TrajectoryCurer: Visual Analysis of Trajectory Data Quality

Xiaodan Miao[*]
School of Software Technology, Zhejiang University

Sitong Pan[†]
Zhejiang University

Shifu Chen[‡]
School of Software Technology, Zhejiang University

Di Weng[§]
School of Software Technology, Zhejiang University

Yingcai Wu[¶]
State Key Lab of CAD&CG, Zhejiang University

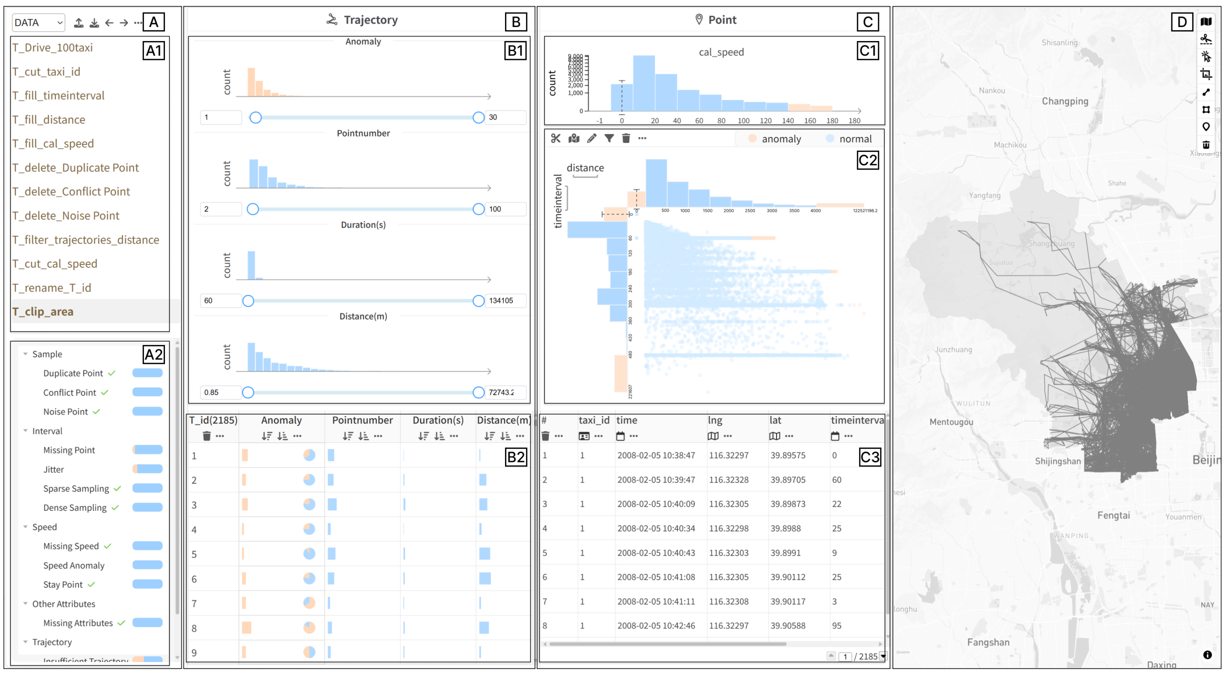

Figure 1: The TrajectoryCurer interface displaying the dataset after area clipping in Usage Scenario 2. (A) Data View records all transformation operations and lists detected quality issues organized by the six-category taxonomy. (B) Trajectory View presents trajectory-level feature distributions and enables wrangling operations such as filtering and segmentation. (C) Point View illustrates point-level characteristics including speed and spatiotemporal intervals, with raw data in tabular format. (D) Map View renders trajectories in their geographical context and supports spatial operations such as clipping and point editing.

Abstract

Trajectory data record spatiotemporal information and related attributes of moving objects. High-quality trajectory data can accurately reflect behavior patterns, providing a reliable foundation for traffic management, logistics optimization, and smart city planning. However, existing research on trajectory data quality management remains limited, lacking interactive tools for addressing quality issues and making it difficult for users to explore and resolve problems in trajectory datasets. To address this gap, we present TrajectoryCurer, an interactive visual analytics system designed for trajectory data quality management. Through expert interviews and synthesis of preprocessing operations from 30 relevant articles on trajectory visualization and management, we construct a taxonomy of 19 trajectory data quality issues across six dimensions. Based on this taxonomy, we design a multilevel visualization approach encompassing Trajectory View, Point View, Map View, and Data View, enabling users to observe and address quality issues across multiple dimensions through an integrated interface. We demonstrate the effectiveness and usability of TrajectoryCurer through two usage scenarios and expert evaluations.

[*]e-mail: xiaodanmiao@zju.edu.cn
[†]e-mail: pstongboa@gmail.com
[‡]e-mail: shifu.chen@outlook.com
[§]e-mail: dweng@zju.edu.cn. Di Weng is the corresponding author.
[¶]e-mail: ycwu@zju.edu.cn

1 Introduction

As global trajectory data continues to grow exponentially, it has become increasingly crucial in various domains, including urban computing [15], human behavior analysis [38], and environmental assessment [14]. 2D georeferenced trajectory data, such as GPS traces from vehicles [53], vessels [37], and mobile devices [54], constitutes the predominant form of trajectory data in urban computing and transportation applications. However, quality issues in trajectory data are widespread [60], compromising data usability and potentially leading to significant losses in downstream applications and related fields. For instance, inaccuracies in location information, such as GPS drift, can lead to misjudgments about congested areas, potentially affecting critical decisions such as emergency vehicle routing.

Currently, trajectory data quality management primarily relies on scripting and GIS tools, which lack interactive features and require significant manual efforts. While interactive data quality management for tabular data has made significant progress [39], with tools like Profiler [27] and CoWrangler [13] demonstrating capabilities in handling conventional data quality issues such as missing values and duplicates, these solutions are insufficient for trajectory data. Trajectory data presents more complex quality challenges due to its inherent spatiotemporal sequence integrity and continuity. Beyond typical data quality issues, trajectory data faces unique challenges including timestamp jumps, GPS drift, and semantic inconsistencies. Common quality management operations such as trajectory segmentation, filtering, and attribute computation require specialized approaches that existing tools cannot adequately address.

In this paper, we propose TrajectoryCurer, an interactive visualization system for managing trajectory data quality. Through our initial design attempts, we identified two major challenges. First, the spatiotemporal characteristics of trajectory data are complex and diverse. Data records not only include timestamps and spatial coordinates but also encompass various trajectory attributes, such as speed and direction, as well as semantic information about the trajectories, such as activity types and traffic patterns. The implicit relationships and dependencies among these data elements complicate the systematic organization and classification of trajectory data quality issues. Second, although methods exist for handling trajectory data, leveraging trajectory data analysis to support diverse and effective data processing operations remains challenging. Ensuring data integrity and accuracy while providing a user-friendly interactive experience requires further exploration.

