_d), K_{R_2}(s_d), ..., K_{R_L}(s_d)) \quad (5)$$

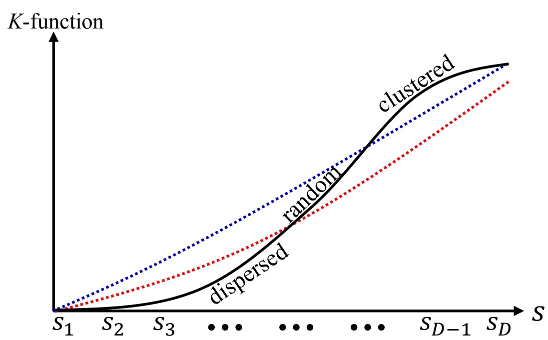


Figure 2: Illustration of a K-function plot, where the black line, red dotted line, and blue dotted line represent the curves of $K_P(s_d)$, $\mathcal{L}(s_d)$, and $\mathcal{U}(s_d)$, respectively.

Figure 2 shows an example of a K-function plot. Once the black curve $K_P(s_d)$ is above the blue dotted curve $\mathcal{U}(s_d)$, domain experts reckon that the dataset has meaningful clusters/hotspots for those thresholds s_d. Otherwise, they regard the dataset to be either random (i.e., the data points are randomly distributed.) or dispersed (i.e., the data points tend to be far away from each other.), which does not have meaningful clusters/hotspots for those thresholds s_d. Note that the parameter s_d in the clustered region (cf. Figure 2) can be further used in geospatial analytic tools of hotspot detection (e.g., using s_d as the bandwidth parameter b of a kernel function (cf. Table 2) to generate KDV (cf. Definition 1)).

In this tutorial, we will also cover the backgrounds of other tools in correlation analysis (i.e., Moran's I and Getis-Ord General G in Table 1).

2.2 Kernel Density Visualization and Its Variants

In the second part of this tutorial, we will review state-of-the-art solutions for generating KDVs. Next, we will illustrate other variants of KDV, including network kernel density visualization (NKDV) and spatiotemporal kernel density visualization (STKDV). After that, we will provide the case studies for adopting our COVID-19 hotspot maps [6, 8] to analyze COVID-19 hotspots in Hong Kong and Macau. Lastly, we will provide hands-on experience for using the fastest python library, LIBKDV [29], to support large-scale location datasets.

State-of-the-art solutions for generating KDVs: We will review four types of methods for improving the efficiency of generating KDVs, including (1) function approximation methods, (2) data sampling methods, (3) computational sharing methods, and (4) parallel/distributed and hardware-based methods. In addition, we will discuss the advantages and disadvantages of these methods.

Function approximation methods: In the first type of research studies, researchers [25, 31, 34, 47, 51] first develop the efficient lower and upper bound functions, $LB(\mathbf{q})$ and $UB(\mathbf{q})$, respectively, for the kernel density function $\mathcal{F}_P(\mathbf{q})$ (cf. Equation 1), i.e., $LB(\mathbf{q}) \leq \mathcal{F}_P(\mathbf{q}) \leq UB(\mathbf{q})$. Then, they incorporate these bound functions into an index structure (e.g., kd-tree [21] and ball-tree [71]) to progressively tighten $LB(\mathbf{q})$ and $UB(\mathbf{q})$ (by traversing the index structure) so that these bound values can achieve the approximation guarantee ε for computing the approximate kernel density value $R(\mathbf{q})$, where:

$$\frac{UB(\mathbf{q})}{LB(\mathbf{q})} \leq 1 + \varepsilon \rightarrow (1 - \varepsilon)\mathcal{F}_P(\mathbf{q}) \leq R(\mathbf{q}) \leq (1 + \varepsilon)\mathcal{F}_P(\mathbf{q}) \quad (6)$$

Data sampling methods: In the second type of research studies, researchers [77–79, 110, 111] propose to obtain the subset S of the dataset P. Then, they can compute the modified kernel density function $\mathcal{F}_S^{(M)}(\mathbf{q})$ for this subset S, where:

$$\mathcal{F}_S^{(M)}(\mathbf{q}) = \sum_{\mathbf{p}_i \in S} w_i \cdot \mathcal{K}(\mathbf{q}, \mathbf{p}_i) \quad (7)$$

They show that $\mathcal{F}_S^{(M)}(\mathbf{q})$ is theoretically close to the original kernel density value $\mathcal{F}_P(\mathbf{q})$ with a probabilistic guarantee. Since they can also provide the non-trivial upper bound for the subset size, computing $\mathcal{F}_S^{(M)}(\mathbf{q})$ can be significantly faster than $\mathcal{F}_P(\mathbf{q})$.

Computational sharing methods: In the third type of research studies, researchers [26, 29, 32, 52] exploit some sharing properties in order to improve the efficiency for computing a single KDV or multiple KDVs. Some of these research studies (e.g., [26, 29, 32]) can further reduce the time complexity for generating KDVs with non-trivial accuracy guarantees.

Parallel/distributed and hardware-based methods: In the fourth type of research studies, researchers propose to adopt (1) parallel/distributed approaches [29, 76, 86, 110] and (2) hardware-based approaches, including GPU [50, 67, 105, 107] and FPGA [50], to significantly boost the practical efficiency of generating KDV. Some of these research studies further combine these approaches with advanced methods, e.g., computational sharing method [29] and data sampling method [110].

Other variants of KDV: After we have discussed the state-of-the-art solutions of KDV, we will discuss two important variants of KDV, namely network kernel density visualization (NKDV) and spatiotemporal kernel density visualization (STKDV).

NKDV: Since some categories of geographical events, including traffic accidents and crime events, mainly occur in a road network, using the Euclidean distance $dist(\mathbf{q}, \mathbf{p})$ in the kernel function $\mathcal{K}(\mathbf{q}, \mathbf{p})$ (cf. Table 2) can overestimate the density value of each position (cf. Figure 3). Therefore, geographical researchers [96, 97] propose to replace $dist(\mathbf{q}, \mathbf{p})$ in $\mathcal{K}(\mathbf{q}, \mathbf{p})$ by the shortest path distance $dist_G(\mathbf{q}, \mathbf{p})$.

In this tutorial, we will also discuss this problem setting and review different methods for efficiently generating NKDV (e.g., [30, 81, 96]).

