CheetahVIS: A Visual Analytical System for Large Urban Bus Data

Wentao Ning[‡], Qiandong Tang[‡], Yi Zhao[‡], Chuan Yang[‡], Xiaofeng Wang[‡], Teng Wang[‡],
Haotian Liu[‡], Chaozu Zhang[‡], Zhiyuan Zhou[‡], Qiaomu Shen[‡], Bo Tang[‡§]*
[‡] Department of Computer Science and Engineering, Southern University of Science and Technology
[§] PCL Research Center of Networks and Communications, Peng Cheng Laboratory
tangb3@sustech.edu.cn

ABSTRACT

Recently, the spatial-temporal data of urban moving objects, e.g., cars and buses, are collected and widely used in urban trajectory exploratory analysis. Urban bus service is one of the most common public transportation services. Urban bus data analysis plays an important role in smart city applications. For example, data analysts in bus companies use the urban bus data to optimize their bus scheduling plan. Map services providers, e.g., Google map, Tencent map, take urban bus data into account to improve their service quality (e.g., broadcast road update instantly). Unlike urban moving cars or pedestrians, urban buses travel on known bus routes. The operating buses form the "bus flows" in a city. Efficient analyzing urban bus flows has many challenges, e.g., how to analyze the dynamics of given bus routes? How to help users to identify traffic flow of interests easily?

In this work, we present CheetahVIS, a visual analytical system for efficient massive urban bus data analysis. CheetahVIS builds upon Spark and provides a visual analytical platform for the stakeholders (e.g., city planner, data analysts in bus company) to conduct effective and efficient analytical tasks. In the demonstration, demo visitors will be invited to experience our proposed CheetahVIS system with different urban bus data analytical functions, e.g., bus route analysis, public bus flow overview, multiple region analysis, in a real-world dataset. We also will present a case study, which compares different regions in a city, to demonstrate the effectiveness of CheetahVIS.

PVLDB Reference Format:
Wentao Ning, Qiandong Tang, Yi Zhao, Chuan Yang, Xiaofeng Wang, Teng Wang, Haotian Liu, Chaozu Zhang, Zhiyuan Zhou, Qiaomu Shen, Bo Tang. CheetahVIS: A Visual Analytical System for Large Urban Bus Data. *PVLDB*, 13(12): 2805-2808, 2020.
DOI: https://doi.org/10.14778/3415478.3415480

1. INTRODUCTION

Many smart city applications (e.g., traffic jam detection, citywide crowd prediction) use urban moving objects data, e.g., GPS trajectories of cars, sharing bikes and pedestrians. In this work, we focus on urban bus data, which has been widely used in analytical applications in recent years, e.g., comprehensive bus boarding analysis in Adelaide [1], metro bus data analysis in Los Angeles [4]. Comparing with the trajectories of cars or pedestrians, urban bus data has many different characteristics. The key differences are that i) each operating bus has its own fixed bus route, ii) each bus route has multiple operating buses at the same time, and iii) each operating bus in a given bus route will operate with its scheduled plan. The operating buses form the public bus flows in a city. It is essential to analyze and explore the citywide public bus flows for many applications. We briefly introduce two of them as follows.

Bus Scheduling Optimization: For a given bus route, the number of get-in and get-off passengers at its bus stops are different. The analysis result of bus passengers could be used to optimize the bus scheduling plan in the public bus service company.

Road Updates Detection: The road status in the city changed frequently, e.g., temporary traffic control, emergency traffic accidents. The quality of map service providers (e.g., Google Map, Tencent Map) can be improved by exploiting the analytical results of urban bus data. For example, the map service providers could broadcast road emergency updates to their users instantly when they detect traffic accidents via urban bus data.

There are two major challenges to analyze urban bus data. (I) Large data volume. For example, there are almost 8,000 operating buses in Shenzhen, and generate 15.3 million GPS points and 3.8 million bus trip transactions per day. (II) Insufficient bus data analytical methods. Almost all existing urban data analyzing methods [2, 8] focus on the moving objects which do not have fixed routes, e.g., taxis and pedestrians. In particular, these techniques do not support users to find insights from the overview of urban bus flows, or to model the dynamics of urban operating bus / bus routes over time.

In this demonstration, we present CheetahVIS, a visual analytical system for massive urban bus data. The system builds upon in-memory computation system Spark [1] and enhances the analyzing ability of end users, e.g., data analysts in bus companies, IT engineers in map service providers, even the passengers.