To address these challenges, we first systematically review various preprocessing operations for trajectory datasets from research on trajectory visualization and management, and then construct a taxonomy of 19 trajectory data quality issues. Based on this taxonomy, we develop an interactive visualization system that supports automated quality issue monitoring, multidimensional trajectory visualization, and data wrangling tasks. To evaluate its effectiveness and usability, we conduct two usage scenarios. The results indicate that TrajectoryCurer effectively facilitates trajectory data quality management. In summary, our main contributions include:

- A taxonomy of 19 trajectory data quality issues across 6 dimensions, synthesized from 30 research papers and expert interviews.
- A diagnostic framework that formally maps quality issue types to visualization components (Tab. 1), enabling systematic detection at both trajectory and point levels.
- TrajectoryCurer, an interactive system integrating quality diagnosis with wrangling operations, bridging the gap between trajectory visualization tools (which assume clean data) and general wrangling tools (which lack spatiotemporal semantics).

2 Related Work

Research on combining visualization technology with trajectory data preprocessing mainly involves three directions: 1) trajectory data visualization, 2) data profiling for trajectory data issue detection, and 3) trajectory data transformation. In the following, we introduce related work from these aspects.

2.1 Trajectory Data Visualization

Trajectory data visualization techniques have become essential for analyzing and understanding complex dynamic trajectories such as traffic flows and collective movements. Andrienko et al. [5] establish the theoretical framework for exploratory spatiotemporal analysis, underscoring the critical role of extracting behavioral patterns and flow dynamics from complex movement data. Existing visualization techniques can be broadly divided into two categories: map-based visualization and attribute visualization.

Map-based visualization is fundamental for rendering trajectories and presenting their spatiotemporal information. The Space-Time Cube (STC) [29] visualizes three-dimensional spatiotemporal paths by introducing a vertical time axis. Hurter et al. [23] introduce edge bundling techniques to resolve visual clutter caused by overlapping trajectories. To support deeper analysis, TrajGraph [20] transforms taxi trajectories on city streets into graph structures to reveal traffic patterns through graph analysis algorithms. SemanticTraj [4] enhances interactivity by associating trajectories with semantic tags, enabling natural language queries and helping users understand urban mobility patterns through both spatial and semantic perspectives. Overall, map-based visualization provides an intuitive interface for trajectory data management, allowing users to directly observe and manipulate trajectory data in a spatial context.

Attribute visualization focuses on representing temporal and multivariate attributes of trajectories. Scheepens et al. [47] propose composite density maps to extract multivariate features from trajectory aggregations, effectively simplifying complex movement data into density fields. 3D visualization techniques such as ribbons and tubes are frequently employed to represent multidimensional trajectory attributes. For instance, stacking-based visualization [51] utilizes 3D ribbons with varying widths and colors to depict spatiotemporal evolution alongside attribute characteristics, while OnTube [46] integrates multivariate visual channels onto tube surfaces to display geographic paths together with attributes such as speed and acceleration. Additionally, TrajectoryLenses [31] reduces data complexity and improves analysis efficiency by allowing users to quickly focus on specific regions or time periods within large-scale data through interactive filtering and focus+context techniques. Dynamic and real-time feedback plays an important role in trajectory data processing visualization. For example, incremental visualization [28] enables users to view the step-by-step effects of each data transformation in real time, helping them understand how data transforms from its original state to the final result.

While existing techniques have made significant progress in trajectory visualization, they provide limited support for exploring and analyzing data quality issues. Therefore, effective trajectory data quality management requires both map-based and attribute visualization working in conjunction, enabling users to identify and resolve quality issues from multiple perspectives.

2.2 Data Profiling for Trajectory Data Issue Detection

In data management, data profiling has become indispensable, especially when dealing with massive datasets, as it provides systematic support for data quality assessment [1, 21]. Data profiling refers to the in-depth analysis of a dataset to extract statistical features, patterns, and potential problems in preparation for subsequent data cleansing, transformation, and analysis [1, 27]. However, although data profiling has been widely applied to tabular data, these methods cannot be directly applied to trajectory data due to the high dimensionality and spatiotemporal nature of trajectory data (e.g., varying sampling frequency and positioning accuracy). Specialized profiling methods for trajectory data remain scarce.

Most existing trajectory data analysis methods focus on data mining and pattern recognition [3, 9, 10, 50]. For example, research in trajectory data mining [58] addresses tasks such as path discovery, anomaly detection, and trajectory clustering [17]. However, these approaches typically assume that trajectory data have been cleaned and transformed, and therefore lack profiling analysis for raw trajectory data. Similarly, although some tools employ data profiling for data quality management (e.g., Profiler [27], CoWrangler [13]), they primarily target tabular data. Tools specifically designed for automated profiling and quality detection of trajectory data remain in early stages of development.

For quality assessment of trajectory data, existing studies mainly focus on data cleaning and error detection, such as rule-based dirty data taxonomy and geographic data quality assessment [6, 34]. More recently, T-Assess [60] proposed an automated trajectory data quality assessment system with criteria spanning validity, completeness, consistency, and fairness. Although these studies have contributed to detecting and quantifying data errors, they primarily focus on automated assessment without providing interactive visualization for exploration or integrated wrangling capabilities. Our work aims to present a system that combines comprehensive trajectory data profiling with interactive visual analytics, enabling users to not only detect data issues at both trajectory and point levels but also directly perform cleaning and transformation operations within a unified interface.

2.3 Trajectory Data Transformation

Data wrangling is the process of transforming raw data into a structured, clean, and analysis-ready format [13]. This task is particularly complex for trajectory data because raw trajectory data often suffer from quality issues such as missing values, noise, errors, and duplicates. The first step in trajectory data processing is usually data cleaning, where algorithms identify and correct outliers, gaps, or duplicate records [34]. Beyond cleaning, data wrangling includes transformation and filtering to meet the requirements of downstream analysis or visualization [19].

Trajectory data require more complex preprocessing steps than tabular data due to their inherent spatiotemporal properties. Preprocessing operations can be categorized by attributes, including the number of data points [3,30], the velocity [3], the overall velocity of the trajectory [3, 16], distance [16, 30, 48], time [4, 16, 30, 36, 41, 48, 51], region [3, 16, 20, 31, 36, 41, 48, 51], and direction [2, 41]. Various preprocessing methods exist for these attributes, including conditional filtering [3, 16, 36], enrichment [30, 51], compression [3, 31], aggregation [2, 22] and segmentation [30, 59].

Modern data wrangling tools have introduced various automated and semi-automated data transformation techniques. For example, CoWrangler [13] is a real-time data wrangling recommendation system that automatically recommends the next best operation and corresponding code snippet based on data quality issues and analysis needs. Foofah [25] helps analysts perform data cleaning and transformation by synthesizing transformation programs from user-provided examples. These tools greatly reduce the burden of manual operations and minimize errors caused by human factors. Similarly, ArcKarma [12] automatically generates conversion programs from user-supplied examples to clean and transform geodata. However, for trajectory data, existing methods still require writing specialized code or performing numerous cumbersome operations, while lacking visualization support to assist wrangling tasks.

3 Formative Study

We conducted a formative study to systematically identify quality issues in trajectory data and understand how they are addressed through preprocessing. Our study focuses on 2D georeferenced trajectory data that records spatial coordinates and timestamps of moving objects on or near the Earth's surface. This scope encompasses common trajectory sources such as GPS-equipped vehicles, mobile phones, and location-based services, while excluding 3D trajectories involving altitude variations (e.g., aircraft, drones) that require fundamentally different visualization and quality assessment approaches. Through a literature review and case analysis, we developed a taxonomy of data quality problems.