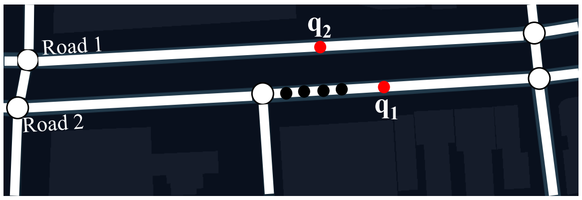
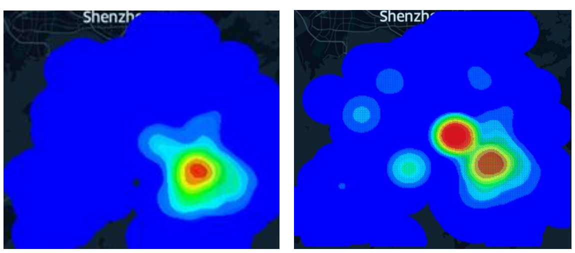


Figure 3: Although q_1 and q_2 are close to the black points (i.e., geographical events) in terms of the Euclidean distance, q_2 is far away from the black points in terms of the shortest path distance. As such, we should assign a smaller density value for q_2 compared with q_1.

<u>*STKDV:*</u> In practice, some geographical phenomena, e.g., the distribution of COVID-19 cases, significantly depend on the event time. Using the COVID-19 cases in Hong Kong (cf. Figure 4) as an example, note that there are two outbreak regions on January 2022, while there is only one outbreak region on December 2020. Therefore, the outbreak regions can change with respect to different timestamps. As such, geographical researchers propose to adopt STKDV [41, 57, 69]. In this tutorial, we will state this problem setting and discuss different methods [27, 86] for efficiently generating STKDV.



(a) December 2020 (b) January 2022

Figure 4: The distribution of COVID-19 cases in Hong Kong, generated by STKDV, depend on the wave/time. (Obtained from [29])

Case studies and hands-on experience: We will provide the case studies for analyzing Hong Kong and Macau COVID-19 hotspots using our COVID-19 hotspot maps [6, 8]. As an example, we show a snapshot of the Hong Kong COVID-19 hotspot map in Figure 5. Furthermore, we will also provide hands-on experience for using our fastest library, LIBKDV [29] (with a few lines of python code), to generate KDVs in the Hong Kong COVID-19 dataset.

2.3 K-function and Its Variants

In the third part of this tutorial, we will review state-of-the-art solutions for computing K-function and discuss different variants of K-function, including network K-function and spatiotemporal K-function.

State-of-the-art solutions for computing K-function. Compared with KDV (cf. Section 2.2), only a few of research studies focus on improving the efficiency for computing K-function, which can be divided into two classes, namely (1) range-query-based methods and (2) parallel/distributed and hardware-based methods. In this tutorial, we will also discuss the pros and cons of these methods.

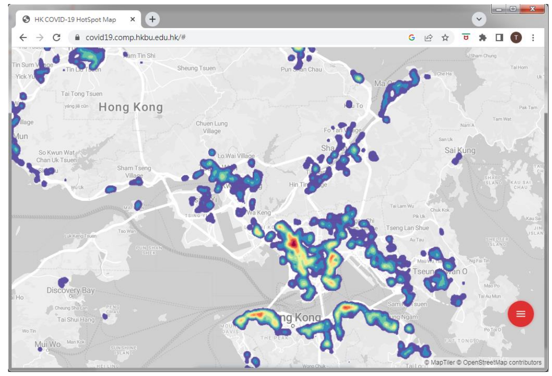


Figure 5: A snapshot of the Hong Kong COVID-19 hotspot map.

<u>*Range-query-based methods:*</u> Recall from Equation 2 that we need to count all data points $\mathbf{p}_j$ that are within the distance s from each data point $\mathbf{p}_i$ in order to compute the K-function. Therefore, one approach is to adopt some index structures, e.g., kd-tree [21], ball-tree [71], and range-tree [40], in order to efficiently obtain the range query set $R(\mathbf{p}_i)$ for each data point $\mathbf{p}_i$, where

$$R(\mathbf{p}_i) = \{\mathbf{p}_j \in P : dist(\mathbf{p}_i, \mathbf{p}_j) \leq s\}$$

Based on this set, K-function (cf. Equation 2) can be expressed as follows.

$$K_P(s) = \sum_{\mathbf{p}_i \in P} |R(\mathbf{p}_i)|$$

<u>*Parallel/distributed and hardware-based methods:*</u> In the geoscience community, researchers propose the parallel/distributed algorithms [106] and adopt the modern hardware, e.g., GPU [91], to improve the efficiency for computing K-function.

Other variants of K-function. Here, we discuss two variants of K-function, namely network K-function and spatiotemporal K-function.

<u>*Network K-function:*</u> Like NKDV (cf. Section 2.2), since many geographical events, e.g., traffic accidents, are mainly on/along with a road, using K-function (based on the Euclidean distance $dist(\mathbf{p}_i, \mathbf{p}_j)$) can overestimate the statistical results [100]. Using Figure 3 as an example, two points, $\mathbf{q}_1$ and $\mathbf{q}_2$, are close to each other in terms of the Euclidean distance can be far away from each other in terms of the shortest path distance. As such, geographical researchers [66, 73, 74, 100] propose the network K-function tool, which replaces the Euclidean distance $dist(\mathbf{p}_i, \mathbf{p}_j)$ by the shortest path distance $dist_G(\mathbf{p}_i, \mathbf{p}_j)$ in Equation 2. In this tutorial, we will discuss this problem setting and review different methods [33, 74, 81] for efficiently computing a network K-function and generating a network K-function plot (like Figure 2).

<u>*Spatiotemporal K-function:*</u> Like STKDV (cf. Section 2.2), some geographical phenomena, e.g., disease outbreak, significantly depend on event time (e.g., different waves). As such, using K-function, which does not consider the occurrence time of each event, may provide misleading analytic results. Therefore, domain experts [55, 56, 94] propose another tool, called spatiotemporal K-function $K_{\widehat{P}}(s, t)$ (cf. Equation 8), which simultaneously considers both the spatial threshold s and temporal threshold t, to analyze

a location dataset $\widehat{P} = \{(\mathbf{p}_1, t_{\mathbf{p}_1}), (\mathbf{p}_2, t_{\mathbf{p}_2}), ..., (\mathbf{p}_n, t_{\mathbf{p}_n})\}$ with n spatiotemporal data points.

$$K_{\widehat{P}}(s,t) = \sum_{(\mathbf{p}_i, t_{\mathbf{p}_i}) \in \widehat{P}} \sum_{(\mathbf{p}_j, t_{\mathbf{p}_j}) \in \widehat{P}} \mathbb{I}(dist(\mathbf{p}_i, \mathbf{p}_j) \leq s, dist(t_{\mathbf{p}_i}, t_{\mathbf{p}_j}) \leq t) \quad (8)$$

