

# VAUD: A Visual Analysis Approach for Exploring Spatio-Temporal Urban Data

Wei Chen, Zhaosong Huang, Feiran Wu, Minfeng Zhu, Huihua Guan, and Ross Maciejewski

**Abstract**—Urban data is massive, heterogeneous, and spatio-temporal, posing a substantial challenge for visualization and analysis. In this paper, we design and implement a novel visual analytics approach, Visual Analyzer for Urban Data (VAUD), that supports the visualization, querying, and exploration of urban data. Our approach allows for cross-domain correlation from multiple data sources by leveraging spatial-temporal and social inter-connectedness features. Through our approach, the analyst is able to select, filter, aggregate across multiple data sources and extract information that would be hidden to a single data subset. To illustrate the effectiveness of our approach, we provide case studies on a real urban dataset that contains the cyber-, physical-, and social-information of 14 million citizens over 22 days.

**Index Terms**—Urban data, Visual Analysis, Visual Reasoning, Heterogeneous, Spatio-temporal.



## 1 INTRODUCTION

**S**ENSING technologies, social media and large-scale computing infrastructures have produced a variety of urban data (e.g., human mobility, mobile phone calls, traffic, etc.). Designing approaches and tools to understand and utilize urban data brings a unique set of research and engineering challenges, specifically with regards to data querying and analysis. Despite the wealth of research on urban data, contemporary analytical tools [4], [5], [21], [26], [42], [44] are often inadequate to handle the large volume, sparseness and heterogeneity of data, let alone support interactive visual analysis in data-intensive applications. Specifically, most visual analytics tools tend to focus only on a single data source making it difficult to discover and link overlapping details of an event from multiple data sources. In order to support real-time visualization and interactive analysis of massive spatio-temporal data, visual analytics approaches commonly adopt in-memory databases and custom-built data representations [13], for example, the space-time cube [20], [22], spatio-temporal aggregation [25], [26] and feature extraction [2], [11], [17]. However, since urban data is collected from different domains (e.g., mobility, power consumption, traffic, social media), there is a need for tools that can perform cross-domain analytical tasks. Such a system demands relation-aware data queries and reasoning that leverages the spatio-temporal inter-connectedness of information within a uniform space.

This work focuses on two key challenges in developing a framework for multi-source urban data analysis, specifically, visual queries and visual reasoning. **Visual Queries:** In analyzing urban data, visualization is often the interface that connects massive data items to human intelligence. As an essential

component, visual queries must provide analysts with the ability to investigate and directly access selected data points or features. However, many patterns and events can be obscured in urban data, requiring the fusion of multiple datasets in order to enable complex pattern analysis and identification. Previous visual analysis approaches are typically designed for exploring only a single data source, e.g., trajectories or movements [9], [15]. To enable breakthroughs in data exploration where information overload is a barrier to insight, there is a dire need for a means of cross-domain visual querying that effectively fuses the knowledge from multiple data sources. **Visual Reasoning:** To allow users to derive insights with no prior knowledge, it is desirable to only show the most relevant portions of a dataset while suggesting directions for potential exploration. The key is the proper utilization of semantic information that resides in the spatio-temporal and social inter-connectedness of urban data. Existing visual reasoning solutions for urban data [6], [38] do not focus on methods for semantically linking multiple urban data sources; instead, they focus on a single, domain specific data set. Our goal is to utilize the inter-connectedness among multiple domains, thus developing a visual analysis system that can deduce implicit relations, reveal hidden patterns, and identify events of interest through an exploratory visual interface. Here, we present a visual analytics system that supports visual queries and reasoning across multiple, semantically linked urban data sets. Our contributions include:

- A new visual query model that enables cross-domain correlation and deduction from multiple data sources.
- A visual analytics framework that supports the visualization, correlation, querying, and reasoning of citywide urban data for various analysis tasks.

Our approach leverages conventional visual analytics approaches for cyber-physical-social (CPS) systems [23] and empowers users with insights and decisions derived from cross-domain data. To demonstrate the efficacy of our framework, this paper will focus on a real urban dataset collected from Jan. 10, 2014 to Jan. 31, 2014 in a city with a population of 14 million, more details are described in Section 3.1. Case studies using this dataset

- *W. Chen is with The State Key Lab of CAD & CG, Zhejiang University and Alibaba-Zhejiang University Joint Institute of Frontier Technologies. E-mail: chenwei@cad.zju.edu.cn*
- *Z. Huang, F. Wu, M. Zhu, and H. Guan are with The State Key Lab of CAD & CG, Zhejiang University. E-mail: {zhaosong\_huang, minfeng\_zhu, ghh} @zju.edu.cn, tomwfr007@gmail.com*
- *Ross Maciejewski is with The School of Computing, Informatics & Decision Systems Engineering, Arizona State University. E-mail: rmacieje@asu.edu*

are provided to demonstrate our visual querying and reasoning functionality. While this dataset is only a small portion of the data domains that are produced citywide, this dataset does cover a variety of urban data categories including: spatio-temporal, environmental, social, cyber, text, and traffic. As such, the dataset and case studies serve as an exemplar for our framework and demonstrate how this framework could be applied for managing and analyzing cross-domain urban data.

## 2 RELATED WORK

Our work builds on visual analytics research for urban data that spans data management, querying, reasoning, and visualization.

**Urban Data Management:** Massive, heterogeneous, and spatio-temporal urban data poses many challenges with regards to data representation and management for visual analytics [8], [13], [45]. We explore urban data in the context of: space (where), time (when), and objects (what) [29], [31]. In the visual analytics community, work has been focused on representing spatio-temporal data trajectories or categorical event data. The mainstream data structures for such work include the space-time cube (STC) [22] which employs a 3D grid where each voxel stores some part of the data, and time is (generally) represented along the Z-axis where the X and Y axes represent the geographical space. Extended variants (e.g., [24]) of the STC facilitate efficient object localization and event detection [20]. To improve the performance of visual exploration, spatio-temporal aggregation techniques [2], [25], [26] are widely employed. In particular, nano-cubes [25] introduces a hierarchical sparse representation in different tree levels to achieve interactive analysis rates. Despite the dramatic progress of the data management community in terms of storage and performance, integrated schemes that can support querying across multi-source urban data are still under-explored. Our work uses the space-time-cube (STC) [22] as the canonical space to manage spatio-temporal objects. In addition, data objects (e.g., persons, cars, places, events) and the connections between the objects are also stored.

**Urban Data Query:** Data querying is an essential function of databases which allows the user to retrieve data from one or more tables or expressions [36]. While standard data queries are performed for relational databases, retrieving information from urban data is challenging, and a variety of spatial database structures have been developed [1], [46]. The challenge is how to encode unstructured data and retrieve information based on a given criteria. Spatio-temporal data typically comes in the form of categorical events, values mapped to an areal unit, or trajectories. Recently, much attention has been paid to clustering and querying trajectories [16], [46]. Typical solutions use the location as an index to perform similarity computations [9], but recent work by Sakr, Attia and Güting [32] has introduced a language that can consistently express and evaluate sets of spatio-temporal pattern queries.

Rather than retrieving information from database systems via programming languages, many systems now employ visual queries as a means of engaging more casual users. In a visual query scheme, analysts can dynamically construct and modify the query by means of a sequence of user interactions, achieving a balance between simplicity and expressiveness. Many models have been proposed for visually querying spatio-temporal data [15], [38]. By allowing users to manipulate strokes and iteratively refine the results in the visualization, the user's intention can be inferred

from the topology of sketches and be used to perform queries. Our query model adopts the notion of defining user intention via dragging and dropping and offers extensions that are designed to support queries over multi-source data.

**Urban Data Visual Analysis:** As more and more data has become available, the need to develop methods for dynamically exploring related datasets has grown. Specifically, in the urban planning community, access to multiple data sources can provide insights into traffic patterns, food deserts and a variety of other issues that planners should account for. Representative work includes inferring air quality, diagnosing urban noise, real-time gas consumption and pollution emission, and real estate ranking and clustering [30], [39], [46]. A major problem with past analyses is that most analytical algorithms process data from start to finish regardless of the time it takes. However, humans conducting interactive tasks expect results to appear quickly, even if the initial results are incomplete or estimations. Such a requirement lends itself well to a visual analytics approach [43].