3.1 Identifying Quality Issues in Trajectory Data

Trajectory data contain complex spatiotemporal attributes, making raw trajectory data often unusable directly due to many quality issues, such as inaccuracies in acquisition equipment, data loss, and non-uniform data formats. Therefore, before trajectory data can be used in analysis and visualization, these quality issues need to be addressed. In addition to addressing quality issues, users also perform various data operations (e.g., filtering, transformation) based on their specific needs. We surveyed $N = 30$ articles that deal closely with trajectory data to systematically identify trajectory data quality issues and categorize them, as well as to identify user-driven data preprocessing steps.

To understand the key challenges in trajectory data quality and preprocessing needs, we conducted a comprehensive literature review of 30 research papers focused on trajectory data visualization, processing, and analysis. These papers span a range of applications, including urban mobility analysis, traffic monitoring, and human behavior modeling. Our review aimed to extract the types of quality issues commonly encountered in trajectory datasets.

Our analysis followed a dual-path approach to 1) identify inherent data quality issues and 2) extract user-initiated preprocessing operations. We collected literature from 2013 to 2025 using databases including IEEE Xplore, ACM Digital Library, and DBLP. This collection encompasses prominent journals and conferences in Visualization, Geographic Information Science, and Urban Science such as TVCG, IEEE VIS, and IJGIS. The search utilized combinations of keywords including *trajectory*, *data*, *cleaning*, *transformation*, *quality*, *visual analytics*, *wrangling*, *preprocessing*, and *processing*. One researcher first identified and coded mentions of data quality issues (e.g., missing values, GPS drift, outliers) related to trajectory data in each article to form preliminary results. Simultaneously, the researcher identified common user-driven operations. Another researcher then reviewed and checked the preliminary results. The two researchers discussed and supplemented the results collaboratively when encountering disagreements or omissions. Subsequently, the first researcher further summarized and categorized the quality issues and operations obtained from each article through semantic analysis [26]. The second researcher reviewed the categorization results and discussed any contradictions with the first researcher. With these initial results, we also conducted interviews with two experts who have over 5 years of experience analyzing and visualizing trajectory data. We discussed with them the handling of trajectory data quality issues in their work and common processing operations. These interview results were integrated into our findings. The formative analysis informed the taxonomy of trajectory data quality issues and the summary of user-driven operations.

3.2 Survey Findings

Based on the literature survey and expert interviews, this section presents two findings: a taxonomy of trajectory data quality issues and a summary of user-driven preprocessing operations.

3.2.1 Taxonomy of Trajectory Data Quality Issues

Our analysis revealed that trajectory data suffer from a wide variety of quality problems due to noisy sensors, signal loss, irregular sampling, and human or device errors. We extracted 19 specific types of quality issues from the literature and classified these issues into six major categories, inspired by general data quality dimensions [55] but extended to address spatiotemporal characteristics.

Each identified issue was mapped to one or more standard data quality dimensions, resulting in a structured taxonomy (see Tab. 1). We adopt six widely acknowledged dimensions from the data quality literature, including *completeness*, *uniqueness*, *effectiveness*, *consistency*, *conciseness*, and *accuracy*, and adapt them to the context of spatiotemporal trajectory data. Below, we describe each dimension with representative examples.

Completeness refers to the presence of required information in trajectory records, e.g., whether trajectory points and attributes are complete and free of defects [35]. If the percentage of missing points is high or the missing parts are atypical, analysis results may be biased or misleading. Completeness problems in trajectory data usually originate from unstable GPS signals or device failures that result in missing position records, as well as records that start too late or end too early so that part of the trip is not captured. Common completeness issues include missing spatiotemporal points, missing speed attributes, and large time intervals between consecutive points, which indicate sparse sampling or data loss. For instance, some trajectories lack the necessary spatial or temporal resolution to accurately reflect movement paths [11].

Uniqueness indicates whether each data entry appears only once. In trajectory data, duplication may appear as repeated spatiotemporal points (duplicate samples), repeated timestamps, or duplicate trajectories from system logging errors or merging artifacts from multiple sources. Duplicate data not only increases storage and processing overhead but also biases analysis results. For example, in GPS logs, if a record of the same vehicle at the same time and location is stored twice, the dataset contains duplicate points. In traffic flow analysis, this can lead to double counting of vehicles or distance traveled, thus overestimating traffic volume [24].

Effectiveness focuses on the meaningfulness and validity of data values, such as whether they fall within reasonable limits and meet required patterns [60]. Common effectiveness issues include geographic coordinates outside valid ranges (e.g., latitude exceeding $\pm 90^{\circ}$), incorrectly formatted or disordered timestamps, and motion parameters that defy physical laws (e.g., negative velocity or values exceeding reasonable limits). These problems reflect acquisition errors or transmission noise in trajectory data, which significantly hinder efficient analysis or visualization [33].

Consistency emphasizes the internal coherence of data formats and semantics. High consistency means that all trajectory records follow a uniform specification and reference system and do not contain contradictory information. Typical consistency issues include different parts of the data using inconsistent coordinate systems or units (e.g., one part using latitude and longitude, another using planar coordinates without conversion), time fields using different time zones or formats, and inconsistent sampling frequencies across different devices [45]. These inconsistencies may lead to difficulties in data alignment and misinterpretation in downstream analysis.

Conciseness means that trajectory data are compact and free of redundancy, expressing complete motion information with the minimum amount of data [43]. A common conciseness problem in trajectory data is the presence of numerous redundant or over-sampled points in dense intervals, which often result in consecutive identical or near-identical points, inflating data volume without contributing new information. For example, moving objects may record positions at high frequency even when stationary, generating many duplicate stationary points, or may record excessive points during nearly linear motion. Other examples include excessive redundant attribute logging.

Accuracy captures the degree to which data values (e.g., position and time) reflect real-world values [57]. For trajectory data, this often manifests as positioning measurement errors and random noise, which cause sampling points to deviate from the true path. In more extreme cases, obvious distortion anomalies occur, which can be caused by brief interruptions in satellite signals or equipment malfunctions. Furthermore, inaccurate speed computations or mismatches during map matching are also common accuracy problems that affect trajectory alignment with road networks.

This classification enables us to understand not only the nature of quality issues but also how they impact downstream analysis. The taxonomy serves as the foundation for designing targeted profiling and wrangling mechanisms in our system.

3.2.2 User-Driven Data Preprocessing

In addition to addressing data quality issues, trajectory data preprocessing often involves operations driven by users' specific analytical goals. These user-driven operations are not necessarily triggered by incorrect or faulty data, but rather by task-oriented requirements that precede effective visualization or modeling. Through our literature review and scenario analysis, we identified several typical types of user-driven preprocessing operations.

Object-level filtering allows analysts to select trajectories of particular entities (e.g., a single taxi or all vehicles in a fleet). This supports fine-grained tracking or comparisons across categories.

Spatial filtering or **clipping** limits the dataset to a region of interest (e.g., only trajectories within a city center or near a specific landmark). Analysts may want to focus on areas with high congestion, critical infrastructure, or environmental sensitivity.