Instead of generating a two-dimensional K-function plot (cf. Figure 2), they generate a three-dimensional spatiotemporal K-function plot (cf. Figure 6). Note that the black surface, red surface, and blue surface denote $K_{\widehat{P}}(s_\alpha, t_\beta)$ (cf. Equation 8), $\mathcal{L}(s_\alpha, t_\beta)$ (cf. Equation 9), and $\mathcal{U}(s_\alpha, t_\beta)$ (cf. Equation 10), respectively, with M spatial thresholds ($1 \leq \alpha \leq M$) and T temporal thresholds ($1 \leq \beta \leq T$).

$$\mathcal{L}(s_\alpha, t_\beta) = \min(K_{\widehat{R}_1}(s_\alpha, t_\beta), K_{\widehat{R}_2}(s_\alpha, t_\beta), ..., K_{\widehat{R}_L}(s_\alpha, t_\beta)) \quad (9)$$

$$\mathcal{U}(s_\alpha, t_\beta) = \max(K_{\widehat{R}_1}(s_\alpha, t_\beta), K_{\widehat{R}_2}(s_\alpha, t_\beta), ..., K_{\widehat{R}_L}(s_\alpha, t_\beta)) \quad (10)$$

where $\widehat{R}_1$, $\widehat{R}_2$,..., $\widehat{R}_L$ are L randomly generated datasets with the same size n.

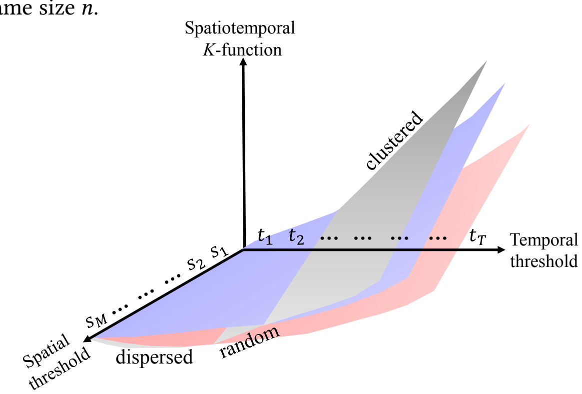


Figure 6: Illustration of a spatiotemporal K-function plot.

In this tutorial, we will state this problem setting and review different efficient methods (e.g., [55]) for computing a spatiotemporal K-function and generating a spatiotemporal K-function plot.

2.4 Future Opportunities

In the fourth part of this tutorial, we will discuss the future opportunities for both researchers and practitioners. In the following, we will highlight some of the promising directions.

Future opportunities for KDV and its variants: There are two main future research studies for this direction.

Optimal solutions for solving KDV, NKDV, and STKDV: Although many advanced algorithms have been proposed to reduce the time complexity for different variants of KDV (e.g., [32] for KDV, [30] for NKDV, and [27] for STKDV), these algorithms have not been proven to be optimal. We use KDV (cf. Definition 1) as an example. Recall that generating KDV needs to compute the kernel density function $\mathcal{F}_P(\mathbf{q})$ (cf. Equation 1) for each pixel $\mathbf{q}$. Therefore, every algorithm needs to at least access all (i.e., n) data points in P and all (i.e., $X \times Y$) pixels, which takes $\Omega(XY + n)$ time. However, the state-of-the-art algorithm [32] takes $O(Y(X + n))$ time, which still has a significant gap from the lower bound time complexity. As such, finding the optimal solutions for these problems is the promising future work.

Complexity-reduced algorithms for other kernel functions: In the state-of-the-art research studies [27, 30, 32], although these methods can reduce the time complexity for generating KDV, NKDV, and STKDV, all these methods only focus on the limited set of kernel functions (e.g., Epanechnikov, quartic, and uniform kernels). Therefore, these research studies cannot be extended to handle other important kernel functions (e.g., Gaussian kernel, cosine kernel, and exponential kernel) that can be supported by some famous software packages (e.g., Scikit-learn [75]). Therefore, finding a complexity-optimized solution for handling other kernel functions is also the important future work.

Future opportunities for K-function and its variants: There are two main future research studies in this direction.

Efficient and exact solutions for K-function and its variants: In recent years, there are a few research studies [33, 81] that can successfully reduce the time complexity for computing the network K-function. However, these studies cannot be extended to handle K-function (cf. Equation 2) and spatiotemporal K-function (cf. Equation 8), which are supported by commonly used software packages (e.g., R packages [4]). Therefore, existing solutions for solving these two problems are still in $O(n^2)$ time, which are not scalable to large-scale location datasets (e.g., New York taxi dataset [9] with 165 million data points), let alone to generate a K-function plot/spatiotemporal K-function plot. Furthermore, it is still unknown whether the time complexity of computing network K-function [33] is optimal. As such, finding efficient and exact solutions, with non-trivial time-complexity guarantees, for supporting K-function and its variants are still the open problems.

Efficient and approximate solutions for K-function and its variants: Although many approximation algorithms (e.g., function approximation methods [25, 34] and data sampling methods [78, 110]) have been developed for efficiently generating an approximate KDV, none of these approaches, to the best of our knowledge, has been extended to support K-function and its variants. Consider Equation 1 and Equation 2. Note that both of them have a common property: need to aggregate multiple terms. Based on this property, it is possible to modify approximate algorithms of KDV for solving these K-function-related problems, which can be another promising future work.

Future opportunities for other geospatial analytic tools: Many other geospatial analytic tools, including IDW, Kriging, Moran's I, and Getis-Ord General G (cf. Table 1), are also very time-consuming, which cannot be scalable to large-scale location datasets. For example, a naïve implementation of IDW takes $O(XYn)$ time [20], where $X \times Y$ and n denote the number of pixels and the number of location data points, respectively. To tackle this issue, we propose three future research studies in this direction.

Complexity-reduced algorithms for other tools: Although there are many complexity-reduced methods, including data sampling methods [77–79, 110, 111] and computational sharing methods [26, 32], for generating KDV with non-trivial accuracy guarantees, no complexity-reduced algorithm, to the best of our knowledge, has been proposed for supporting other tools. As such, developing efficient algorithms with non-trivial accuracy and time-complexity guarantees for other geospatial analytic tools can be the promising future work. For example, we can investigate whether some existing methods for KDV, e.g., data sampling methods, computational sharing methods, and function approximation methods in

Section 2.2, can be extended to support these tools with non-trivial guarantees.

Parallel/distributed and hardware-based algorithms for other tools: Although some parallel/distributed and hardware-based algorithms have been proposed to improve the efficiency for supporting other tools (e.g., [36, 53, 109] for Kriging), all these algorithms are only based on some basic methods, which can still be slow if a location dataset contains many data points (e.g., 165 million data points in the New York taxi dataset [9]). Therefore, investigating parallel/distributed and hardware-based approaches (e.g., GPU) for improving the efficiency of complexity-reduced (newly developed) algorithms can be another promising future work.