Yet, the visualization of urban data is still a challenging task. The amount of urban data collected often exceeds the upper limit of interactive visualization tools. While data management and analysis algorithms can reduce the number of items and dimensions through feature extraction, low-dimensional embedding, sampling or aggregation, a major challenge when connecting analytical algorithms to interactive visualizations is maintaining interactivity. For spatio-temporal urban data, a visual analysis system typically employs the STC representation to support manipulation and querying of information in a unified space [2]. Conventional solutions [4], [11], [17], [18] couple analytical, topological, and visual methods for dissecting and studying spatio-temporal and multivariate data. To support communication and coordination in collaborative sense making, other views can also be integrated [28]. However, very few systems exist that explore methods of cross data source analysis, fusion and visualization. In fact, most studies focus on visual analytics systems of single source urban data [5], [27]. For instance, taxi trajectories and movement data have been used for road evaluation [38], discovering significant places [3] and traffic analysis [40]. Other work has explored the correlation between traffic cell patterns and link/route flow patterns [41], and public utility data has been used for analyzing service performance [44] and crowd movement patterns [42]. Our system allows visual analysis via a drag-and-drop based interface that supports interactive visual queries and exploration over cross domains data.

**Visual Reasoning:** Another essential component of visual analysis is visual reasoning [14] to establish and verify facts and justify practices based on visually communicated information. With the rapid increase of urban data, there is a dire need for visual reasoning tools for large-scale urban data. Recent work [37], [47] has focused on visual reasoning as a classification problem and employs data mining techniques to optimize the classification objective. For instance, Arietta et al. [6] propose a novel technique for automatically identifying and validating predictive relationships between the visual appearance of a city and its non-visual attributes (e.g., crime statistics, housing prices, population density, etc.). Alternatively, we can regard reasoning as a deduction process based on a knowledge graph representation [7] where the objects are encoded as nodes and the edges represent their relationships [35]. Visual reasoning through a large-scale graph representation can be efficiently accomplished by interactive

network navigation [10]. A recently developed visual query system [33] supports the flexible exploration of a single data source by interactively constructing queries as visual hypergraphs. Compared to classifiers that tackle one specific task, the graph based representation enables querying over a large constructed knowledge base and can potentially answer more complicated questions. In contrast to these approaches, our visual analysis work focuses on multi-source data, incorporating a wide range of heterogeneous information, allowing us to answer a diverse set of visual queries. In many cases, we need to harness a diverse set of data sources (e.g., human mobility data, road networks, social networks, etc.) to improve visual reasoning.

### 3 THE VAUD DATA QUERY MODEL

#### 3.1 Data Description

To demonstrate the efficacy of our system, this paper focuses on a real urban dataset collected from Jan. 10, 2014 to Jan. 31, 2014 in a city with 14 million citizens. The dataset contains the following information:

- Geographical data: A road network of the city from OpenStreet Map [19] containing 34,997 nodes and 3,794 segments with a total length of 4,524 km.
- Points of Interests (POIs) data: The information of 938,712 POI locations where each record contains the longitude, latitude, name, and functionality of a structure in the urban environment (e.g., shopping malls, restaurants).
- Street view data: Street view data has been downloaded from the Baidu map service to provide linked imagery of locations.
- Real estate data: 5,684 estate records in residential sub-districts where each record contains the name, longitude, latitude, sales price, and the year that the building was constructed.
- Mobile phone location data: 308 billion location records of 7 million anonymized mobile phone users (around 50% of the population in the city) where each record contains an anonymous User ID, a cell tower ID, and a time stamp. The location accuracy is 100- to 5000- meters depending on the cell tower coverage in the area, and the regional functionality of each cell tower location is also provided.
- Social network data: 27 million mobile phone call records among 7 million users with each record containing two anonymous user IDs and a time stamp. A social network is also extracted from the call records.
- Microblog data: 93,491 posts of a popular microblog website whose geotags fall inside the city boundaries. Each record contains textual information, a time stamp, and a geotag (when available).
- Taxi GPS trajectory data: 272,470,343 trajectory records for 3,691 taxis recorded every 20 seconds where each record contains a taxi ID, a GPS location, the speed, the occupancy status, and a time stamp. On average, one million trips are recorded each day.
- Taxi profile data: The detailed information of all 3,691 taxis where each record contains the taxi ID, the taxi driver's traffic records, and the affiliated taxi company.

#### 3.2 Data Representation

While it is straightforward to apply a spatial relational database model [34] to heterogeneous urban data, our goal is to manage data

objects (e.g., persons, cars, places, events) and the connections between the objects that can be inferred when using multiple data sources. The most frequent and important relationships may be derived from the spatio-temporal interconnectedness of the multiple data sources. Thus, space and time must be considered as first class entities that can provide a rich source of new capabilities for analyzing urban data. While spatio-temporal information can be stored in various forms and at various levels, relational support to use this information in analyzing urban data is lacking. In our proposed framework, the geographical and time-oriented properties of objects should be normalized into a canonical space so that objects can be related by shared locations and time. In this way, a set of heterogeneous urban data can be represented with two classes of representations: object-based and space-time-cube based.

**Object based:** A list of objects can be extracted from each type of urban data, like the users from the mobile phone location data or the taxi drivers from the taxi GPS data. Each object consists of four distinct attributes: Identification Attributes (*which*), Spatial attributes (*where*), Temporal attributes (*when*), and Descriptive attributes (*what*):

- 1) **Identification Attributes** contain information which identifies the object, such as the user name, the taxi ID, or phone number.
- 2) **Spatial Attributes** contain the spatial information of an object, such as the latitude and longitude or street address.
- 3) **Temporal Attributes** contain the temporal attributes associated with an object, such as a date or a period of time.
- 4) **Descriptive Attributes** contain the descriptive information associated with an object, such as the age, speed, or direction.

In addition, there are direct or indirect relations among different types of objects. Some relations can be pre-built and retrieved during runtime, such as the social network from the mobile phone call data. Alternatively, some relations are generated on-the-fly during the analysis process, such as the riding experience of a taxi driver and a passenger, as demonstrated in the first case study.

**Space-time-cube based:** We leverage the space-time-cube (STC) [22] as the canonical space for accommodating spatio-temporal objects. Specifically, we split the entire time period of the urban data into slices (e.g., days). We construct an STC for each time slice and uniformly subdivide the STC into a 3D grid for a given resolution, where the resolution is determined based on the analysis tasks. As such, a cell of the STC refers to a geographical location and a time interval in the time slice associated with the STC. Finally, we sequentially relate records of each object into an STC cell by leveraging the time stamp and location information. A reference to the object is then recorded in the cell. The spatio-temporal data and associated STCs (Figure 1) support fast querying of spatio-temporal information and facilitate indirect connections of objects by means of the spatial-temporal interconnectedness. Other data types, e.g. POI data, have no temporal information and do not fit in an STC. We store such data in a database and build indexes on the spatial attributes to support fast queries.

#### 3.3 The VAUD Query Model

The VAUD query model has been designed to enable cross-domain queries and data fusion. In order to enable clear query

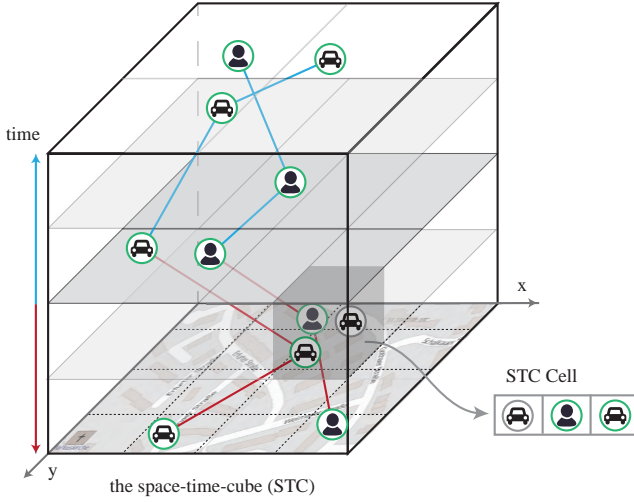


Fig. 1. Representing two cars and a person with STCs. STCs employ a 3D grid where each voxel stores some part of the data, and time is represented along the Z-axis where the X and Y axes represent the geographical space. Records of each object are related into an STC cell by leveraging the time stamp and location information.