Temporal filtering enables analysts to restrict the dataset to a specific time range (e.g., selecting only daytime trajectories or excluding weekends). Such operations allow users to focus on relevant time windows for their analysis, such as rush hours in traffic studies or specific event periods in crowd movement analysis.

Trajectory segmentation or aggregation manipulates trajectories based on semantic or temporal criteria. For example, users may want to segment long trajectories into sub-trips based on dwell time thresholds, or aggregate multiple trips by the same vehicle for statistical analysis.

These user-driven operations highlight that data wrangling is not limited to quality repair–it also supports exploratory focus. Therefore, our system is designed to provide both automatic quality issue detection and interactive operations to meet these flexible, user-defined goals, as further discussed in the system design requirements in Section 4.

4 Requirements

To inform the system design, we conducted interviews with three experts who have extensive experience in trajectory data analysis and wrangling. We held weekly meetings with them to share our initial ideas and design artifacts generated through the iterative design process over a nine-month period. Through these discussions, we established the scope of our system: focusing on data cleaning and transformation operations (e.g., removal, segmentation, filtering, imputation) that address the quality issues identified in our taxonomy (Section 3), rather than downstream analytical tasks such as clustering or advanced feature extraction that typically consume already-cleaned data. Additionally, we held discussions with two experts in geographic information science to gather their insights and feedback. During these sessions, we presented our concepts and design prototypes, engaging in in-depth discussions to understand their needs and collect suggestions for improvement. Through multiple rounds of discussions, we iteratively extracted and summarized the following five design requirements.

R1: Trajectory-Level Quality Visualization. The system should provide efficient identification and visualization of quality issues at the trajectory level, such as excessively short trajectories and sparsely distributed samples. Specifically, it should: (1) present an overview of trajectory-level feature distributions (e.g., trajectory length, point count, duration) to help users quickly locate problematic trajectories; and (2) support detailed visualization of individual trajectory attributes to facilitate in-depth inspection. This addresses the current inefficiency of manual checking or script-based examination of trajectory features.

R2: Point-Level Quality Visualization. The system should support visualization and detection of quality issues at the point level. Specifically, it should: (1) provide an interface for viewing point attributes, including location, timestamp, speed, and spatiotemporal intervals; (2) automatically detect and flag potential anomalies such as outliers, duplicate points, and missing values;

Table 1: Our taxonomy categorizes 19 types of trajectory quality issues by data objects and quality dimensions, where empty intersections indicate no observed issues. The specific System Views supporting each issue are marked in brackets (DV: Data View, TV: Trajectory View, PV: Point View, MV: Map View).

	Completeness	Uniqueness	Effectiveness	Consistency	Conciseness	Accuracy
Sample	Missing latitude, longitude, or timestamps **[DV, PV]**	Duplicate locations or timestamps **[DV, PV, MV]**		Inconsistent coordinate systems or timestamp formats **[DV, PV]**		Deviation from other locations **[DV, PV, MV]**
Interval	Missing samples (excessively-long intervals) **[DV, PV, MV]**		Incorrect sample order Sparsely-distributed samples **[DV, PV, MV]**	Jittering (excessively-short intervals) **[DV, PV, MV]**	Densely-distributed samples **[DV, PV, MV]**	
Speed	Missing speed **[DV, PV]**		Anomalous speed **[DV, PV, MV]**	Inconsistent speed units **[DV, PV]**		Inaccurate speed calculation **[DV, PV]**
Other attributes	Missing attributes **[DV, PV]**			Inconsistent attribute types **[DV, PV]**		Inaccurate attributes **[DV, PV]**
Trajectory		Duplicate trajectory **[DV, TV, PV, MV]**	Excessively-short trajectories **[DV, TV, PV, MV]**			
Map-matching	Missing map-matching **[DV, MV]**					Inaccurate map-matching **[DV, MV]**

and (3) visually distinguish between normal and anomalous points to assist in rapid identification. This requirement emerged from experts' experiences with tedious manual inspection of individual data points.

R3: Geographical Feature Visualization. The system should display trajectory data through a map interface to support analysis of geographical characteristics. Specifically, it should: (1) render trajectories in their spatial context; (2) support interactive operations, including zooming, panning, and trajectory selection; and (3) provide real-time visual feedback when data are modified, enabling users to assess processing outcomes in their geographical context.

R4: Data Cleaning and Transformation Operations. The system should support diverse wrangling operations for trajectory data, including: (1) corrective operations for quality issues, such as removing noise points and segmenting trajectories at anomalous points; (2) transformation operations for analytical needs, such as spatial clipping, temporal filtering, and attribute-based filtering; and (3) batch operations for common tasks, such as deduplication and missing value imputation.

R5: Provenance Support. The system should support iterative data quality management by: (1) preserving wrangling operation history; (2) allowing users to revert to previous data states; and (3) triggering quality reassessment after each operation. This ensures transparency and traceability throughout the cleaning process.

5 TRAJECTORYCURER

TrajectoryCurer comprises four views: Data View provides a comprehensive display of the dataset list after each operation, along with an overview and detailed information regarding data quality issues (R5). **Trajectory View** focuses on trajectory-level features of the dataset (R1), including anomaly ratios, point counts, total duration, total distance, and dynamic changes in trajectories. **Point View** showcases the characteristics of point-level features (R2): speed, time interval, and distance interval, and presents the raw data in tabular format. **Map View** illustrates the geographical distribution of trajectories (R3), enabling users to intuitively observe and analyze spatial information on the map. Each view supports data processing operations on trajectory data (R4), enabling users to efficiently explore and optimize data from various perspectives. The interface design is directly guided by the taxonomy. Specifically, we provide a formal mapping between the 19 types of quality issues and the specific visualization components used for their detection (see Tab. 1). The system utilizes Python for data detection and processing, and employs Vue3 and TypeScript integrating Mapbox GL JS [42], D3.js [7], and ECharts [32] for geospatial rendering and interactive visualization.

5.1 Data View

Data View (Figure 1A) primarily consists of two components to facilitate understanding of trajectory datasets.

Dataset List Panel (Figure 1A1). The panel presents a list of datasets generated after each operation, naming them based on the operation's title and relevant details. This facilitates documentation of the content and outcomes of each operation, enabling users to effectively track the history of data processing. Users can click on dataset names to swiftly revert to the data state of that step (R5).

Issue Panel (Figure 1A2). The panel employs a tree structure to systematically present detailed information regarding quality issues (R5). The primary categories correspond to six data objects within the classification system, including *sample* (individual data points' timestamps and geographical coordinates), *interval* (the gaps and continuity between successive points), *other attributes* (additional data attributes, such as direction and semantics), *speed*, *trajectory*, and *map-matching*. Under each primary category, specific issues are enumerated and visualized through bar charts that depict their frequency and proportion, helping users intuitively understand the severity of these issues. Users can access relevant information associated with an identified issue by clicking on it (Figure 6). Upon resolution, issues will be marked as resolved.

Furthermore, this panel enables users to customize detection rules, such as adjusting the system's default threshold for anomalous speed from 140 km/h to 120 km/h, to accommodate the requirements of various application scenarios. This design not only aids users in understanding the quality issues within the dataset but also significantly enhances the flexibility and adaptability of data management.