Computational hardness of other tools: Instead of improving the efficiency for supporting other geospatial analytic tools, another important research topic is to analyze the hardness of each tool (like the lower bound time complexity $\Omega(n^{\frac{4}{3}})$ of the DBSCAN problem [48, 49]) such that researchers can understand whether their newly developed algorithms are theoretically optimal.

Future opportunities for software development: Although many software packages, e.g., QGIS [11], ArcGIS [1], R packages [4], and PySAL [10] (a python package), have been developed to support geospatial analytic tools (cf. Table 1), all of these packages adopt naïve algorithms, which are inefficient (or even not feasible) to support large-scale location datasets nowadays. Therefore, the first promising future work is to develop new packages, based on efficient algorithms, for these geospatial analytic tools, e.g., python packages (like our recently developed python library, LIBKDV [29]) and R packages. Furthermore, since QGIS and ArcGIS are very famous software packages for conducting spatial analysis, the second promising future work is to develop QGIS and ArcGIS plugins (by integrating state-of-the-art algorithms) for supporting these two software packages. Moreover, domain experts can also adopt web-based geographic information systems, e.g., QGIS Cloud [12] and ArcGIS Online [2], to analyze their location datasets, the third promising future work is to integrate efficient algorithms into these web-based systems.

3 BIOGRAPHIES

All the presenters have jointly worked on many geospatial analytic problems and published their research results in top-tier venues [25–27, 30–34], including SIGMOD, PVLDB, ICDE, and TKDE. Moreover, they also have rich experience for software and system development in this topic. For example, they have developed the web-based system prototype, KDV-Explorer [28], and the python libraries, LIBKDV [29] and PyNKDV [35]. Furthermore, they have also jointly developed the Hong Kong COVID-19 hotspot map [6] and Macau COVID-19 hotspot map [8], which are now in use by Hong Kong and Macau citizens, respectively.

Tsz Nam Chan is a Research Assistant Professor in the Hong Kong Baptist University. He received his PhD degree and BEng degree from the Hong Kong Polytechnic University in 2019 and 2014, respectively. His research interests include spatiotemporal data management, geographic information systems, data visualization, kernel methods, and similarity search.

Leong Hou U received the Ph.D. degree from the University of Hong Kong, the M.Sc. degree from the University of Macau, and the B.Sc. degree from Taiwan Chi Nan University. He is now an Associate Professor with the State Key Laboratory of Internet of Things for Smart City, the Department of Computer and Information Science, University of Macau (UM). He is currently the interim head at the Centre for Data Science, UM. His research interests include spatio-temporal databases, large data query processing, graph learning, reinforcement learning, and optimization problems.

Byron Choi obtained the PhD and MSE degrees in Computer and Information Science from the University of Pennsylvania. He received a BEng degree in Computer Engineering from HKUST. He is currently the Associate Head and a Professor at the Department of Computer Science, Hong Kong Baptist University (HKBU). His research interests include graph-structured databases, database usability, database security, and time series analysis. He was awarded a distinguished program committee (PC) member from ACM SIGMOD 2021 and a best reviewer award from ACM CIKM 2021. He received the distinguished reviewer award from PVLDB 2019.

Jianliang Xu received the BEng degree in computer science and engineering from Zhejiang University, Hangzhou, China, in 1998 and the PhD degree in computer science from the Hong Kong University of Science and Technology in 2002. He is currently a Professor in the Department of Computer Science, Hong Kong Baptist University. He held visiting positions at Pennsylvania State University and Fudan University. His research interests include database, blockchain, and trusted computing. He has published 200+ papers in top-tier conferences and journals. He received the best paper awards of WISE2019 and MUST2021, and the best paper award runner-up of CIKM2020. He has served as the associate editor of TKDE and PVLDB, and the program committee member of SIGMOD, VLDB, and ICDE.

Reynold Cheng is a Professor of the Department of Computer Science in the University of Hong Kong (HKU). His research interests are in data science, big graph analytics, and uncertain data management. He received his BEng (Computer Engineering) in 1998, and MPhil (Computer Science and Information Systems) in 2000 from HKU. He then obtained his MSc and PhD degrees from Department of Computer Science of Purdue University in 2003 and 2005, respectively. He received the SIGMOD Research Highlights Award 2020. He is a member of IEEE, ACM, ACM SIGMOD, and UPE, was a PC co-chair of IEEE ICDE 2021, and has been serving on the program committees and review panels for leading database conferences and journals like SIGMOD, VLDB, ICDE, KDD, and TODS. He is on the editorial board of IS, KAIS, and DAPD, and was a former editorial board member of TKDE.

ACKNOWLEDGEMENT

This work was supported by the NSFC grant 62202401, the Science and Technology Development Fund Macau SAR 0015/2019/AKP, 0031/2022/A, SKL-IOTSC-2021-2023, the Research Grant of University of Macau MYRG2022-00252-FST, Wuyi University Hong Kong and Macau joint Research Fund 2021WGALH14, HKRGC RIF R2002-20F, HKRGC C2004-21GF, GDNSF 2019B1515130001, the University of Hong Kong (Projects 104005858 and 10400599), the Guangdong–Hong Kong-Macau Joint Laboratory Program 2020 (Project No: 2020B1212030009), and the Hong Kong Jockey Club Charities Trust 260920140. Leong Hou U is also affiliated with the State Key Laboratory of Internet of Things for Smart City, UM, and with the Centre for Data Science, UM.