CheetahVIS consists of three major modules, as shown in Figure 1. Data preprocessing module collects heterogeneous and massive urban bus data and preprocesses the collected dataset by exploiting the data cleansing techniques in our previous research paper [5]. It also builds spatial objects index to accelerate spatial query processing. Computation engine module is the core module that builds upon Spark to perform the user analyzing tasks. The

*Corresponding author: Bo Tang.

Proceedings of the VLDB Endowment, Vol. 13, No. 12
ISSN 2150-8097.
DOI: https://doi.org/10.14778/3415478.3415480

[1] https://spark.apache.org/

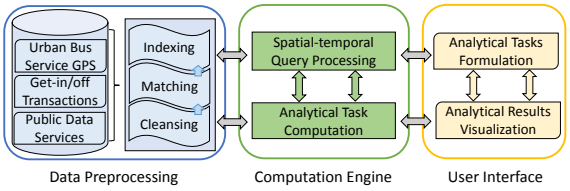

Figure 1: System architecture of CheetahVIS

analyzing tasks on urban bus data are very time consuming naturally as i) massive data size, and ii) complex analyzing queries. We adopted the techniques in two of our previous research works [6, 7] for the important subroutines in CheetahVIS, i.e., similar subtrajectory search in region analysis tasks and time series correlation computation in road analysis tasks. It improves the performance of CheetahVIS system significantly. The Visual Interface Module is implemented upon computation engine, which bridges the gaps between the end users and system provided data analysis functions. This module allows users to issue their analytical tasks by simple operations, which is vital for those users who are not familiar with SQL-like query language. Furthermore, we devised a suite of visualization charts to illustrate the analyzing results effectively. Our demonstration will invite VLDB participants to issue ad-hoc analyzing tasks (e.g., region analysis, bus route analysis) in CheetahVIS, and discover the insights interactively through the visual interface.

The remainder of this paper is organized as follows. We introduce the system architecture of CheetahVIS, and briefly present two techniques, i.e., subtrajectory similarity search [6] and time series correlation computation [7], to accelerate the performance of analytical tasks in Section 2. In Section 3, we describe the visual interface module of CheetahVIS and present the analytical components in it. We conclude by a demonstration proposal with a real-world case study in Section 4.

2. SYSTEM ARCHITECTURE

Figure 1 depicts the system architecture of CheetahVIS. It consists of three modules, i.e., data preprocessing module, computation engine module and visual interface module.

2.1 Data preprocessing module

Data preprocessing module is the fundamental module in CheetahVIS. It collects heterogeneous and massive urban data and integrates them. After that, the urban bus data can be used for further analytical tasks. In general, it has three categories of urban bus data: (1) bus configurations, which include bus schedules, bus routes and corresponding stations; (2) bus trajectories, which are recorded as passing locations of buses. These data are collected from the GPS devices with fixed frequency; (3) passenger dynamics, which are the get-in and get-off records at bus stations. They are collected by the transaction system in public bus service companies.

In many cities (includes Shenzhen), the public bus service only has the passengers' boarding (get-in) transactions, it does not have the departing (get-off) information. The first research problem in data preprocessing module is inferring the correct get-off bus stop for each passenger correctly. We formulate it as get-off location inferring problem in Problem 1.

PROBLEM 1. *Given one-week bus operating GPS trajectory data, one-week passenger transaction data, and all bus stop locations, the get-off location inferring problem is completing the passenger trips by finding the get-off location of each passenger in each trip.*

The general steps for this problem are that: (i) it first blocks the passengers by the bus route, (ii) for all the passengers in the same bus route, it infers the get-off bus stops by the entity matching scheme. For example, suppose the passenger A took bus route 81 at 8:00 at bus stop S_i with forward direction in a weekday, and her next transaction is that she took bus route 81 at 18:00 at bus stop S_j with backward direction. Thus, our system infers the get-off positions of A's trip at 8:00 and 18:00 are S_j and S_i, respectively. The get-off time can be derived by further matching the operating bus IDs in the dataset. We devised a data fusion model to solve the above problem exactly, the technical details are presented in our previous research paper [5].

In addition, data preprocessing module also removes dirty/error data, integrates trajectories with road networks, and constructs R-tree to organize the GPS points in GeoSpark [9].