5.2 Trajectory View

Trajectory View (Figure 1B) focuses on trajectory-level features, including the number of anomalous points, the proportion of anomalous points, the total number of points, total duration, and total distance (R1). It comprises two components.

Trajectory Feature Visualization Panel (Figure 1B1). The panel summarizes the distribution of features, helping users understand key trends and potential anomalies. Through charts, users can rapidly identify trajectories with a high frequency of anomalies or those exhibiting abnormal lengths (either excessively short or long), providing critical guidance for subsequent data cleaning and analysis. Additionally, users can filter trajectories through slider adjustments or by inputting specific ranges, enhancing the flexibility and

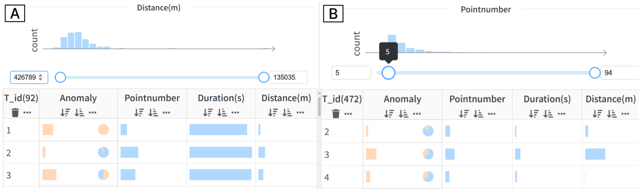

Figure 2: Filtering features in the trajectory feature visualization panel. (A) The distance threshold is given to filter trajectories that meet the specified range. (B) The slider is dragged to filter trajectories with at least 5 points.

specificity of the analysis (Figure 2) (R4).

Trajectory LineUp Panel (Figure 1B2). Inspired by LineUp [18], the panel enables in-depth analysis and detailed management of individual trajectories. Its sorting functionality allows users to flexibly order trajectories based on key characteristics, such as the number of anomalies, the total number of points, total duration, and total distance. This sorting mechanism not only aids users in swiftly identifying trajectories with significant features but also establishes a clear prioritization for subsequent analyses. Icons below the panel title offer various operational options for trajectories, including deletion, segmentation, and filtering (R4). Each row visualizes the specific characteristics of individual trajectories using bar charts, while pie charts illustrate the proportion of anomalies within each trajectory, enabling users to quickly assess individual trajectory performance and focus on those of research interest or requiring further attention.

This design enables rapid identification of quality issues across the trajectory, interval, and sample levels. For example, when a high proportion of anomalous points is detected in the pie chart for the trajectory with *T_id* 4, the user can click on this trajectory for further analysis. In the temporal-spatial distribution panel, by analyzing features such as speed, time interval, and distance interval for the points in the trajectory, the user can determine that the anomalies primarily stem from excessively large sampling time intervals (Figure 3B). Specifically, the temporal-spatial distribution panel visually indicates that the sampling time intervals are concentrated around 600 seconds, as corroborated by the "*timeinterval*" column in the raw data table (Figure 3D). Furthermore, the sampling distance intervals do not show a significant increase, which is also reflected in the trajectory path rendered in Map View (Figure 3C). In handling complex trajectory datasets, the LineUp panel demonstrates its unique advantages through careful examination of individual trajectories.

5.3 Point View

Point View (Figure 1C) focuses on point-level features, including point speed, time intervals, distance intervals, and the raw data for each point (R2).

Speed Distribution Panel (Figure 1C1). The top of the panel presents a histogram showing the speed distribution of all points in the dataset. Points with a speed of -1 are marked as conflict points, indicating that multiple locations correspond to the same timestamp, which violates the principle of spatiotemporal consistency and is classified as anomalous data. Points with a speed of 0 may represent stationary points, but this requires further verification using information from adjacent points; a single point with a speed of 0 cannot be definitively identified as a stationary point. Based on actual needs and regulatory requirements, TrajectoryCurer defaults to classifying speeds exceeding 140 km/h as anomalous points to balance data quality with the complexity of driving behavior.

Temporal-Spatial Distribution Panel (Figure 4). The middle section integrates a scatter plot with two histograms. The coordinate system of the scatter plot is based on two key parameters from the trajectory data: time interval and distance interval. To mitigate potential visualization issues arising from extensive data ranges, both *X* and *Y* axes feature a dual-scale design. Each point in the scatter plot is positioned according to its corresponding time and distance interval values, thereby providing a visual representation of the distribution of trajectory data in both temporal and spatial dimensions. Based on a predefined speed threshold, points are categorized into "normal speed" and "abnormal speed," with distinct color coding for each category. Histograms are embedded along the X and Y axes, with the histogram on the X axis reflecting the distribution of distance intervals of trajectory points, while the histogram on the Y axis displays the distribution of time intervals. This design enables users to rapidly understand the distance and time interval distributions of trajectory data, thereby inferring the sampling frequency of the trajectory data. To intuitively present the detection results, histogram bars corresponding to anomalous intervals are highlighted in orange.

Figure 3: The trajectory LineUp panel interacts with other panels to examine a single trajectory. (A) The trajectory with *T_id* 4 is highlighted in the panel. (B) The scatter plot displays features such as speed, time interval, and distance interval for the points in the trajectory with *T_id* 4. (C) The map view visualizes the trajectory, with orange and blue segments indicating interval anomalies and normal data, respectively. (D) The raw data table shows the values of relevant features for the trajectory with *taxi_id* 4.

Anomalous Interval Detection. The detection is based on an automated approach. The system flags points with intervals less than or equal to zero as anomalies. A distance interval of zero may indicate a stationary point in the trajectory, while a time interval of zero represents duplicate or conflict points. These specific types of points are separately quantified, allowing users to quickly focus on them. For data with intervals greater than zero, the system employs an Interquartile Range (IQR) method based on optimal knee-point detection to identify outliers. In the IQR method [52], the detection sensitivity is controlled by a scaling coefficient *k*, which adjusts the threshold range relative to the data's quartile distribution. To determine the optimal *k*, we set a reasonable search range. The minimum value is set to 0.75, based on half of the empirical value of 1.5, to ensure the algorithm begins with a strict standard. The maximum value is set to 5 or 10, which is sufficient to cover extreme cases that may appear in the dataset. Within this predefined range, the algorithm iterates through different *k* values, calculating and recording the corresponding number of outliers. This process generates a curve where the number of outliers decreases rapidly as *k* gradually increases, then slows abruptly, and finally becomes nearly flat. The system employs a curvature-based analysis method to identify

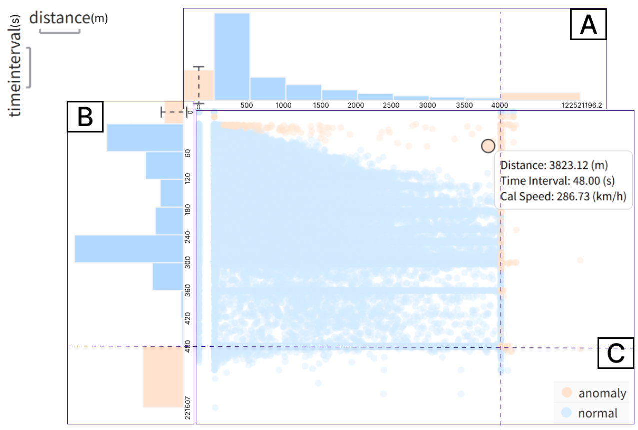

Figure 4: The temporal-spatial distribution panel visualizes the spatiotemporal distribution of points. The top histogram displays the distribution of distance intervals for all points (A), while the left histogram shows the distribution of time intervals (B). The central scatter plot maps each point based on time and distance intervals, using color coding to distinguish points with abnormal speeds (C). Both the X and Y axes feature dual scales.

this transition point from rapid decrease to a flat slope, and the corresponding k is determined as the optimal threshold. Finally, the system uses this optimal k to calculate the upper and lower outlier bounds, and any data point falling outside this range is flagged as an anomaly.