REFERENCES

[1] 2023. ArcGIS. https://www.esri.com/en-us/arcgis/about-arcgis/overview.
[2] 2023. ArcGIS Online. https://www.arcgis.com/index.html.
[3] 2023. Chicago Data Portal. https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-Present/ijzp-q8t2.
[4] 2023. The Comprehensive R Archive Network. https://cran.r-project.org/.
[5] 2023. CrimeStat: Spatial Statistics Program for the Analysis of Crime Incident Locations. https://nij.ojp.gov/topics/articles/crimestat-spatial-statistics-program-analysis-crime-incident-locations.
[6] 2023. Hong Kong COVID-19 Hotspot Map. https://covid19.comp.hkbu.edu.hk/.
[7] 2023. Hong Kong GeoData Store. https://geodata.gov.hk/gs/view-dataset?uuid=d4ccd9be-3bc0-449b-bd27-9eb9b615f2db&sidx=0.
[8] 2023. Macau COVID-19 Hotspot Map. http://degroup.cis.um.edu.mo/covid-19/.
[9] 2023. NYC Yellow Taxi Trip Data. https://data.cityofnewyork.us/Transportation/2014-Yellow-Taxi-Trip-Data/gkne-dk5s.
[10] 2023. PySAL. https://pysal.org/.
[11] 2023. QGIS. https://www.qgis.org/en/site/.
[12] 2023. QGIS Cloud. https://qgiscloud.com/.
[13] 2023. SANET: Spatial Analysis along Networks. http://sanet.csis.u-tokyo.ac.jp/.
[14] 2023. spatstat: analysing spatial point patterns. https://spatstat.org/.
[15] 2023. spNetwork: Spatial Analysis on Network. https://cran.r-project.org/web/packages/spNetwork/index.html.
[16] Rosana Aguilera, Thomas Corringham, Alexander Gershunov, and Tarik Benmarhnia. 2021. Wildfire smoke impacts respiratory health more than fine particles from other sources: Observational evidence from Southern California. *Nature communications* 12, 1 (2021), 1–8.
[17] Josilene D. Alves, André S. Abade, Wigis P. Peres, Jonatas E. Borges, Sandra M. Santos, and Alessandro R. Scholze. 2021. Impact of COVID-19 on the indigenous population of Brazil: a geo-epidemiological study. *Epidemiology and Infection* 149 (2021), e185. https://doi.org/10.1017/S0950268821001849
[18] Tessa K. Anderson. 2009. Kernel density estimation and K-means clustering to profile road accident hotspots. *Accident Analysis & Prevention* 41, 3 (2009), 359–364. https://doi.org/10.1016/j.aap.2008.12.014
[19] Adrian Baddeley, Ege Rubak, and Rolf Turner. 2015. *Spatial point patterns: methodology and applications with R.* CRC press.
[20] Patrick M. Bartier and C.Peter Keller. 1996. Multivariate interpolation to incorporate thematic surface data using inverse distance weighting (IDW). *Computers & Geosciences* 22, 7 (1996), 795–799. https://doi.org/10.1016/0098-3004(96)00021-0
[21] Jon Louis Bentley. 1975. Multidimensional Binary Search Trees Used for Associative Searching. *Commun. ACM* 18, 9 (1975), 509–517.
[22] P. Jeffrey Brantingham, Jeremy Carter, John MacDonald, Chris Melde, and George Mohler. 2021. Is the recent surge in violence in American cities due to contagion? *Journal of Criminal Justice* 76 (2021), 101848. https://doi.org/10.1016/j.jcrimjus.2021.101848
[23] Chris Brunsdon, Jonathan Corcoran, and Gary Higgs. 2007. Visualising space and time in crime patterns: A comparison of methods. *Comput. Environ. Urban Syst.* 31, 1 (2007), 52–75. https://doi.org/10.1016/j.compenvurbsys.2005.07.009
[24] Ysabel A Castle and John M Kovacs. 2021. Identifying seasonal spatial patterns of crime in a small northern city. *Crime Science* 10, 1 (2021), 1–20.
[25] Tsz Nam Chan, Reynold Cheng, and Man Lung Yiu. 2020. QUAD: Quadratic-Bound-based Kernel Density Visualization. In *SIGMOD*. 35–50. https://doi.org/10.1145/3318464.3380561
[26] Tsz Nam Chan, Pak Lon Ip, Leong Hou U, Byron Choi, and Jianliang Xu. 2021. SAFE: A Share-and-Aggregate Bandwidth Exploration Framework for Kernel Density Visualization. *Proc. VLDB Endow.* 15, 3 (2021), 513–526.
[27] Tsz Nam Chan, Pak Lon Ip, Leong Hou U, Byron Choi, and Jianliang Xu. 2021. SWS: A Complexity-Optimized Solution for Spatial-Temporal Kernel Density Visualization. *Proc. VLDB Endow.* 15, 4 (2021), 814–827.
[28] Tsz Nam Chan, Pak Lon Ip, Leong Hou U, Weng Hou Tong, Shivansh Mittal, Ye Li, and Reynold Cheng. 2021. KDV-Explorer: A Near Real-Time Kernel Density Visualization System for Spatial Analysis. *Proceedings of the VLDB Endowment* 14, 12 (2021), 2655–2658. http://www.vldb.org/pvldb/vol14/p2655-chan.pdf
[29] Tsz Nam Chan, Pak Lon Ip, Kaiyan Zhao, Leong Hou U, Byron Choi, and Jianliang Xu. 2022. LIBKDV: A Versatile Kernel Density Visualization Library for Geospatial Analytics. *Proc. VLDB Endow.* 15, 12 (2022), 3606–3609. https://www.vldb.org/pvldb/vol15/p3606-chan.pdf
[30] Tsz Nam Chan, Zhe Li, Leong Hou U, Jianliang Xu, and Reynold Cheng. 2021. Fast Augmentation Algorithms for Network Kernel Density Visualization. *Proc. VLDB Endow.* 14, 9 (2021), 1503–1516.
[31] Tsz Nam Chan, Leong Hou U, Reynold Cheng, Man Lung Yiu, and Shivansh Mittal. 2022. Efficient Algorithms for Kernel Aggregation Queries. *IEEE Trans. Knowl. Data Eng.* 34, 6 (2022), 2726–2739. https://doi.org/10.1109/TKDE.2020.3018376
[32] Tsz Nam Chan, Leong Hou U, Byron Choi, and Jianliang Xu. 2022. SLAM: Efficient Sweep Line Algorithms for Kernel Density Visualization. In *SIGMOD*. ACM, 2120–2134. https://doi.org/10.1145/3514221.3517823