2.2 Computation engine module

It is the core module in CheetahVIS. It has two major functions: (i) processing spatial-temporal queries and (ii) computing analytical results. For the spatial-temporal query processing, we use SparkSQL to process the queries from the visual interface. For example, it will generate a spatial range query to retrieve all spatial objects (moving buses, bus GPS locations) in a specified region for region analysis tasks as follows.

```
SELECT time, route_id, bus_id, gps_location
FROM bus_gps
WHERE  ST_Contains(region, gps_location)
GROUPBY bus_id
```

The analytical results computing submodule conducts further analysis on the query results. For example, the similar subtrajectory search is a frequent subroutine to identify public bus dynamics (e.g., temporary traffic control). In particular, we model the dynamics of a bus as it does not operate as its schedule (e.g., its own bus route). For example, the bus will change its bus route when the scheduled route is out-of-service with emergence traffic accident. Formally, we formulate the trajectory pattern discovery problem to find the new path of the operating buses in Problem 2.

PROBLEM 2. *Given a set of trajectory data $\mathcal{T}$, i.e., the trajectories of these buses which did not operate as its scheduled bus route, the trajectory pattern discovery problem is finding the most frequent subtrajectory pattern $\mathcal{P}$ in $\mathcal{T}$ with given distance measurement $dist()$.*

The result of Problem 2 can be used to identify the changed path of operating buses in a given bus route, or highlight the changed bus stops in its bus route. Obviously, it is very expensive to discover the most frequent pattern in T by brute force solution. Suppose the subtrajectory length is ξ, the time complexity of brute force solution is $O(n^3+n^2\xi^2)$ where n is the total length of all trajectories in $\mathcal{T}$, as analyzed in [7]. We devised several novel lower bounds (e.g., cell, cross, band-based lower bounds) and integrated them into a group-based framework to accelerate its performance. Through this, it achieves two to three orders of magnitude speedup over brute force solution, our detailed technical contributions are introduced in [7].

Another computation-intensive task is deriving the time series correlation of in-flow and out-flow of operating bus for a given region. We present the time series correlation computation problem in Problem 3.

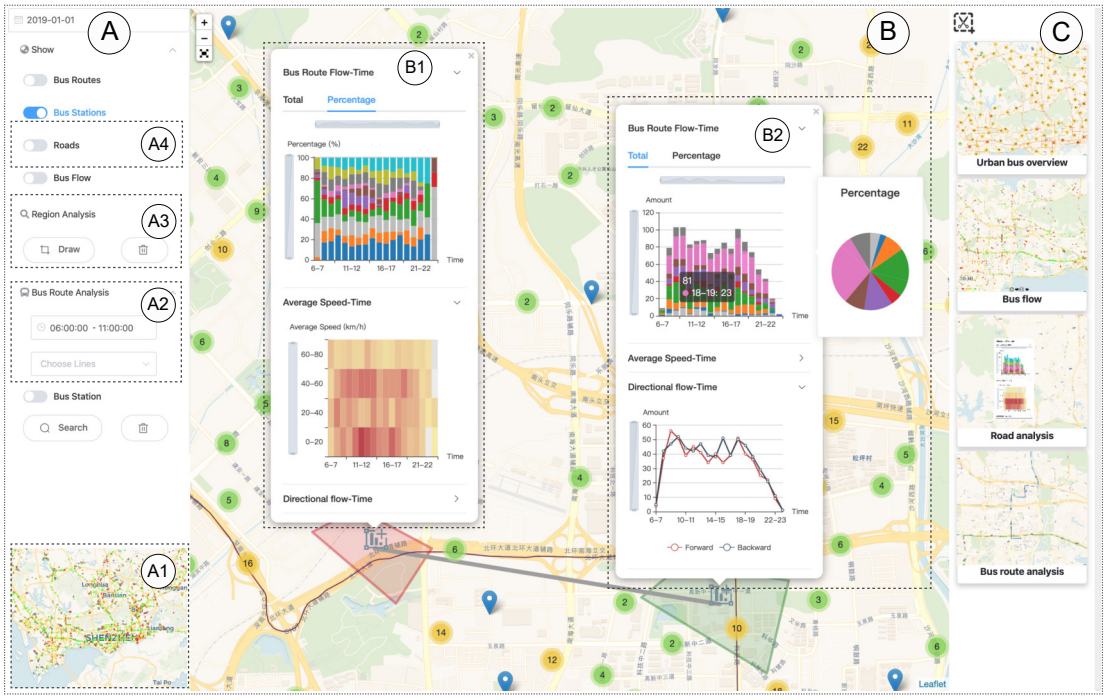

Figure 2: Visual interface of CheetahVIS

PROBLEM 3. *Given two time series T_A and T_B with length l, the Pearson correlation between T_A and T_B is*

$$\rho(T_A, T_B) = \frac{\sum_{i=0}^{l-1} T_A[i]T_B[i] - l\mu(T_A)\mu(T_B)}{l\delta(T_A)\delta(T_B)}.$$

It holds $dist(T_A, T_B) = \sqrt{2l(1-\rho(T_A, T_B))}$, as shown in [3]. It means we can derive the Pearson correlation $\rho(T_A, T_B)$ by computing Euclidean distance between T_A and T_B, i.e., $dist(T_A, T_B)$. We exploit our proposed vectorization techniques in [6] to accelerate the Euclidean distance computation.