Raw Data Table (Figure 1C3). The bottom section of the raw data table serves as one of the core components for data presentation and interactive operations (R3). Its design is inspired by Tableau [49]. The table employs a paginated layout, with each trajectory corresponding to a separate page, and each row of data representing a point within the trajectory, encompassing key attributes. Users can select data types via icons located beneath the header bar, enabling operations such as deletion, filtering, trajectory segmentation, and unit conversion. Furthermore, the system supports basic table editing functionalities, including modification of data values and addition or deletion of columns (R4).

Interactions (Figure 5). Users can select individual points in the scatter plot or click on bars in the histograms to highlight a category of points. Additionally, users can click on specific trajectories in the trajectory LineUp panel or Map View, prompting TrajectoryCurer to display the distribution of those trajectory points in the scatter panel. These operations not only facilitate real-time updates of the scatter plot and histogram but also synchronize with the raw data table and geospatial trajectory rendering. Through this multi-view linkage approach, users can gain a deeper understanding of the spatiotemporal characteristics of trajectory data.

5.4 Map View

Map View (Figure 1D) visualizes 2D georeferenced trajectory data, supporting real-time updates and allowing users to interactively explore specific information by zooming, panning, and clicking on trajectories and points displayed on the map (R3). To facilitate quality diagnosis, anomalous points and segments representing *interval* issues are highlighted in orange upon inspection (Figure 6C, Figure 3C). This explicit visual indication enables users to perform targeted repairs on anomalies directly within the geospatial context. The functionalities of Map View include the ability to switch the map background, insert and delete points, segment trajectories, remove drawn content, and crop trajectories within a selected area, among others (R4). These features support efficient analysis and enhance user interaction.

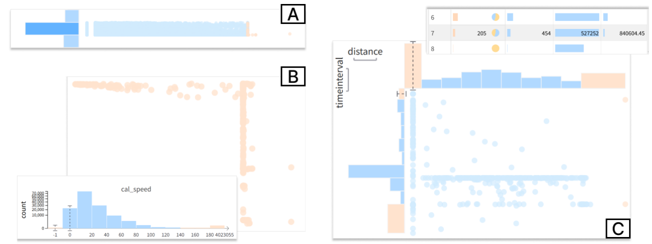

Figure 5: The interactions of the Temporal-Spatial Distribution Panel include viewing points that appear most frequently within the range of sampling intervals in the dataset (A). Additionally, users can examine points with speeds exceeding 180 km/h (B) and select specific trajectories to display the feature distribution of their points (C).

6 Usage Scenarios

We present two usage scenarios to illustrate the functionality, flexibility, and usability of TrajectoryCurer. These scenarios demonstrate the performance of TrajectoryCurer in practical applications.

6.1 Data Cleaning for Quality Issue Resolution

Emily, a geospatial data analyst, selects a subset of trajectory data from the T-DRIVE dataset [56], which comprises GPS trajectories of taxis in Beijing from February 2 to February 8, 2008, as the dataset for her research analysis. Although Emily possesses some programming expertise, she aims to optimize the selected data in a more exploratory manner while minimizing the amount of code required, thereby ensuring high data quality. Consequently, she opts to utilize TrajectoryCurer for this optimization process.

Emily uploads the original dataset to TrajectoryCurer, which quickly displays the dataset name in the dataset list panel and conducts a quality assessment. Emily observes that only a single trajectory is displayed as $T\_id1\ldots100$ in the trajectory LineUp panel. Upon examining the raw data table, she discovers that the dataset is stored in a single file, causing TrajectoryCurer to interpret it as one complete trajectory. Consequently, Emily chooses to segment the trajectories by *taxi_id*. The dataset list panel updates, generating a new dataset named *T_cut_taxi_id*. The LineUp panel displays 100 trajectories, and the raw data table is divided into 100 pages.

Emily's attention remains focused on the raw data table, where she discovers that the columns for *timeinterval*, *distance*, and *cal_speed* (the speed calculated from the distance and time interval) are empty, indicating that the dataset only has *taxi_id*, *time*, *lat*, and *lng* attributes. She performs a one-click operation, and TrajectoryCurer promptly fills in the missing values (R4). The issue panel indicates that the missing speed and attribute values are successfully addressed, thereby enhancing the completeness of the dataset.

Subsequently, Emily notices the duplicate points in the issue panel and clicks on that issue, causing the duplicate points to be highlighted in both Point View and Map View. After reviewing and confirming, she clicks the delete button in the raw data table. TrajectoryCurer automatically retains the first point among the duplicates. Emily also applies the similar procedures to conflict points and noise points (Figure 6).

Next, Emily observes that the issue panel indicates the presence of excessively short trajectories. She examines the trajectory feature visualization panel and discovers that the minimum number of points is two and the minimum total distance is zero. Looking at the trajectory LineUp panel, she notes that the bar for the trajectory with *T_id* 8 exhibits a significantly lower height and discovers that it contains only 5 points with a total distance of just 13.77 meters.

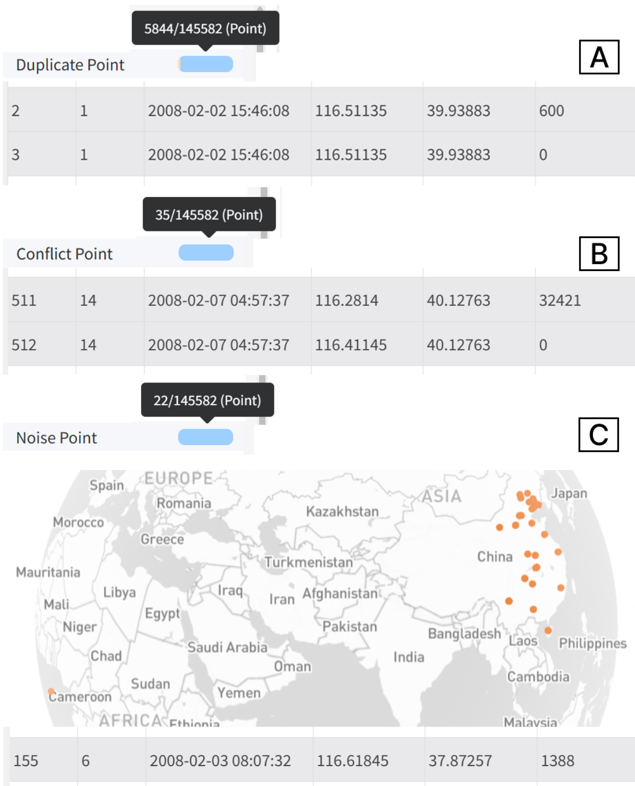

Figure 6: The detected quality issues in the samples include duplicate points with the same timestamp and coordinates (A), conflicting points with the same timestamp but different coordinates (B), and noise points with coordinates outside the Beijing area (C).