[33] Tsz Nam Chan, Leong Hou U, Yun Peng, Byron Choi, and Jianliang Xu. 2022. Fast Network K-function-based Spatial Analysis. *Proc. VLDB Endow.* 15, 11 (2022), 2853–2866. https://www.vldb.org/pvldb/vol15/p2853-chan.pdf
[34] Tsz Nam Chan, Man Lung Yiu, and Leong Hou U. 2019. KARL: Fast Kernel Aggregation Queries. In *ICDE*. 542–553. https://doi.org/10.1109/ICDE.2019.00055
[35] Tsz Nam Chan, Rui Zang, Pak Lon Ip, Leong Hou U, and Jianliang Xu. 2023. PyNKDV: An Efficient Network Kernel Density Visualization Library for Geospatial Analytic Systems. In *SIGMOD (To appear)*.
[36] Tangpei Cheng. 2013. Accelerating universal Kriging interpolation algorithm using CUDA-enabled GPU. *Computers & Geosciences* 54 (2013), 178–183. https://doi.org/10.1016/j.cageo.2012.11.013
[37] Jihoon Chung and Heeyoung Kim. 2019. Crime risk maps: A multivariate spatial analysis of crime data. *Geographical analysis* 51, 4 (2019), 475–499.
[38] Gao Cong and Christian S. Jensen. 2016. Querying Geo-Textual Data: Spatial Keyword Queries and Beyond. In *SIGMOD*. ACM, 2207–2212. https://doi.org/10.1145/2882903.2912572
[39] Jack Cordes and Marcia C. Castro. 2020. Spatial analysis of COVID-19 clusters and contextual factors in New York City. *Spatial and Spatio-temporal Epidemiology* 34 (2020), 100355. https://doi.org/10.1016/j.sste.2020.100355
[40] Mark de Berg, Otfried Cheong, Marc J. van Kreveld, and Mark H. Overmars. 2008. *Computational geometry: algorithms and applications, 3rd Edition.* Springer. https://www.worldcat.org/oclc/227584184
[41] Eric Delmelle, Coline Dony, Irene Casas, Meijuan Jia, and Wenwu Tang. 2014. Visualizing the impact of space-time uncertainties on dengue fever patterns. *International Journal of Geographical Information Science* 28, 5 (2014), 1107–1127. https://doi.org/10.1080/13658816.2013.871285
[42] Min Deng, Xuexi Yang, Yan Shi, Jianya Gong, Yang Liu, and Huimin Liu. 2019. A density-based approach for detecting network-constrained clusters in spatial point events. *International Journal of Geographical Information Science* 33, 3 (2019), 466–488. https://doi.org/10.1080/13658816.2018.1541177
[43] Ahmed Eldawy and Mohamed F. Mokbel. 2017. The Era of Big Spatial Data. *Proc. VLDB Endow.* 10, 12 (2017), 1992–1995. https://doi.org/10.14778/3137765.3137828
[44] Richard Elson, Tilman M. Davies, Iain R. Lake, Roberto Vivancos, Paula B. Blomquist, Andre Charlett, and Gavin Dabrera. 2021. The spatio-temporal distribution of COVID-19 infection in England between January and June 2020. *Epidemiology and Infection* 149 (2021), e73. https://doi.org/10.1017/S0950268821000534
[45] Junchuan Fan and Kathleen Stewart. 2021. Understanding collective human movement dynamics during large-scale events using big geosocial data analytics. *Computers, Environment and Urban Systems* 87 (2021), 101605.
[46] Ivan Franch-Pardo, Michael R Desjardins, Isabel Barea-Navarro, and Artemi Cerdà. 2021. A review of GIS methodologies to analyze the dynamics of COVID-19 in the second half of 2020. *Transactions in GIS* 25, 5 (2021), 2191–2239.
[47] Edward Gan and Peter Bailis. 2017. Scalable Kernel Density Classification via Threshold-Based Pruning. In *SIGMOD*. 945–959.
[48] Junhao Gan and Yufei Tao. 2015. DBSCAN Revisited: Mis-Claim, Un-Fixability, and Approximation. In *SIGMOD*. ACM, 519–530. https://doi.org/10.1145/2723372.2737792
[49] Junhao Gan and Yufei Tao. 2017. On the Hardness and Approximation of Euclidean DBSCAN. *ACM Trans. Database Syst.* 42, 3 (2017), 14:1–14:45. https://doi.org/10.1145/3083897
[50] A. Gramacki. 2017. *Nonparametric Kernel Density Estimation and Its Computational Aspects.* Springer International Publishing. https://books.google.com.hk/books?id=PCpEDwAAQBAJ
[51] Alexander G. Gray and Andrew W. Moore. 2003. Nonparametric Density Estimation: Toward Computational Tractability. In *SDM*. 203–211.
[52] Alexander G. Gray and Andrew W. Moore. 2003. Rapid Evaluation of Multiple Density Models. In *AISTATS*. Society for Artificial Intelligence and Statistics. http://research.microsoft.com/en-us/um/cambridge/events/aistats2003/proceedings/180.pdf
[53] Qingfeng Guan, Phaedon C. Kyriakidis, and Michael F. Goodchild. 2011. A parallel computing approach to fast geostatistical areal interpolation. *Int. J. Geogr. Inf. Sci.* 25, 8 (2011), 1241–1267. https://doi.org/10.1080/13658816.2011.563744
[54] Tung Hoang and Tho Thi Anh Tran. 2021. Ambient air pollution, meteorology, and COVID-19 infection in Korea. *Journal of medical virology* 93, 2 (2021), 878–885.
[55] Alexander Hohl, Eric Delmelle, Wenwu Tang, and Irene Casas. 2016. Accelerating the discovery of space-time patterns of infectious diseases using parallel computing. *Spatial and Spatio-temporal Epidemiology* 19 (2016), 10 – 20. https://doi.org/10.1016/j.sste.2016.05.002
[56] Alexander Hohl, Minrui Zheng, Wenwu Tang, Eric Delmelle, and Irene Casas. 2017. Spatiotemporal point pattern analysis using Ripley's K function. *Geospatial Data science techniques and applications* (2017), 155–76.
[57] Yujie Hu, Fahui Wang, Cecile Guin, and Haojie Zhu. 2018. A spatio-temporal kernel density estimation framework for predictive crime hotspot mapping and evaluation. *Applied Geography* 99 (2018), 89 – 97. https://doi.org/10.1016/j.