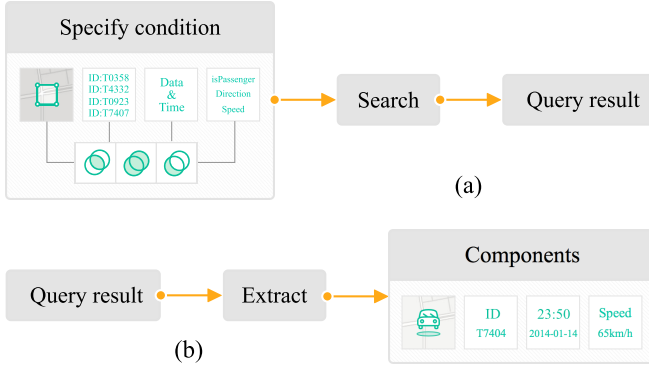


Fig. 2. Illustration of (a) an atomic query and (b) an extraction.

specification, we split all queries into a series of atomic expressions. An **atomic query** (Figure 2 (a)) is composed of three components: a query condition, a query operation, and query results. The condition is a combination of four component constraints (*which*, *when*, *where*, *what*). The query results are a set of records that fit the query conditions. The data source of the objects is encoded as a subscription, e.g.,  $object_1$  means that it is from dataset 1. The query operation is encoded with a directional arrow  $\rightarrow$ . For instance, Equation 1 denotes a query  $Q$  that retrieves objects that appear in a location from dataset 1. A combination can be a *union*, an *intersection*, or a *complement*. First, we use the symbol “,” between condition A and B to represent *union*, e.g., objects that satisfy A and B. Equation 2 denotes a query  $Q$  that searches for objects which appear in a location or move in a period of time from dataset 2. Likewise, the *intersection* returns objects that meet the condition A and condition B and can be represented by assigning “+” between A and B. For instance, Equation 3 denotes a query  $Q$  that retrieves objects that appear in a location in a certain time period from the dataset 1.

$$Q = where \rightarrow objects_1 \quad (1)$$

$$Q = when, where \rightarrow objects_2 \quad (2)$$

$$Q = when + where \rightarrow objects_1 \quad (3)$$



Fig. 3. Illustration of a query sequence that is composed of a set of (a) query operations and (b) extraction operations.

An **extraction** (Figure 2 (b)) is composed of three components, the query results, an extraction operation, and a component of the object. An *extraction* indicates that a component is to be extracted from the object. The extraction operation is encoded with a dashed directional arrow and is represented with the symbol  $\pi$  in the expression. The symbol  $\pi_i object$  is used to denote a component  $i$  which is extracted from the object, where the subscript refers to the type of component. The extracted components can be used as a new *atomic query* condition. For example,  $\pi_{which} object$  denotes the *which* element (Identification Attributes) of the object.

By assembling an *atomic query* and *extraction*, comprehensive query operations can be executed to perform complicated tasks. A **query sequence** consists of a series of atomic query and extract operations which connect end-to-end. Typically, a *query sequence* represents the analytical process of analysts. For example, if the analyst wants to find who rode a taxi that passed the central square, the analyst creates a query sequence that consists of three *atomic operations* and three *extraction operations*. First, the analyst needs to locate the central square, so the  $which \rightarrow object_{POI}$  expression is needed, where “*which*” contains the “ $id=center\ square$ ” condition, and the data source is the POI dataset. Once the position of the central square has been identified, the analyst can then specify a  $where \rightarrow object_{Taxi}$  expression to find taxis that pass the central square. After carefully studying all resulting candidates, the analyst determines which car best matches the specified query condition. Finally the analyst performs the expression  $where + when \rightarrow object_{Person}$ , in which “*where+when*” denotes a spatio-temporal query of the taxi trajectory. This is needed to find the persons who rode in the taxi. The aforementioned process can be summarized as a query sequence presented in (Figure 3).

Note that the queries in our approach can be performed across different data sources by leveraging the spatio-temporal and social inter-connectedness. Only one data source is used in each atomic query operation. Complicated query tasks can be regarded as a boolean combination of a list of atomic query operations. Here, we list the representative query modes:

- **Query in a data source:** When using a single data source, our model is the same as the one proposed in Ferreira et al, [15]. Thus, our model naturally supports  $when + where \rightarrow object$ ,  $((\pi_{which}(when \rightarrow object) \cap \pi_{which}(where \rightarrow object)) \rightarrow object)$  and other tasks, such as origin-destination queries.

- **Origin-destination (OD) query:** An OD query collects all objects that move from an origin to a destination in a data source and can be represented as  $Q_{od} = (\pi_{which}(where_o + when_o \rightarrow object_1) \cap \pi_{which}(where_d + when_d \rightarrow object_1)) \rightarrow object_1$ .
- **Multi-source Query:** A multi-source query can be executed by employing a query sequence in which each query works on a single data source. A typical task is to find objects in data source 1 that matches the time period and the location that overlap with object E in data source 2 occurs:  $Q_1 = (\pi_{where}objectE + \pi_{when}objectE) \rightarrow object_2$ . Likewise, we may want to match objects in data source 1 and data source 2 by time and location. For instance, the query of finding a person P and a taxi rode by the person can be determined by querying from a taxi trajectory dataset and a mobile phone location dataset:  $Q_2 = (\cap_1^N \pi_{which}((\pi_{where}objectP + \pi_{when}objectP) \rightarrow object_{car}))$ , where  $N$  denotes the count of co-occurring data points from the two trajectories.

## 4 SYSTEM

The VAUD interface, Figure 4, consists of two main views: a scene view that shows the situational information as well as the properties of selected objects, and a query view that supports visual reasoning with intuitive drag-and-drop based user interactions.

### 4.1 The Query View

The query view (Figure 4 (b)) uses a flow metaphor to support the construction of cross-domain query tasks by means of drag-and-drop interactions. The flow metaphor represents the process of the analysis (specifying the query conditions, performing querying, analysing the result, extracting conditions and performing new querying).

We design two-tuple nodes and a directed Beziér curve representation to encode and organize components of a flow metaphor. The interface keeps a historical action list to record the user's operations and an information panel to show an overview of data from each dataset.

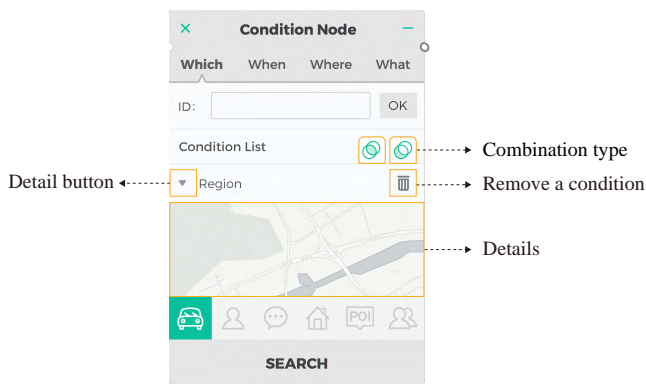


Fig. 5. Visual design for the condition node.

The **condition node** allows the analyst to specify query conditions and data sources (Figure 5). For example, the analyst selects a data source “car” and an entry “ $id=T0230$ ” as the query condition. Selected conditions are displayed on the filter panel. In addition,

the analyst can view the condition details by clicking the detail button, and defining combination types for the conditions. Note that the default combination of selected conditions is *union*.

The **result node** presents the queried items and has a similar visual design as the condition node. The name of each item is automatically assigned by the system but can be modified by the analyst. The analyst can select one, or multiple, items and then view the information of selected elements in the scene view. The result node also includes a statistical chart to support the detailed study of queried items, e.g., a histogram of speed that indicates the traffic situation, or a heatmap that reveals the geographical distribution of vehicles.

As the analysis becomes complex, both the condition node and result node can be folded to get a concise interface. When the condition node is folded, the widgets of the node and its icon are filled with green color, while the color for the result node is white (Figure 6). This color scheme is employed uniformly to match the system color style and distinguish between two nodes.

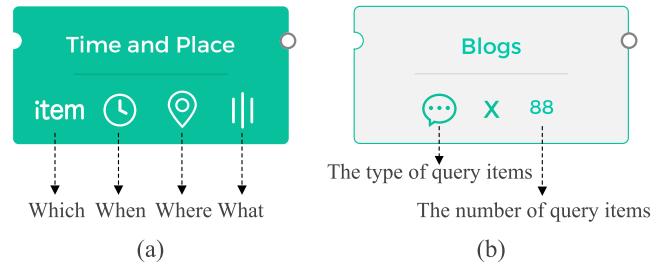


Fig. 6. The node folding for (a) the condition node, and (b) the result node.

The **operations** are encoded with directed links as arrows. There are two types of operations: *retrieving*, which filters items with given conditions, and *reasoning*, which enables user-driven inference such as selecting a geo-tag of a blog post from the scene view and setting it as a query condition. In the *retrieving* operation, the link is modeled as a Beziér curve (Figure 7 (a)). In the *extracting* operation, the link is represented with a dashed Beziér curve (Figure 7 (b)).

The **action list** preserves built operations and the *id* of corresponding nodes. The query condition is also preserved when the analyst adds it to a new condition node. Similarly, a retrieving operation is saved with its data source and associated nodes (Figure 4 (c)).

The **information panel** presents an overview of the data from each dataset. Attributes of data are displayed to show what information is available. Data size and distribution are also preserved to help analysts understand the characteristics of the data.

### 4.2 The Scene View

In the scene view, the road network data is displayed as a list of geometric line primitives. The map and a time control (Figure 4 (f)) are used to provide a visual guide to the constructed STCs. A point of interest (POI), such as a bookstore, school or a shopping mall, is shown with a representative glyph (Figure 4 (e)). The time-varying location information, such as a GPS or mobile phone trajectory, is encoded with polylines.