2.3 Visual interface module

Figure 2 illustrates the visual interface of CheetahVIS. The left panel, as Figure 2(A) shown, are the visual analytical functions, we will elaborate shortly in Section 3. The main panel is in the middle, users can issue analytical tasks by the functions provided in the left panel, as Figure 2(B) illustrated. The analytical results will be visualized in it by visualization charts. CheetahVIS users could further analyze the visualized results interactively. The right panel, see Figure 2(C), is used to store the visualization charts during the analytical procedure.

3. VISUAL ANALYTICAL FUNCTIONS

In order to provide an effective and efficient visual analytical system for massive urban bus data, we devise coordinate views in CheetahVIS for (i) bus flow analysis, (ii) (multiple) region analysis, (iii) bus route analysis, and (iv) road analysis. We briefly introduce the above analysis functions in this section.

Public bus flows, see (A1) in Figure 2: CheetahVIS provides an overview of public bus flows, its visualization result is illustrated in Figure 3. It shows the runtime status of all buses in the city. The red points show the buses run at low speed (e.g., less than 15km/h), and the green points refer to the buses with high speed (e.g., large than

Figure 3: Bus flows in Shenzhen

40km/h). The low and high speed thresholds can be customized by the users through the slide-bar in Figure 3(A). The trend of the total number of operating buses over time is visualized by the time-serial chart, as shown in Figure 3(B). The bus flow overview helps the users to identify the interests of bus flow easily. For example, when many of the buses are running with low speeds? or when the operating buses increased dramatically?

Bus route analysis, see (A2) in Figure 2 : For the bus routes, user could analyze their runtime status by bus route analysis routine in CheetahVIS. First, the user could query the specified bus routes in any time period. Then, CheetahVIS will simulate the operating buses of the querying routes during the given period in the road map. At last, the number of get-in passengers at each bus stop, the number of operating buses over time, and the time series of forward and backward operating buses are calculated, they can be visualized for further analysis in the charts. Figure 4 shows an example of bus route analysis. The statistics of bus route 81 and 369, i.e., the number of get-in passengers in bus route 81 (see Figure 4(A)) and the time series of forward and backward operating buses of bus route 369 (see Figure 4(B)), are computed and visualized in the charts. It could provide insights of both two bus routes. E.g., as shown in Figure 4(B), the forward and backward operating bus series are negative correlated in bus route 369.

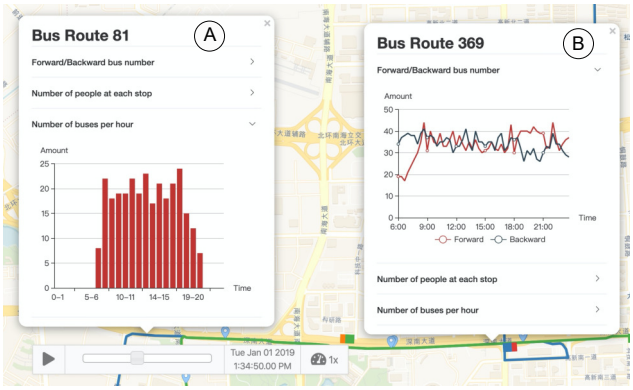

Figure 4: Bus routes analysis

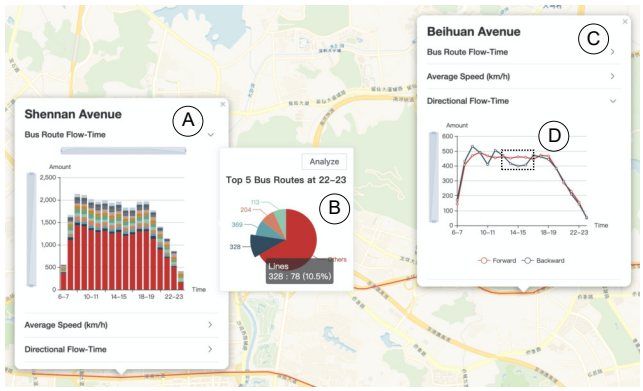

Figure 5: Road analysis

Region analysis, see (A3) in Figure 2: CheetahVIS users can draw one or multiple polygons in the underlying map. The views with dash boundaries (Figure 2 B1, B2) in the main panel presents the analytical results of two selected regions. For each region, CheetahVIS visualizes the statistics by three charts. From the top to bottom, there are (i) the number of buses in different bus routes (both cumulative number and percentage of operating buses per time slot with different granularities), (ii) the average speeds of every operating bus over different time slot granularities, and (iii) the time series of the in-flow and out-flow of operating buses in a given time period, respectively. Users also can compare different regions in the city by dragging these visualization results together.