apgeog.2018.08.001
[58] Jianwei Huang and Mei-Po Kwan. 2022. Uncertainties in the Assessment of COVID-19 Risk: A Study of People's Exposure to High-Risk Environments Using Individual-Level Activity Data. *Annals of the American Association of Geographers* 112, 4 (2022), 968–987. https://doi.org/10.1080/24694452.2021.1943301 arXiv:https://doi.org/10.1080/24694452.2021.1943301
[59] Atina Husnayain, Ting-Wu Chuang, Anis Fuad, and Emily Chia-Yu Su. 2021. High variability in model performance of Google relative search volumes in spatially clustered COVID-19 areas of the USA. *International Journal of Infectious Diseases* 109 (2021), 269–278. https://doi.org/10.1016/j.ijid.2021.07.031
[60] Paul Jeffrey Brantingham, George E Tita, and George Mohler. 2021. Gang-related crime in Los Angeles remained stable following COVID-19 social distancing orders. *Criminology & Public Policy* 20, 3 (2021), 423–436.
[61] Gaige Hunter Kerr, Daniel L Goldberg, and Susan C Anenberg. 2021. COVID-19 pandemic reveals persistent disparities in nitrogen dioxide pollution. *Proceedings of the National Academy of Sciences* 118, 30 (2021), e2022409118.
[62] Ourania Kounadi, Alina Ristea, Michael Leitner, and Chad Langford. 2018. Population at risk: Using areal interpolation and Twitter messages to create population models for burglaries and robberies. *Cartography and geographic information science* 45, 3 (2018), 205–220.
[63] Maria Krommyda and Verena Kantere. 2020. Visualization Systems for Linked Datasets. In *ICDE*. IEEE, 1790–1793. https://doi.org/10.1109/ICDE48307.2020.00171
[64] Pei-Fen Kuo and Dominique Lord. 2019. A promising example of smart policing: A cross-national study of the effectiveness of a data-driven approach to crime and traffic safety. *Case Studies on Transport Policy* 7, 4 (2019), 761–771. https://doi.org/10.1016/j.cstp.2019.08.005
[65] Pei-Fen Kuo, Dominique Lord, and Troy Duane Walden. 2013. Using geographical information systems to organize police patrol routes effectively by grouping hotspots of crash and crime data. *Journal of Transport Geography* 30 (2013), 138–148. https://doi.org/10.1016/j.jtrangeo.2013.04.006
[66] David S. Lamb, Joni A. Downs, and Chanyoung Lee. 2016. The network K-function in context: examining the effects of network structure on the network K-function. *Transactions in GIS* 20, 3 (2016), 448–460. https://doi.org/10.1111/tgis.12157 arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/tgis.12157
[67] Ove Daae Lampe and Helwig Hauser. 2011. Interactive visualization of streaming data with Kernel Density Estimation. In *PacificVis*. 171–178. https://doi.org/10.1109/PACIFICVIS.2011.5742387
[68] Jay Lee, Junfang Gong, and Shengwen Li. 2017. Exploring spatiotemporal clusters based on extended kernel estimation methods. *Int. J. Geogr. Inf. Sci.* 31, 6 (2017), 1154–1177. https://doi.org/10.1080/13658816.2017.1287371
[69] Yunxuan Li, Mohamed Abdel-Aty, Jinghui Yuan, Zeyang Cheng, and Jian Lu. 2020. Analyzing traffic violation behavior at urban intersections: A spatio-temporal kernel density estimation approach using automated enforcement system data. *Accident Analysis & Prevention* 141 (2020), 105509. https://doi.org/10.1016/j.aap.2020.105509
[70] Ahmed R. Mahmood and Walid G. Aref. 2017. Query Processing Techniques for Big Spatial-Keyword Data. In *SIGMOD*. ACM, 1777–1782. https://doi.org/10.1145/3035918.3054773
[71] Andrew W. Moore. 2000. The Anchors Hierarchy: Using the Triangle Inequality to Survive High Dimensional Data. In *UAI*. 397–405.
[72] Kyriakos Mouratidis. 2017. Geometric Approaches for Top-k Queries. *Proc. VLDB Endow.* 10, 12 (2017), 1985–1987. https://doi.org/10.14778/3137765.3137826
[73] A. Okabe and K. Sugihara. 2012. *Spatial analysis along networks: statistical and computational methods*. Wiley. https://books.google.com.hk/books?id=48GRqj51_W8C
[74] Atsuyuki Okabe and Ikuho Yamada. 2001. The K-function method on a network and its computational implementation. *Geographical analysis* 33, 3 (2001), 271–290.
[75] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake VanderPlas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12 (2011), 2825–2830.
[76] Alexandre Perrot, Romain Bourqui, Nicolas Hanusse, Frédéric Lalanne, and David Auber. 2015. Large Interactive Visualization of Density Functions on Big Data Infrastructure. In *LDAV*. 99–106. https://doi.org/10.1109/LDAV.2015.7348077
[77] Jeff M. Phillips. 2013. ϵ-Samples for Kernels. In *SODA*. 1622–1632. https://doi.org/10.1137/1.9781611973105.116
[78] Jeff M. Phillips and Wai Ming Tai. 2018. Improved Coresets for Kernel Density Estimates. In *SODA*. 2718–2727. https://doi.org/10.1137/1.9781611975031.173
[79] Jeff M. Phillips and Wai Ming Tai. 2018. Near-Optimal Coresets of Kernel Density Estimates. In *SOCG*. 66:1–66:13. https://doi.org/10.4230/LIPIcs.SoCG.2018.66
[80] Aloys L Prinz and David J Richter. 2022. Long-term exposure to fine particulate matter air pollution: An ecological study of its effect on COVID-19 cases and fatality in Germany. *Environmental research* 204 (2022), 111948.
[81] Suman Rakshit, Adrian Baddeley, and Gopalan Nair. 2019. Efficient Code for Second Order Analysis of Events on a Linear Network. *Journal of Statistical Software, Articles* 90, 1 (2019), 1–37. https://doi.org/10.18637/jss.v090.i01
[82] Alex Reinhart. 2018. A review of self-exciting spatio-temporal point processes and their applications. *Statist. Sci.* 33, 3 (2018), 299–318.
[83] Alina Ristea, Mohammad Al Boni, Bernd Resch, Matthew S. Gerber, and Michael Leitner. 2020. Spatial crime distribution and prediction for sporting events using social media. *Int. J. Geogr. Inf. Sci.* 34, 9 (2020), 1708–1739. https://doi.org/10.1080/13658816.2020.1719495
[84] Gabriel Rosser, Toby O. Davies, Kate. Bowers, Shane D. Johnson, and T. Cheng. 2017. Predictive Crime Mapping: Arbitrary Grids or Street Networks? *Journal of Quantitative Criminology* 33 (2017), 569 – 594.
[85] Ibrahim Sabek and Mohamed F. Mokbel. 2019. Machine Learning Meets Big Spatial Data. *Proc. VLDB Endow.* 12, 12 (2019), 1982–1985. https://doi.org/10.14778/3352063.3352115
[86] Erik Saule, Dinesh Panchananam, Alexander Hohl, Wenwu Tang, and Eric Delmelle. 2017. Parallel Space-Time Kernel Density Estimation. In *ICPP*. 483–492. https://doi.org/10.1109/ICPP.2017.57
[87] Xun Shi, Meifang Li, Olivia Hunter, Bart Guetti, Angeline Andrew, Elijah Stommel, Walter Bradley, and Margaret Karagas. 2019. Estimation of environmental exposure: interpolation, kernel density estimation or snapshotting. *Annals of GIS* 25, 1 (2019), 1–8. https://doi.org/10.1080/19475683.2018.1555188
[88] Zhicheng Shi and Lilian S.C. Pun-Cheng. 2019. Spatiotemporal Data Clustering: A Survey of Methods. *ISPRS International Journal of Geo-Information* 8, 3 (2019). https://doi.org/10.3390/ijgi8030112
[89] Katarzyna Sila-Nowicka, Jan Vandrol, Taylor Oshan, Jed A. Long, Urska Demsar, and A. Stewart Fotheringham. 2016. Analysis of human mobility patterns from GPS trajectories and contextual information. *Int. J. Geogr. Inf. Sci.* 30, 5 (2016), 881–906. https://doi.org/10.1080/13658816.2015.1100731
[90] Nan Tang, Eugene Wu, and Guoliang Li. 2019. Towards Democratizing Relational Data Visualization. In *SIGMOD*. ACM, 2025–2030. https://doi.org/10.1145/3299869.3314029
[91] Wenwu Tang, Wenpeng Feng, and Meijuan Jia. 2015. Massively parallel spatial point pattern analysis: Ripley's K function accelerated using graphics processing units. *International Journal of Geographical Information Science* 29, 3 (2015), 412–439. https://doi.org/10.1080/13658816.2014.976569 arXiv:https://doi.org/10.1080/13658816.2014.976569
[92] Lalita Thakali, Tae J. Kwon, and Liping Fu. 2015. Identification of crash hotspots using kernel density estimation and kriging methods: a comparison. *Journal of Modern Transportation* 23, 2 (2015), 93–106. https://doi.org/10.1007/s40534-015-0068-0
[93] Mehmet Baran Ulak, Eren Erman Ozguven, O Arda Vanli, and Mark W Horner. 2019. Exploring alternative spatial weights to detect crash hotspots. *Computers, Environment and Urban Systems* 78 (2019), 101398.
[94] Alese Wooditch and David Weisburd. 2016. Using space–time analysis to evaluate criminal justice programs: An application to stop-question-frisk practices. *Journal of quantitative criminology* 32, 2 (2016), 191–213.
[95] Kun Xie, Kaan Ozbay, Abdullah Kurkcu, and Hong Yang. 2017. Analysis of Traffic Crashes Involving Pedestrians Using Big Data: Investigation of Contributing Factors and Identification of Hotspots. *Risk Analysis* 37, 8 (2017), 1459–1476. https://EconPapers.repec.org/RePEc:wly:riskan:v:37:y:2017:i:8:p:1459-1476
[96] Zhixiao Xie and Jun Yan. 2008. Kernel Density Estimation of traffic accidents in a network space. *Computers, Environment and Urban Systems* 32, 5 (2008), 396 – 406. https://doi.org/10.1016/j.compenvurbsys.2008.05.001
[97] Zhixiao Xie and Jun Yan. 2013. Detecting traffic accident clusters with network kernel density estimation and local spatial statistics: an integrated approach. *Journal of Transport Geography* 31 (2013), 64 – 71. https://doi.org/10.1016/j.jtrangeo.2013.05.009
[98] Rui Xin, Tinghua Ai, Linfang Ding, Ruoxin Zhu, and Liqiu Meng. 2022. Impact of the COVID-19 pandemic on urban human mobility-A multiscale geospatial network analysis using New York bike-sharing data. *Cities* 126 (2022), 103677.
[99] Li Xu, Mei-Po Kwan, Sara McLafferty, and Shaowen Wang. 2017. Predicting demand for 311 non-emergency municipal services: An adaptive space-time kernel approach. *Applied Geography* 89 (2017), 133–141. https://doi.org/10.1016/j.apgeog.2017.10.012
[100] Ikuho Yamada and Jean-Claude Thill. 2004. Comparison of planar and network K-functions in traffic accident analysis. *Journal of Transport Geography* 12, 2 (2004), 149–158. https://doi.org/10.1016/j.jtrangeo.2003.10.006
[101] Bo Yang, Lin Liu, Minxuan Lan, Zengli Wang, Hanlin Zhou, and Hongjie Yu. 2020. A spatio-temporal method for crime prediction using historical crime data and transitional zones identified from nightlight imagery. *International Journal of Geographical Information Science* 34, 9 (2020), 1740–1764.
[102] Hao Yu, Pan Liu, Jun Chen, and Hao Wang. 2014. Comparative analysis of the spatial analysis methods for hotspot identification. *Accident Analysis and Prevention* 66 (2014), 80 – 88. https://doi.org/10.1016/j.aap.2014.01.017
[103] Jia Yu and Mohamed Sarwat. 2019. Geospatial Data Management in Apache Spark: A Tutorial. In *ICDE*. IEEE, 2060–2063. https://doi.org/10.1109/ICDE.2019.00239