The scene view keeps a scene list to manage the objects shown on the scene view (Figure 4 (d)). The analyst can freely add objects to the scene list by dragging and dropping objects from the

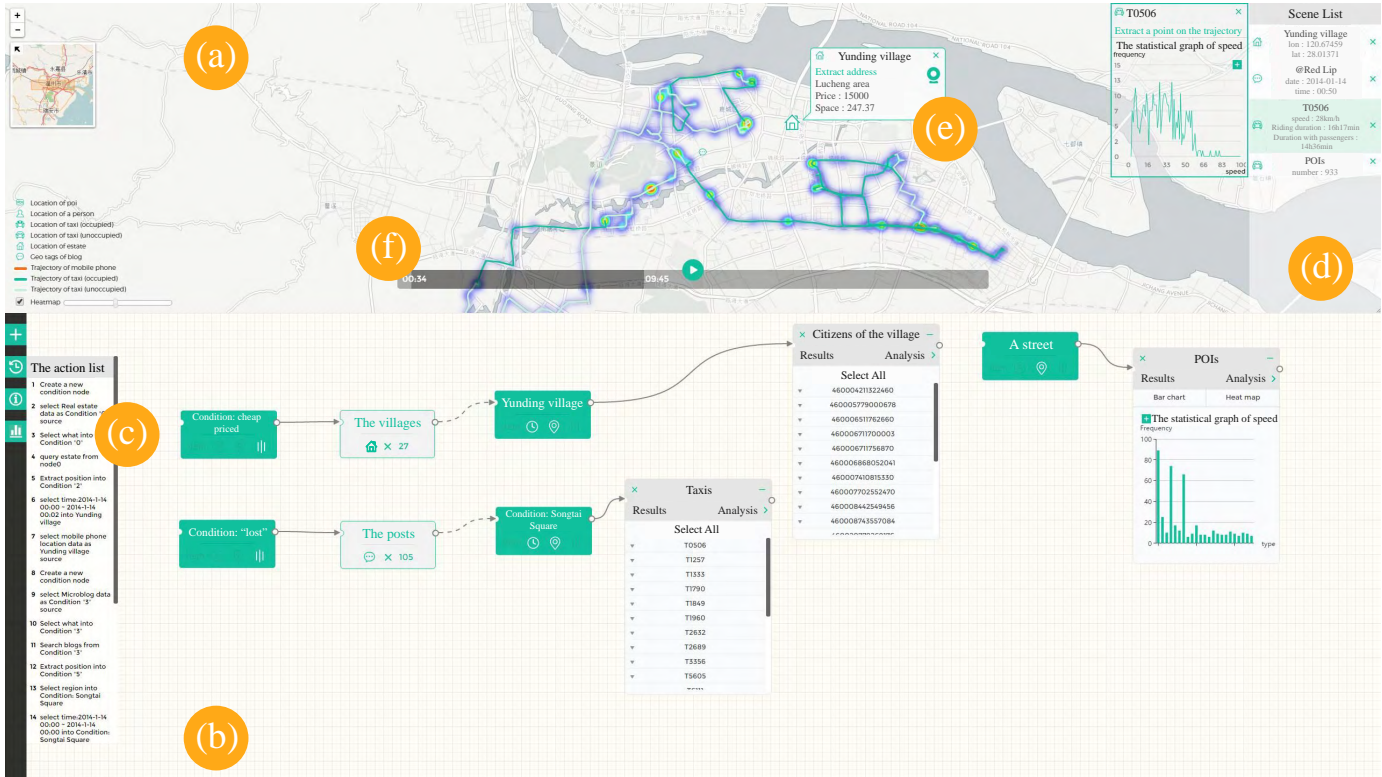


Fig. 4. The interface of VAUD. (a) The scene view. (b) The query view. (c) The action list. (d) The scene list. (e) The icon and its details of a geographical object. (f) The time control.

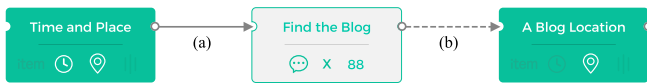


Fig. 7. Visual design for the query operations. (a) A retrieving operator. (b) A extracting operator.

result nodes. When an object is selected from the result nodes, its detailed information is shown in the corresponding visualization scheme on top of the scene view. For different objects, we utilize several different visual representations. For example, if the analyst selects a microblog post from the scene view, the user ID, the text, and the date entry are shown respectively. If a taxi is identified in the scene view, information on the taxi, the time interval specified in the time selector, and the trajectory information are shown. We have also developed specialized widgets to help the analyst. For instance, the time line slider is used to explore the trajectory of a person, and the statistical chart shows the distribution of the selected data.

The **comparison panel** supports the comparison of selected statistical graphs from the scene view. Once a statistical graph of an object is selected, the graph will be added to the comparison panel to help analysts compare with other graphs.

### 4.3 User Interactions

In order to perform a query task, three steps need to be performed: selecting the data source, specifying the condition, and querying the operation. To help the analyst explore the data and infer facts from heterogeneous data sources, drag-and-drop based user interactions are provided.

**Manipulating nodes:** The analyst can create a node by moving a node onto the query view. The position of the nodes can be freely specified by the users. Conceptually, folding or unfolding a node is triggered when a double click takes place. The analyst can rename the node, e.g., “origin-car” can be used to denote a node that is used to find a car passing the origin.

**Specifying conditions:** The analyst sets a query condition by first adding a node in the query view and then specifying the detailed conditions. There are three ways to specify the query condition.

- 1) The analyst may enter a specific condition, for example, to query the POI which is named “central square”, the analyst inputs “central square” and clicks the search button to specify  $id$  = “central square” to be a condition (Figure 8 (a)).
- 2) A set of selection interactions are provided for different data types, for example, the geographical region selection can be used to define a rectangular region of the map to be the “where” condition or a time-picker selection can be used to specify the “when” type condition (Figure 8 (b)). Note that the time-picker selection supports both selecting intervals along a time line model by time-picker and cyclic time model by Time Wheel [12].
- 3) The analyst can specify a condition by dragging items from the result node or from the scene view and dropping them into the condition node (Figure 8 (c)).

Boolean operations can also be applied to the conditions once they are selected.

**Exploring results:** The analyst is able to select one or more objects from the result node and place these in the scene

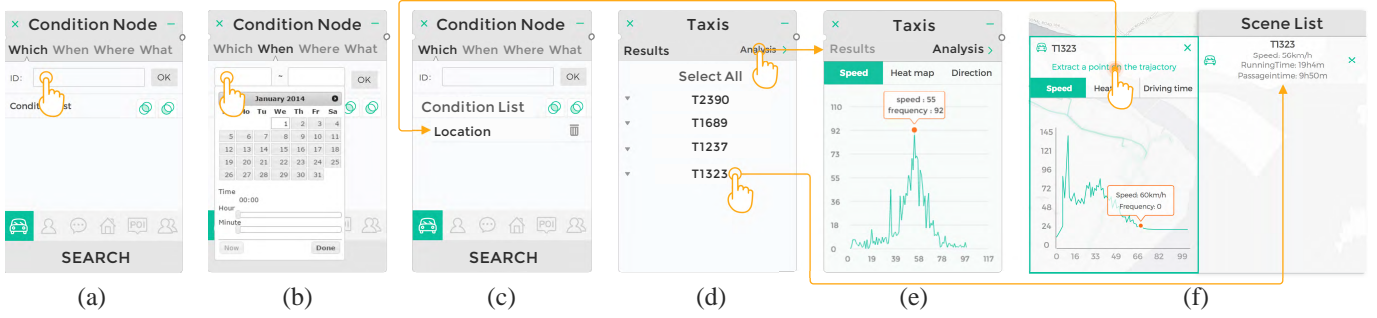


Fig. 8. User interactions. (a) Text input. (b) Item selection. (c) Selection from results; (d) Add an object to the map. (e) View object details. (f) Visual inspection.

TABLE 1  
Visual forms supported in our interface

Data types	Forms
Taxi GPS data	Line chart: Time durations over speed Heatmap: Geographic distribution of taxis Bar chart: Time duration of riding and time duration of occupancy in minutes
Mobile phone location data	Heatmap: Geographic distribution of users
Microblog data	Heatmap: Geographic distribution of posts Line chart: Number of posts over time
Real estate data	Scatter plot: Price and count of real estate
POIs data	Heatmap: The geographic distribution of POIs Bar chart: The statistics of POIs over type
Social network data	Graph: force directed graph of calling network

view (Figure 8 (d)). The analyst can pan and zoom to explore details in the scene view. Furthermore, an analyst can explore detailed information by clicking an object (Figure 8 (e)). Several visualization forms have also been implemented as listed in Table 1 (please refer to the video demo for details). The result node also provides an analysis panel to illustrate the query item characteristics (Figure 8 (f)).