Considering that this is a week-long trajectory dataset, and after analyzing the total distance histogram, Emily finds that most trajectories have a total distance exceeding 400 kilometers. Therefore, she determines that the minimum total distance threshold should be set to greater than 400 kilometers. She then filters the trajectories by inputting this value (Figure 2A).

Emily also notes the issue of unreasonable speeds. She checks the histogram of *cal_speed* in Point View and observes an unusually high label value under the last bar. Upon checking, she finds that the speeds of these points exceed 180 km/h, which is clearly unreasonable (Figure 5B). Therefore, she decides to remove these outlier points by segmenting the trajectories. Ultimately, Emily successfully addresses various data quality issues using TrajectoryCurer, significantly enhancing the quality and reliability of the dataset and providing a solid foundation for subsequent analysis work.

6.2 Data Transformation for Analysis Requirements

After cleaning the dataset with TrajectoryCurer, Emily refocuses on her analysis needs. She aims to analyze travel trajectories in Haidian District, Beijing, during weekends. She finds that TrajectoryCurer offers various preprocessing options and decides to continue using it to generate a trajectory dataset that aligns with her research requirements.

Initially, the previous data cleaning operations disrupted the sequential order of certain taxi identifiers (*taxi_id*). For example, when segmenting the trajectories associated with *taxi_id* 1, the resulting sub-trajectories are renamed as *1#1*, *1#2*, and so forth. Emily first re-numbers and sorts the cleaned dataset to ensure the continuity and uniqueness of the *taxi_id* values.

Subsequently, she utilizes the area clipping tool within Map View and uploads the geographical boundary data file for Haidian District, Beijing. TrajectoryCurer updates the clipped dataset in real time and renders the clipped area and trajectories in Map View (Figure 1).

Next, Emily selects the function to filter data points and notices a *weekend* option, finding TrajectoryCurer highly convenient for data filtering. After she clicks it, the weekend trajectory data is extracted. Given the previous operations of area clipping and trajectory extraction, the number of trajectory points has significantly decreased. Emily recognizes that an insufficient number of trajectory points may compromise the representativeness of the data and the accuracy of her analyses. She decides to utilize the point number slider to set a filtering condition, retaining only those trajectories that contain at least five sampling points (Figure 2B).

Emily also notices inconsistent units in the data: time is in seconds, distance in meters, and speed in km/h. This unit configuration is preset by the system to facilitate analysis of trajectory dataset characteristics. To ensure consistency in data units and facilitate subsequent analysis, she converts the speed from km/h to m/s.

Finally, Emily exports the optimized dataset. Through these operations, she experiences TrajectoryCurer's powerful capabilities in identifying and addressing data quality issues, as well as its efficiency and flexibility in trajectory data processing. The system significantly simplifies data preprocessing complexities, enhancing work efficiency and providing a reliable foundation for subsequent analysis.

7 Domain Expert Interview

We conducted one-on-one interviews with five domain experts: three professors and two doctoral students specializing in Human-Computer Interaction, Geographic Information Systems, and Data Science, all with extensive experience in trajectory data processing. After presenting the system's design philosophy and demonstrating the workflow, the experts interacted with TrajectoryCurer and replicated the usage scenarios.

The experts unanimously acknowledged that trajectory data quality issues are prevalent and significantly impact practical applications. One expert emphasized that in urban traffic flow analysis, GPS drift and discontinuous timestamps can lead to misjudgments that affect traffic planning and emergency vehicle routing. A GIS expert noted that quality issues not only compromise analysis accuracy but also substantially reduce efficiency, as manual checking consumes considerable time and increases labor costs. The experts praised TrajectoryCurer's interactive visualization capabilities for intuitively displaying multidimensional trajectory characteristics and quality issues. They found the automated detection and refinement operations highly practical, enabling rapid identification and correction of common problems. As one expert remarked, "The system has a gentle learning curve. Complex cleaning tasks that were previously tedious can now be completed through a few interactive operations." The experts particularly valued the explicit mapping between functional modules and the 19-issue taxonomy, describing it as "a professional checklist that transforms data cleaning from fragmented empirical judgment into a systematic verification process." The experts also suggested optimizing large-scale data handling to reduce potential latency and integrating advanced AI techniques to enhance accuracy in complex scenarios. These insights will guide future system optimization.

8 Discussion

We discuss the lessons learned from designing TrajectoryCurer, comparisons with existing workflows and tools, limitations, and future work.

8.1 Lessons Learned

We learned two lessons during TrajectoryCurer's development.

First, expert interviews revealed that assessing trajectory quality solely by outlier count is insufficient. Trajectory length, outlier proportion, and outlier distribution are equally critical. For example, a short trajectory may consist entirely of outliers, while a longer one with more outliers but a lower proportion may actually have superior quality. To address this, we added pie charts alongside the outlier bar charts in the LineUp view to visualize outlier proportions together with absolute counts.

Second, the high point density in trajectory data makes it difficult to discern distributions in scatter plots. Reducing point transparency proved ineffective. We realized that not all points require equal attention, so we adopted color-coding to highlight speed outliers and implemented selective hiding of out-of-range points when users select specific categories (Figure 5). This design improves both visualization clarity and user focus.

8.2 Comparison with Existing Workflows and Tools

To systematically evaluate the efficacy of TrajectoryCurer, we compare it against three mainstream workflows: Scripting (represented by Python), GIS Tools (represented by QGIS [44]), and General Wrangling Tools (represented by Profiler [27] and CoWrangler [13]). Adopting the evaluation methodology of RidgeBuilder [40], we strictly delimit our assessment scope to the trajectory quality taxonomy (see Tab. 1) and design requirements (see Section 4) proposed in this paper. Based on the extent to which each baseline supports the functions within this scope, we define four evaluation levels:

- **Direct Support:** The baseline provides native configurations that cover all functionality of TrajectoryCurer.
- **Indirect Support:** The baseline can fully replicate the functionality but relies on manual coding or complex configuration combinations, imposing strict requirements on user expertise.
- **Limited Support:** The baseline covers only a subset of the functional space supported by TrajectoryCurer.
- **Not Supported:** The baseline's configuration or underlying data model cannot support functions in this dimension.

We elaborate on three aspects: quality issue detection, visualization and interaction, and wrangling operations (Figure 7).

Quality issue detection. *Scripting* is comprehensive but requires manual coding. *GIS Tools* specialize in the spatial geometric dimension and excel at identifying geographic and map-matching anomalies but lack support for temporal sequences and physical logic. *General Wrangling Tools* are restricted to numerical constraints, failing to detect spatial and sequence anomalies. In contrast, TrajectoryCurer covers all the issues in the taxonomy.