Road analysis, see (A4) in Figure 2: Road network is the vessel of the modern city. CheetahVIS offers road analysis interface to end users. For example, Figure 5 shows the visualization results of Beihuan Avenue, a popular road in Shenzhen. The number of operating buses for different bus routes over time of that road are visualized in Figure 5(A). For different times, the top-5 bus routes and its percentage over the total operating buses in that time are computed for further analysis (see Figure 5(B)). The direction-flow over time chart (see Figure 5(C)) shows the time series of forward and backward operating bus flows in that road over time. We then conclude that, the forward direction of Beihuan Avenue is slightly busy than backward direction from 13:00-16:00, as shown in Figure 5(D).

4. DEMONSTRATION

This demonstration would be the first public demonstration of CheetahVIS with the real-world dataset to our community. VLDB attendees will experience CheetahVIS from the perspective of an end user (e.g., urban planners or data analysts in map service company). CheetahVIS allows visitors to interact with it for (i) public bus flow analysis, (ii) multiple regions analysis, (iii) bus routes analysis, and (iv) road analysis. The attendees explore the insights with the above analytical result by tuning the low and high speed thresholds, analyzing the dominated bus routes in a road, observing the dynamics of the operating buses, comparing two regions / bus routes, etc.

Besides, to allow visitors to issue their own analytical tasks, we will present a case study that can be used to compare two different urban regions as follows.

- Step 1: choose two different urban regions, and perform urban region analysis on them.
- Step 2: compare the number of operating buses, average speeds and the operating buses in-flow and out-flow simultaneously.
- Step 3: drag the visualization charts of the above two regions together, compute the results of intersection / union of operating buses, and analyze the operating bus correlations.

5. ACKNOWLEDGMENTS

This work was supported by the Science and Technology Innovation Committee Foundation of Shenzhen (Grant No. JCYJ20180302174301157), the Guangdong Natural Science Foundation (Grant No. 2018A030310129), the Education Department of Guangdong (Grant No. 2020KZDZX1184), the National Science Foundation of China (NSFC No. 61802163) and PCL Future Regional Network Facilities for Large-scale Experiments and Applications (PCL2018KP001).

6. REFERENCES

[1] Bus boarding analysis in adelaide. Kaggle competition. https://www.kaggle.com/rednivrug/comprehensive-bus-boarding-analysis.

[2] H. Doraiswamy, E. Tzirita Zacharatou, F. Miranda, M. Lage, A. Ailamaki, C. T. Silva, and J. Freire. Interactive visual exploration of spatio-temporal urban data sets using urbane. In *SIGMOD*, pages 1693–1696, 2018.

[3] Y. Li, M. L. Yiu, and Z. Gong. Quick-motif: An efficient and scalable framework for exact motif discovery. In *ICDE*, pages 579–590, 2015.

[4] K. Nguyen, J. Yang, Y. Lin, J. Lin, Y.-Y. Chiang, and C. Shahabi. Los angeles metro bus data analysis using gps trajectory and schedule data (demo paper). In *SIGSPATIAL*, pages 560–563, 2018.

[5] B. Tang, C. Yang, L. Xiang, and J. Zeng. Deriving real-time city crowd flows by heterogeneous big urban data. In *IEEE Big Data*, pages 3485–3494, 2018.

[6] B. Tang, M. L. Yiu, Y. Li, and L. H. U. Exploit every cycle: Vectorized time series algorithms on modern commodity cpus. In *Data Management on New Hardware*, pages 18–39. 2016.

[7] B. Tang, M. L. Yiu, K. Mouratidis, and K. Wang. Efficient motif discovery in spatial trajectories using discrete fréchet distance. In *EDBT*, 2017.

[8] C. Yang, Y. Zhang, B. Tang, and M. Zhu. Vaite: A visualization-assisted interactive big urban trajectory data exploration system. In *ICDE*, pages 2036–2039, 2019.

[9] J. Yu, Z. Zhang, and M. Sarwat. Spatial data management in apache spark: The geospark perspective and beyond. *Geoinformatica*, 23(1):37–78, 2019.