#### 4.4 Data Processing Details

**Data preprocess:** The locations of each mobile user are recorded with respect to the cell towers their phones' access. As such, their geographical locations are relatively inaccurate, and a set of data preprocessing operations, including ping-pong effect removal, noise removal, duplication removal and missing value completion, are performed. Note that the taxi GPS trajectories have a much higher accuracy than the mobile phone data. The social network links are also extracted from the mobile phone call records. Specifically, a call detail record (CDR) is a data record containing the phone numbers of both users, the start time, and the duration of the call. A graph structure that denotes the inter-user relationships among mobile phone users is extracted based on the communication network of the mobile phone call records. The social media information comes from a Microblog web service. The street view of locations in the city are linked to Baidu map services. Similarly, the POI dataset is downloaded from an online location-based service provider. The mobile phone location data and the taxi trajectories are spatio-temporal and are stored as a list of pairs of time and locations.

The use of mobile phone location data has an accuracy of 0.5 km depending on the cell tower coverage in the area. The average accuracy is 0.269 km, 86.96% are below 0.5 km. As the locations of the cell towers are provided, when performing a spatial query over mobile phone location data, we first find cell

towers in the condition region, then put mobile phones linked to the cell towers as query result. Although the queried locations are coarse-grained, they do provide adequate information for locating a mobile phone in general dense urban regions. With spatially fine-grained data like taxi trajectory data, this inaccuracy can be compensated by cross-linking multiple data sources, as demonstrated in Section 5.1.

**Data storage:** To enable cross-domain analysis by leveraging the spatio-temporal inter-connectedness, we build a sequence of STCs for spatio-temporal objects. In our implementation, the time period is sliced on the basis of days. The STC sequence contains 22 items over 22 days data. The resolution of an STC for one day is  $300 \times 300 \times 1440$ , where 300 denotes the resolution along the longitude and latitude, and 1440 denotes a time interval of 1 minute. We choose  $300 \times 300$  because our data covers a city. If we divide this city into  $300 \times 300$  regions, each region will approximately cover a single street block. During the process of analysis, spatial restrictions are always beyond or equal to a street block. Note that each constructed STC is structurally sparse because there are many empty cells that are not covered by spatio-temporal objects. For example, an STC for the trajectories of 1 million persons in one day has 73.16% of its cells empty.

The average memory consumption of an STC is 5Gb. Therefore, the total consumption for 22 STCs is about 110Gb. We store all STCs individually in the harddisk and construct a spatio-temporal index structure to accelerate the online query. The indices store pointers to the 3D cell locations of each STC and can be used to quickly retrieve objects on a specific cell (a location and a time point). To further enable on-the-fly query of objects and associated attributes, we store the data of all objects individually as files in the harddisk and construct an array of pointers to the files.

The memory consumption for the two index structures (STC based and object based) is about 100Mb. During runtime, our system loads two index structures and supports on-the-fly querying from both the STC based representation and the object based representation. The performance achieves interactive rates in our experiments.

## 5 CASE STUDIES

The experimental platform is an Intel Xeon ES430 2.66 GHz desktop that is configured with 16 GB of main memory. Our case studies were designed based on the interviews with several experts in city planning and public security and bilateral collaborations over one year. We designed and performed several case studies for

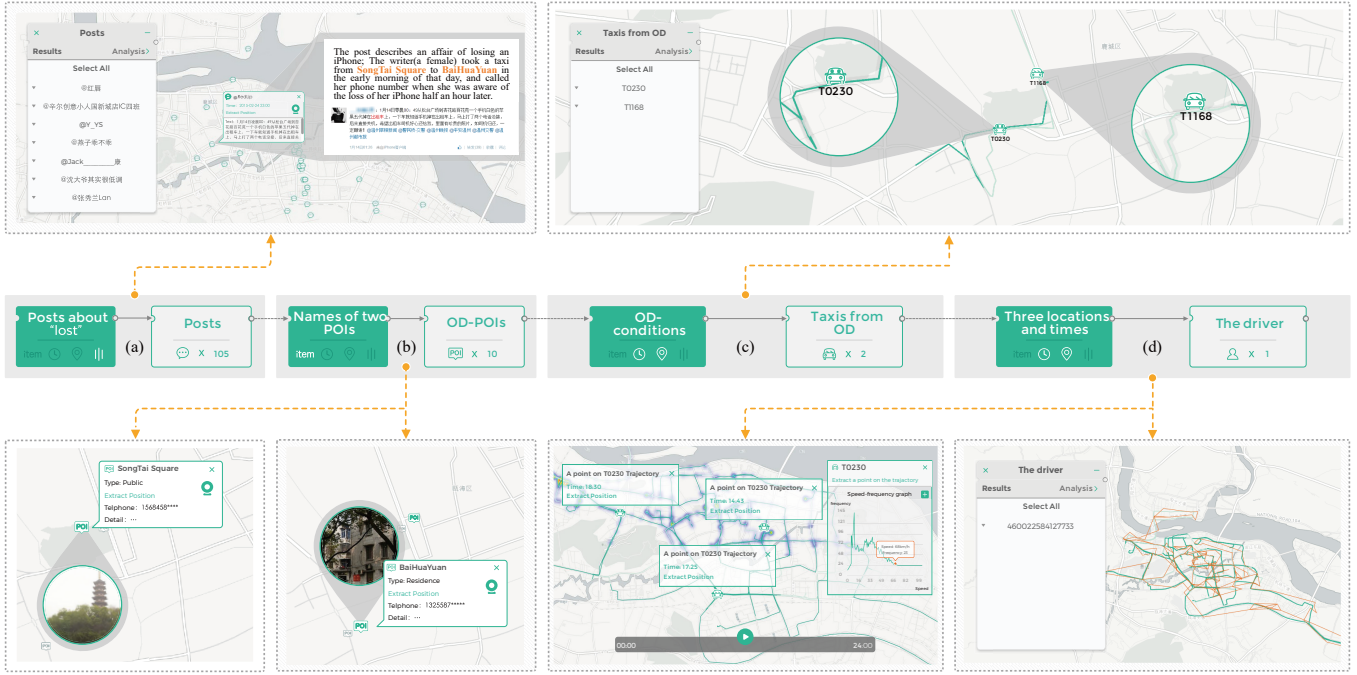


Fig. 9. (a) The analyst inputs “lost” as the keyword and performs a query over a Microblog dataset. The analyst explores the queried Microblog posts and identifies a post in which an occupant described the loss of her iPhone while taking a taxi. The post contains the time and location of the origin and destination of the taxi ride. (b) The analyst identifies the origin and destination geo-coordinates by searching the location-description-tags of the origin and destination of the taxi ride from the post. (c) Based on this information, the analyst interactively queries taxis that were near the points of interest near the times noted in the blog post. The analyst carefully studies the trajectories of each taxi in the scene view and identifies a taxi which drives a passenger from SongTai square to BaiHuaYuan. It is deduced by the analyst to be the one taken by the writer. (d) Next, to further refine the search, the analyst finds the phone numbers of the identified taxi drivers by performing a spatio-temporal query over the mobile phone location dataset. Finally, the analyst compares the trajectory of the taxi and the phones and confirms the phone number of the taxi driver who can then be contacted to retrieve the missing phone.

various analysis tasks. Note that our emphasis is on the efficiency of visual analysis and reasoning features. Please refer to the supplementary video for more details.

### 5.1 Case 1: Finding the lost phone

Our first case study explores the retrieval of a lost iPhone as a demonstration of how multiple datasets can be linked to explore information that would not be captured in a single dataset alone. Figure 9 illustrates the query procedures that are designed to locate the missing phone using seven critical steps.

- 1) The analyst chooses to explore the MicroBlog posts and focuses on seeing what items have recently been noted as missing in the city. The analyst inputs “lost” as the keyword in a new condition node and then selects the Microblog dataset as the source and performs the query (Figure 9 (a)). The analyst explores the query results and notices an interesting post from 00:49 am. The post describes losing an iPhone: the blog writer took a taxi from SongTai Square to BaiHuaYuan early that morning and later realized she had misplaced her phone between leaving her home and arriving at her destination. She believed she dropped her phone in the taxi.
- 2) Given that she rode in a taxi and that the origin and destination (SongTai Square and BaiHuaYuan) were provided in the blog, the analyst believes that the taxi cab can be identified and contacted to see if the blog writer left her phone in the car. To get the origin and destination geo-coordinates, the analyst

performs a *which*  $\rightarrow$  *object<sub>POI</sub>* query where “*which*” is the location-description-tags of the origin and destination of the taxi ride from the post (Figure 9 (b)). In this way, the points-of-interest (SongTai Square and BaiHuaYuan) can be transformed into geo-coordinates. The analyst browses the Street view of SongTai Square and finds that it is a tourist attraction.