[104] Xiaonan Yu, Cesunica Ivey, Zhijiong Huang, Sashikanth Gurram, Vijayaraghavan Sivaraman, Huizhong Shen, Naveen Eluru, Samiul Hasan, Lucas Henneman, Guoliang Shi, Hongliang Zhang, Haofei Yu, and Junyu Zheng. 2020. Quantifying the impact of daily mobility on errors in air pollution exposure estimation using mobile phone location data. *Environment International* 141 (2020), 105772. https://doi.org/10.1016/j.envint.2020.105772

[105] Guiming Zhang. 2022. Detecting and Visualizing Observation Hot-Spots in Massive Volunteer-Contributed Geographic Data across Spatial Scales Using GPU-Accelerated Kernel Density Estimation. *ISPRS International Journal of Geo-Information* 11, 1 (2022), 55.

[106] Guiming Zhang, Qunying Huang, A-Xing Zhu, and John H Keel. 2016. Enabling point pattern analysis on spatial big data using cloud computing: optimizing and accelerating Ripley's K function. *International Journal of Geographical Information Science* 30, 11 (2016), 2230–2252.

[107] Guiming Zhang, A-Xing Zhu, and Qunying Huang. 2017. A GPU-accelerated adaptive kernel density estimation approach for efficient point pattern analysis on spatial big data. *International Journal of Geographical Information Science* 31, 10 (2017), 2068–2097. https://doi.org/10.1080/13658816.2017.1324975

[108] Sui Zhang, Minghao Wang, Zhao Yang, and Baolei Zhang. 2022. Do spatiotemporal units matter for exploring the microgeographies of epidemics? *Applied Geography* 142 (2022), 102692.

[109] Yueheng Zhang, Xinqi Zheng, Zhenhua Wang, Gang Ai, and Qing Huang. 2018. Implementation of a parallel GPU-based space-time kriging framework. *ISPRS International Journal of Geo-Information* 7, 5 (2018), 193.

[110] Yan Zheng, Jeffrey Jestes, Jeff M. Phillips, and Feifei Li. 2013. Quality and efficiency for kernel density estimates in large data. In *SIGMOD*. 433–444.

[111] Yan Zheng and Jeff M. Phillips. 2015. L∞ Error and Bandwidth Selection for Kernel Density Estimates of Large Data. In *SIGKDD*. 1533–1542. https://doi.org/10.1145/2783258.2783357

[112] Haixiang Zou, Yang Yue, Qingquan Li, and Anthony GO Yeh. 2012. An improved distance metric for the interpolation of link-based traffic data using kriging: a case study of a large-scale urban road network. *International Journal of Geographical Information Science* 26, 4 (2012), 667–689.

[113] Andreas Züfle, Goce Trajcevski, Dieter Pfoser, Matthias Renz, Matthew T. Rice, Timothy Leslie, Paul L. Delamater, and Tobias Emrich. 2017. Handling Uncertainty in Geo-Spatial Data. In *ICDE*. IEEE Computer Society, 1467–1470. https://doi.org/10.1109/ICDE.2017.212