- 3) The Origin-Destination (OD) from the POI dataset gives latitude and longitude, and the blog post provides a time of day. Next, to locate taxis that are near the points of interest in the early morning, an OD query  $\cap_{i=1}^2 (where_i + when_i) \rightarrow object_{Taxi}$  is performed against the taxi’s trajectory data to locate potential taxis the blog writer may have ridden in (Figure 9 (c)).
- 4) A set of taxis the writer potentially rode in is detected. The analyst carefully studies the trajectories of each taxi in the scene view and identifies a taxi which takes a passenger from SongTai square to BaiHuaYuan. It is deduced by the analyst to be the one taken by the writer.
- 5) The analyst then wants to find the phone number of the taxi driver. The analyst locates three points on the taxi’s trajectory and performs a spatio-temporal query  $\cap_{i=1}^3 (\pi_{where point_i} + \pi_{when point_i}) \rightarrow object_{MobilePhoneLocation}$  over the mobile phone location dataset by matching the trajectories of the taxi and the mobile phones (Figure 9 (d)).
- 6) Finally, the analyst compares the trajectory of the taxi and the mobile phone and identifies the phone number of



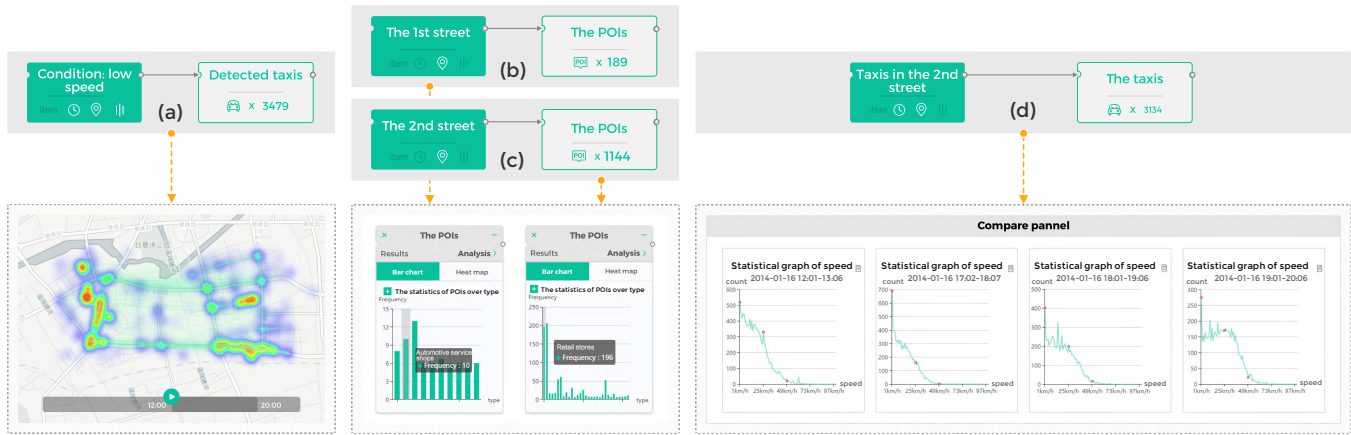


Fig. 10. The second case study analyzes a traffic jam. (a) The analyst identifies two primary streets that appear to have a large amount of congestion. (b) The analyst explores POIs near the first street and finds that the first street has many automotive service shops where taxis are going for maintenance in the afternoon. (c) The analyst explores POIs near the second street and finds that the second street is next to a commercial centre as its surrounding POIs are almost all retail stores. (d) The analyst studies when the congestion begins and ends.

the taxi driver who can then be contacted to ask if a cell phone was left in the car.

In this case, the analyst queries from the MicroBlog data, POI data, taxi trajectory data, and mobile phone location data. The analyst successfully reviews a case of “lost-phone” and finally helps the MicroBlog writer find the phone number of the taxi driver through OD queries, trajectory matching, and a series of multi-source heterogeneous queries.

## 5.2 Case 2: Analyzing a traffic jam

While the first case study demonstrates how linking multiple data sources can provide hidden information, we can also use such data to aid in urban planning and design. In this case study, the analyst wants to explore traffic flows and patterns, specifically focusing on congestion and traffic jams. Here (Figure 10), a series of four queries were developed to explore traffic congestion.

- 1) First, to identify where traffic jams are most likely to occur, the analyst preforms a *what + when + where*  $\rightarrow$   $object_{Taxi}$  where “what” is the speed range “0-20 km/h” and “during 12:00-20:00 2014-1-16” is a single day of the week (Figure 10 (a)). The analyst carefully checks the heatmap view of the taxi trajectories and notices two primary streets that appear to have a large amount of congestion.
- 2) To explore what might be causing the congestion, the analyst does a *where*  $\rightarrow$   $object_{POI}$  to see what businesses or other city structures are located in the nearby areas. The analyst finds that the first street that appears congested has many automotive service shops where taxis are going for maintenance in the afternoon (Figure 10 (b)). The second street is next to a commercial centre as its surrounding POIs are almost all retail stores (Figure 10 (c)).
- 3) The analyst wants to know more about the second street’s traffic situation, specifically when the congestion begins and ends. The analyst performs a *where + when*  $\rightarrow$   $object_{Taxi}$  for taxis passing the major intersection, during “2014-1-16 12:00-21:00” (Figure 10 (d)).

- 4) The analyst then studies the statistical graph of speed returned by the query and observes that the street is still crowded from 12:00 to 19:00 pm with the assistance of the Compare panel.

Case 2 analyzes traffic flows via different statistical graphs and heat maps which demonstrates that our system helps analysts observe distributions of data and multiple types of information across the city.

## 5.3 Case 3: Comparing the daily lives of humans

We can also use our system to compare and contrast the everyday lives of different citizens. In this case study, the analyst explores the world of two groups of people who live in different villages (Figure 11):

- 1) First, the analyst queries from the real estate dataset by using the house price as a query condition and identifies two villages characterized by high and cheap housing prices respectively (Figure 11 (a)). The analyst identifies a high-priced village located in downtown and a cheap-priced village located in suburbs to compare the differences about these two kinds of villages. The analyst browses the street view and finds that the high-priced village is with high-rise buildings and a river, besides, the cheap village is with mid-rise buildings and nearby a expressway.
- 2) For each village, the analyst identifies citizens of the villages by performing *where + when*  $\rightarrow$   $object_{MobilePhoneLocation}$  queries in the mobile phone location dataset (Figure 11 (b)). The “when” condition is set to be in the time period from [0:00 am, 6:00 am from Jan. 1st to 7th] to ensure that the queried persons live in the villages.
- 3) The analyst studies the trajectories of two groups of citizens respectively and finds that these individuals live in cheap apartment works in downtown and southwest of the city and have large scale trajectories. In contrast, all the citizens in the high-priced village seem to work in downtown.



Fig. 11. The third case study seeks to study human mobility patterns. (a) The analyst identifies two villages characterized by high and cheap housing prices that located in downtown and suburbs respectively. The analyst browses the street view and identifies citizens of the villages. (b) The analyst observes the trajectories of two group of citizens and finds that these individuals live in cheap apartments and work in downtown and southwest of the city. In contrast, all the citizens in the high-priced village seem to work in downtown. (c) For each group of citizens, the analyst queries their social network and finds that citizens in the high-price village appear to have more social connections than those in the cheap village (d) The analyst studies the POIs near the two villages.

- 4) For each group, the analyst queries their social network from the social network dataset by means of *which*  $\rightarrow$  *object<sub>SocialNetwork</sub>* queries (Figure 11 (c)). The analyst finds that the persons in the high-price village have more social connections than those in the cheap village.
- 5) The analyst then queries POIs from the POI dataset by using *where* + *when*  $\rightarrow$  *object<sub>POI</sub>* queries respectively, in which *where* is the neighbors of the two villages (Figure 11 (d)). The analyst studies the queried POIs and finds that the POIs associated with the cheap village’s neighborhood are factory-related. In contrast, the high-priced village is located in a commercial streets.

We use Case 3 to analyze the behaviors of people and compares the daily lives of two groups of citizens who live in different villages. The distribution of POI types and social network graphs effectively show different kinds of life styles in the city.

## 6 USER STUDY

We performed a user study with 14 CS students (3 females and 11 males, ages 20 to 30) to evaluate whether our system is helpful in analyzing cross-domain urban data. First, we introduced our interface to the participants and showed a case study with a 3-minutes video to explain our query workflow. Then, our participants were asked to use our system to analyze real urban data described in Section 3. The user-study tasks were:

- **T1:** Find the harbor named “AnLanTing”. During T1,  $Q_{T1} = \text{which} \rightarrow \text{object}_{POI}$  should be preformed where “*which*” represents “AnLanTing”.

- **T2:** Analyze the traffic situation that occurred in the morning on January 10 and locate the most congested crossing around the harbor. In this task, participants should perform a  $Q_{T2} = (\pi_{\text{where}}(\text{object}_{POI}) \cap \text{when}) \rightarrow \text{object}_{Taxi}$  and check the heatmap view of the taxi trajectories.
- **T3:** Find major architectural types surrounding the harbor using the POI and blog data. Desired queries could be represented as  $Q_{T3a} = \pi_{\text{where}}(\text{object}_{POI}) \rightarrow \text{object}_{Blog}$  and  $Q_{T3b} = \pi_{\text{where}}(\text{object}_{POI}) \rightarrow \text{object}_{POI}$ .

These tasks were designed to analyze how a user selects different types of conditions to query cross-domain urban data. In addition, multiple types of visualizations are needed to assist the analyst in obtaining information from the multi source data. Moreover, these three tasks have ground truth so that we can compute the accuracy rate of each task. After completing the tasks, the participants were asked to evaluate our system with respect to three aspects:

- **A1:** The importance of knowing all the attributes of the data.
- **A2:** The convenience of our system for querying cross-domain urban data.
- **A3:** The effectiveness of our visualizations of cross-domain urban data.

For each aspect, subjects were assigned a grade of “Poor”, “Average”, “Good”, or “Very Good”.

Figure 12(a) shows the accuracy of the three tasks. All participants found the correct answer to T1, while the accuracy of T2 and T3 were 85.8% and 78.6% respectively. By analyzing participant’s answer sheets, we identified several common mistakes when they analyzed the urban data. The first mistake

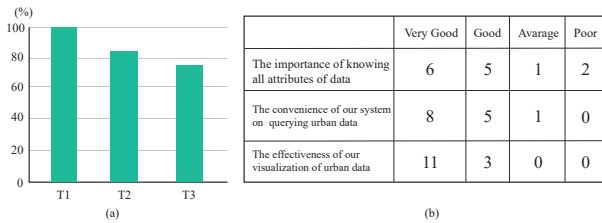


Fig. 12. The results of the user study. (a) The average accuracy of all participants in T1, T2, T3. (b) Participants' scores of the three aspects using ratings of "Poor", "Average", "Good", and "Very Good"

was that when they chose the surrounding region of the harbor, the selected range was too large to accurately cover the neighboring POI of the harbor. Meanwhile, some participants did not use the heat maps of taxis to analyze the traffic situation thus wasting lots of time. Moreover, participants always forgot to set temporal restrictions which led to wrong results.

Figure 12(b) shows the grades of the three aspects evaluated by the participants. Many participants considered the structure of each dataset as necessary information, while the others thought it was somewhat helpful but not necessary. Most participants were satisfied with our system while two participants want us to improve interactions on condition selection and visual encoding to help guide the analysis of urban data.

Along with the user study, we also interviewed participants about the pros and cons of our system. All participants stated that our system would be useful for urban data visual analysis, and our user study demonstrated that most of the participants could effectively complete common urban analytic tasks with our system. Some key quotes from the interviews include: "The system shows orderly information to help me analyze the tasks. It is necessary to list all attributes of each urban dataset in the information table, which can assist in querying information;" "The system is well designed and the interactions are simple and easy to understand," and; "It is convenient to query cross-domain urban data in a unified query interface."

In addition, we also solicited comments from the participants with respect to the user interactions, analysis process, and visualization performance. The participants provided many reasonable suggestions for improvement. "Some of the interactions are not reasonable, such as when I choose a condition, I must click the 'OK' button which interrupts the flow of the analysis." "It is confusing when selecting 'which' or 'what' condition, I don't know what exact information they represent."

Users also noted that they wanted specific guidance about data fields and information to help them in their analysis. "I often don't know what to do next. If there are some information prompts to help me choose the direction of analysis, the system will be better." One participant also noted issues with the performance of the rendering parts. "The system provides a fast query of urban data. However, as the data size increases, rendering and interaction with the data become slower." Such comments indicate the need for more research into a combination of intelligent tutoring and visualization and even further research into how to generate performance increasing for exploring large urban data.

Additionally, we interviewed several domain experts from the traffic safety field and the urban planning field after using our system. We asked them to analyze the traffic jam problem using our system and provide feedback. Overall, the experts had a great interest in visual analysis over cross domain urban data: "Cross

domain data provides huge amounts of information about our city. I think this system is a good application to visualize cross domain data. Their query model fits with the reasoning procedure we usually use. One can analyze urban data step by step and get information about whatever you want." While our system deals with off-line data, experts suggested that we could deploy VAUD on real time data. "VAUD can be used to cope with event analysis, policy making, etc. If it is able to deal with streaming data, it will be useful in a City Surveillance System."

## 7 DISCUSSION AND FUTURE WORK

In this paper, we present VAUD, a visual analysis framework for exploring and understanding heterogeneous urban data. A visually assisted query model is introduced as a foundation for interactive exploration coupled with simple, yet powerful, structural abstractions and reasoning functionalities. By leveraging the spatio-temporal and social inter-connectedness, VAUD achieves high efficiency in terms of storage, query and analysis. VAUD extends conventional visual analytics approaches for citywide urban data to the cyber-physical-social system context and empowers users by allowing for interactive multi-source querying from real-time social and physical data. The implemented system enables geo-spatial, social-network, temporal, statistical, and structured and unstructured analysis, providing a context-rich analytic experience for users. The case studies and user studies demonstrate the unique capabilities of our approach.

For future work, we plan to research automatic algorithms that can recommend potential analysis directions of urban data exploration. We also intend to design flexible interactions to help analyst select query conditions. Last but not least, we will focus on handling streaming data within our system, which requires an efficient way of representing and transferring the online data into our visual analysis pipeline.

## ACKNOWLEDGMENTS

This research has been sponsored in part by the National 973 Program of China (2015CB352503), Major Program of National Natural Science Foundation of China (61232012) and National Natural Science Foundation of China (61422211, 61772456).

## REFERENCES

- [1] S. Al-Dohuki, Y. Wu, F. Kamw, L. Xin, L. Xin, Z. Ye, X. Ye, C. Wei, M. Chao, and W. Fei. Semantictraj: A new approach to interacting with massive taxi trajectories. *IEEE Transactions on Visualization & Computer Graphics*, 23(1):11–20, 2017.
- [2] G. Andrienko and N. Andrienko. Spatio-temporal aggregation for visual analysis of movements. In *IEEE Symposium on Visual Analytics Science and Technology*, pages 51–58, 2008.
- [3] G. Andrienko, N. Andrienko, C. Hurter, S. Rinzivillo, and S. Wrobel. Scalable analysis of movement data for extracting and exploring significant places. *IEEE Transactions on Visualization and Computer Graphics*, 19(7):1078–1094, 2013.
- [4] N. Andrienko and G. Andrienko. Exploratory spatiotemporal visualization: An analytical review. *Journal of Visual Languages and Computing*, 14(6):503–541, 2003.
- [5] N. Andrienko and G. Andrienko. Visual analytics of movement: An overview of methods, tools and procedures. *Information Visualization*, 12(1):3–24, 2013.
- [6] S. Arietta, A. A. Efros, R. Ramamoorthi, and M. Agrawala. City forensics: Using visual elements to predict non-visual city attributes. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2624–2633, 2014.

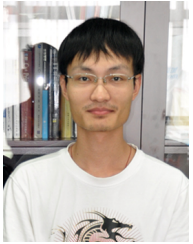
- [7] M. Chein and M.-L. Mugnier. *Graph-based Knowledge Representation: Computational Foundations of Conceptual Graphs*. Springer Science & Business Media, 2009.
- [8] W. Chen, F. Guo, and F. Y. Wang. A survey of traffic data visualization. *IEEE Transactions on Intelligent Transportation Systems*, 16(6):2970–2984, 2015.
- [9] Z. Chen, H. T. Shen, X. Zhou, Y. Zheng, and X. Xie. Searching trajectories by locations: An efficiency study. In *Proceedings of ACM Special Interest Group on Management of Data*, pages 255–266, 2010.
- [10] T. Crnovrsanin, I. Liao, Y. Wu, and K.-L. Ma. Visual recommendations for network navigation. *Computer Graphics Forum*, 30(3):1081–1090, 2011.
- [11] H. Doraiswamy, N. Ferreira, T. Damoulas, J. Freire, and C. T. Silva. Using topological analysis to support event-guided exploration in urban data. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2634–2643, 2014.
- [12] R. Edsall and D. Peuquet. A graphical user interface for the integration of time into gis. In *Proceedings of the 1997 American congress of surveying and mapping annual convention and exhibition, Seattle, WA*, pages 182–189, 1997.
- [13] J.-D. Fekete and C. Silva. Managing data for visual analytics: Opportunities and challenges. *IEEE Data Engineering Bulletin*, 35(3):27–36, 2012.
- [14] E. S. Ferguson. *Engineering and the mind's eye*. MIT Press, 1994.
- [15] 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 Transactions on Visualization and Computer Graphics*, 19(12):2149–2158, 2013.
- [16] R. K. Florian Haag and T. Ertl. Vespa: A pattern-based visual query language for event sequences florian. *International Conference on Information Visualization Theory & Applications*, pages 48–59, 2016.
- [17] D. Guo, J. Chen, A. MacEachren, and K. Liao. A visualization system for space-time and multivariate patterns (VIS-STAMP). *IEEE Transactions on Visualization and Computer Graphics*, 12(6):1461–1474, 2006.
- [18] D. Guo and X. Zhu. Origin-destination flow data smoothing and mapping. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):2043–2052, 2014.
- [19] M. Haklay and P. Weber. OpenStreetmap: User-generated street maps. *IEEE Pervasive Computing*, 7(4):12–18, 2008.
- [20] T. Kapler and W. Wright. Geotime information visualization. *Information Visualization*, 4(2):136–146, 2005.
- [21] J. Kehrer and H. Hauser. Visualization and visual analysis of multi-faceted scientific data: A survey. *IEEE Transactions on Visualization and Computer Graphics*, 19(3):495–513, 2013.
- [22] M.-J. Kraak. The space-time cube revisited from a geovisualization perspective. In *International Cartographic Conference*, pages 1988–1996, 2003.
- [23] J. Lee, B. Bagheri, and H.-A. Kao. Recent advances and trends of cyber-physical systems and big data analytics in industrial informatics. In *IEEE International Conference on Industrial Informatics*, pages 1–6, 2014.
- [24] X. Li and M.-J. Kraak. New views on multivariable spatiotemporal data: the space time cube expanded. In *International Symposium on Spatio-temporal Modelling, Spatial Reasoning, Analysis, Data Mining and Data Fusion*, pages 199–201, 2005.
- [25] L. Lins, J. T. Klosowski, and C. Scheidegger. Nanocubes for real-time exploration of spatiotemporal datasets. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2456–2465, 2013.
- [26] I. Lopez, R. Snodgrass, and B. Moon. Spatiotemporal aggregate computation: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 17(2):271–286, 2005.
- [27] Y. Ma, T. Lin, Z. Cao, C. Li, F. Wang, and W. Chen. Mobility viewer: An eulerian approach for studying urban crowd flow. *IEEE Transactions on Intelligent Transportation Systems*, 17(9):2627–2636, 2016.
- [28] N. Mahyar and M. Tory. Supporting communication and coordination in collaborative sensemaking. *IEEE Transactions on Visualization and Computer Graphics*, 19(3):75–82, 2014.
- [29] J. L. Mennis, D. J. Peuquet, and L. Qian. A conceptual framework for incorporating cognitive principles into geographical database representation. *International Journal of Geographical Information Science*, 14(6):501–520, 2000.
- [30] T. C. S. H. A. G. A. N. Stacking-based visualization of trajectory attribute data. *IEEE Transactions on Visualization and Computer Graphics*, 18(12):2565–2574, 2012.
- [31] D. J. Peuquet. It's about time: A conceptual framework for the representation of temporal dynamics in geographic information systems. *Annals of the Association of American Geographers*, 84(3):441–461, 1994.
- [32] M. A. Sakr and R. H. Güting. Group spatiotemporal pattern queries. *Geoinformatica*, 18(4):699–746, 2014.
- [33] R. Shadoan and C. Weaver. Visual analysis of higher-order conjunctive relationships in multidimensional data using a hypergraph query system. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2070–2079, 2013.
- [34] S. Shekhar, M. Coyle, B. Goyal, D. Liu, and S. Sarkar. Data models in geographic information systems. *ACM Communications*, 40(4):103–111, 1997.
- [35] Z. Shen, K.-L. Ma, and T. Eliassi-Rad. Visual analysis of large heterogeneous social networks by semantic and structural abstraction. *IEEE Transactions on Visualization and Computer Graphics*, 11(3):1427–1439, 2006.
- [36] B. Shneiderman. Dynamic queries for visual information seeking. *Craft of Information Visualization*, 11(6):14–21, 2003.
- [37] W. Tang, H. Zhuang, and J. Tang. Learning to infer social ties in large networks. In *Machine Learning and Knowledge Discovery in Databases*, pages 381–397. Springer, 2011.
- [38] F. Wang, W. Chen, F. Wu, Y. Zhao, H. Hong, T. Gu, L. Wang, R. Liang, and H. Bao. A visual reasoning approach for data-driven transport assessment on urban road. In *Proceedings of IEEE Visual Analytics Science and Technology*, pages 2149–2158, 2014.
- [39] F. Wang, W. Chen, Y. Zhao, T. Gu, S. Gao, and H. Bao. Adaptively exploring population mobility patterns in flow visualization. *IEEE Transactions on Intelligent Transportation Systems*, PP(99):1–10, 2017.
- [40] Z. Wang, M. Lu, X. Yuan, J. Zhang, and H. v. d. Wetering. Visual traffic jam analysis based on trajectory data. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2159–2168, 2013.
- [41] Z. Wang, T. Ye, M. Lu, X. Yuan, H. Qu, J. Yuan, and Q. Wu. Visual exploration of sparse traffic trajectory data. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1813–1822, 2014.
- [42] W. Zeng, C. W. Fu, S. M. Arisona, A. Erath, and H. Qu. Visualizing mobility of public transportation system. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1833–1842, 2014.
- [43] E. Zraggen, S. M. Drucker, D. Fisher, and R. Deline. (s|q)eries: Visual regular expressions for querying and exploring event sequences. In *ACM Conference on Human Factors in Computing Systems*, pages 2683–2692, 2015.
- [44] J. Zhang, Y. E. J. Ma, Y. Zhao, B. Xu, L. Sun, J. Chen, and X. Yuan. Visual analysis of public utility service problems in a metropolis. *IEEE Transactions on Visualization and Computer Graphics*, 20(12):1843–1852, 2014.
- [45] X. Zheng, W. Chen, P. Wang, D. Shen, S. Chen, X. Wang, Q. Zhang, and L. Yang. Big data for social transportation. *IEEE Transactions on Intelligent Transportation Systems*, 17(3):620–630, 2016.
- [46] Y. Zheng, L. Capra, O. Wolfson, and H. Yang. Urban computing: Concepts, methodologies, and applications. *ACM Transaction on Intelligent Systems and Technology*, 5(3):1–55, 2014.
- [47] Y. Zhu, A. Fathi, and F.-F. Li. Reasoning about object affordances in a knowledge base representation. In *European Conference on Computer Vision*, pages 408–424. Springer, 2014.



**Wei Chen** is a professor at the State Key Lab of CAD&CG, Zhejiang University. His research interests is on visualization and visual analysis, and has published more than 30 IEEE/ACM Transactions and IEEE VIS papers. He actively served as guest or associate editors of IEEE Transactions on Visualization and Computer Graphics, IEEE Transactions on Intelligent Transportation Systems, and Journal of Visualization. For more information, please refer to <http://www.cad.zju.edu.cn/home/chenwei/>



**Zhaosong Huang** received his B.S. in computer science from the Shandong University, China in 2015. He is currently a PhD student in the College of Computer Science and Technology at the Zhejiang University of State Key Lab of CAD&CG. His research interests include visualization and visual analysis of urban data.



**Feiran Wu** received the Ph.D. degree in Computer Science and Technology from Zhejiang University, China, in 2016. His research interests include information visualization and visual analytics.



**Minfeng Zhu** received his B.S. in mathematics from the Zhejiang University, China in 2015. He is currently a PhD student in the College of Computer Science and Technology at the Zhejiang University. His research interests include visualization and visual analysis of urban data.



**Huihua Guan** Huihua Guan received her B.S. in computer science from the Zhejiang University, China in 2015. She is currently a Master student in the College of Computer Science and Technology at the Zhejiang University. Her research interests include information visualization, visual analysis and human-computer interaction



**Ross Maciejewski** is an Associate Professor at Arizona State University in the School of Computing, Informatics & Decision Systems Engineering. His primary research interests are in the areas of geographical visualization and visual analytics focusing on public health, dietary analysis, social media, criminal incident reports, and the food-energy-water nexus. He is the a recipient of an NSF CAREER Award (2014) and was recently named a Fulton Faculty Exemplar and Global Security Fellow at Arizona State. For

more information, visit <http://vader.lab.asu.edu>.