✓ Direct Support ○ Indirect Support △ Limited Support × Not Support

	Quality Issue Detection						Vis.&Int.	Wrangling
	Sample	Interval	Speed	Oth.Attr.	Traj.	Map Mat.		
Scripting	○	○	○	○	○	○	×	○
GIS Tools	△	△	△	△	△	△	△	△
Wran.Tools	△	×	△	△	×	×	△	△
TrajCurer	✓	✓	✓	✓	✓	✓	✓	✓

Figure 7: This matrix displays the functional commonalities and differences between TrajectoryCurer and three mainstream workflows used in practice. These workflows include Scripting (represented by Python), GIS Tools (represented by QGIS [44]), and General Wrangling Tools (represented by Profiler [27] and CoWrangler [13]). The comparison covers three core dimensions: Quality Issue Detection, Visualization Interaction, and Wrangling Operations. The colors encode the level of support.

Visualization and interaction. *Scripting* is typically limited to generating static charts, lacking dynamic interactivity. *GIS Tools* and *General Wrangling Tools* specialize in spatial and statistical interactions, respectively, but neither supports coordinated linking between maps and charts. In contrast, TrajectoryCurer achieves multidimensional exploration and interaction through the deep linking of Data View, Point View, Trajectory View, and Map View.

Wrangling operations. *Scripting* offers flexibility but requires programming skills. *GIS Tools* specialize in geometric editing but rely on embedded scripting for some operations, while *General Wrangling Tools* handle numerical cleaning but lack spatiotemporal semantics. In contrast, TrajectoryCurer integrates these functions directly into the visual interface, providing direct support without requiring coding.

8.3 Limitations and Future Work

Despite TrajectoryCurer's effectiveness, several limitations warrant discussion. First, testing was conducted on a limited set of trajectory datasets. While these encompass common quality issues, the system may experience performance degradation with large-scale data, affecting real-time interaction responsiveness. Second, our classification framework of 19 trajectory quality issues lacks systematic evaluation. It remains unclear whether this taxonomy comprehensively covers user-encountered challenges or whether the detection rules are appropriately designed. Third, we deliberately prioritized 2D visualization to ensure operational precision for urban ground traffic, as 3D techniques often introduce occlusion and interaction complexity that hinder precise cleaning; however, this limits applicability to scenarios involving altitude-varying trajectories [8]. Fourth, the system supports basic repair operations (deletion, segmentation, imputation) but lacks advanced techniques such as trajectory reconstruction or map-matching-based correction.

Additionally, while we provide qualitative comparisons against mainstream workflows, our evaluation lacks quantitative evidence of performance gains. Future work will include user studies comparing TrajectoryCurer against scripting-based and GIS-based approaches to measure task completion time, error rates, and efficiency improvements. We also plan to empirically validate how different cleaning configurations affect downstream tasks such as map-matching accuracy, stay-point detection, and trajectory clustering, acknowledging that cleaning involves trade-offs between noise removal and preservation of rare but meaningful behaviors.

9 Conclusion

We propose TrajectoryCurer, a visual analytics system designed for comprehensive profiling and transformation of raw trajectory data. Based on a literature review and expert interviews, we developed a taxonomy of 19 data quality issues across six dimensions and a corresponding set of preprocessing operations. The system integrates automated anomaly detection, multidimensional visualization, and interactive cleaning and transformation functionalities through four coordinated views: Data View, Trajectory View, Point View, and Map View. Validation through real-world scenarios and expert evaluations demonstrates that TrajectoryCurer significantly enhances preprocessing efficiency and provides flexible tool support for data exploration, diagnosis, and preparation. The source code is available at `https://xiaodan-miao.github.io/TrajectoryCurer/`.

Acknowledgments

We thank the anonymous reviewers for their constructive comments. This work was supported in part by NSFC (62402421) and Ningbo Yongjiang Talent Programme (2024A-399-G).

REFERENCES

[1] Z. Abedjan, L. Golab, and F. Naumann. Profiling relational data: a survey. *The VLDB Journal*, 24(4):557–581, 2015. 2

[2] A. P. Afonso, A. Ferreira, L. Ferreira, and R. Vaz. RoseTrajVis: Visual analytics of trajectories with rose diagrams. In *International Conference Information Visualisation (IV)*, pages 378–384. IEEE, 2020. 3

[3] S. Al-Dohuki, F. Kamw, Y. Zhao, X. Ye, J. Yang, and S. Jamonnak. An open source TrajAnalytics software for modeling, transformation and visualization of urban trajectory data. In *IEEE Intelligent Transportation Systems Conference*, pages 150–155. IEEE, 2019. 2, 3

[4] S. Al-Dohuki, Y. Wu, F. Kamw, J. Yang, X. Li, Y. Zhao, X. Ye, W. Chen, C. Ma, and F. Wang. Semantictraj: A new approach to interacting with massive taxi trajectories. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):11–20, 2017. 2, 3

[5] G. Andrienko, N. Andrienko, P. Bak, D. Keim, and S. Wrobel. *Visual Analytics of Movement*. Springer, 2013. 2

[6] A. Basiri, M. Jackson, P. Amirian, A. Pourabdollah, M. Sester, A. Winstanley, T. Moore, and L. Zhang. Quality assessment of OpenStreetMap data using trajectory mining. *Geo-spatial Information Science*, 19(1):56–68, 2016. 3

[7] M. Bostock, V. Ogievetsky, and J. Heer. D3: Data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, 2011. 5

[8] F. Chen, Y. Yu, L. Ni, Z. Zhang, and Q. Lu. Dstvis: toward better interactive visual analysis of drones' spatio-temporal data. *Journal of Visualization*, 27:623–638, 2024. 9

[9] 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. 2

[10] Y. Chen, J. Wu, Y. Wu, and D. Liu. T-foresight: Interpret moving strategies based on context-aware trajectory prediction. *Visual Informatics*, page 100261, 2025. 2

[11] Y. Chen, H. Zhang, W. Sun, and B. Zheng. RNTrajRec: Road network enhanced trajectory recovery with spatial-temporal transformer. In *IEEE International Conference on Data Engineering (ICDE)*, pages 829–842. IEEE, 2023. 4

[12] Y.-Y. Chiang, B. Wu, A. Anand, K. Akade, and C. A. Knoblock. A system for efficient cleaning and transformation of geospatial data attributes. In *Proceedings of ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pages 577–580. ACM, 2014. 3

[13] B. Chopra, A. Fariha, S. Gulwani, A. Z. Henley, D. Perelman, M. Raza, S. Shi, D. Simmons, and A. Tiwari. CoWrangler: Recommender system for data-wrangling scripts. In *Companion of the 2023 International Conference on Management of Data (SIGMOD)*, pages 147–150. ACM, 2023. 2, 3, 9

[14] Z. Deng, D. Weng, J. Chen, R. Liu, Z. Wang, J. Bao, Y. Zheng, and Y. Wu. Airvis: Visual analytics of air pollution propagation. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):800–810, 2019. 1

[15] Z. Deng, D. Weng, S. Liu, Y. Tian, M. Xu, and Y. Wu. A survey of urban visual analytics: Advances and future directions. *Computational Visual Media*, 9(1):3–39, 2023. 1

[16] H. Ding, M. Cai, X. Lin, T. Chen, L. Li, and Y. Liu. RTVEMVS: Real-time modeling and visualization system for vehicle emissions on an urban road network. *Journal of Cleaner Production*, 309:127166, 2021. 3

[17] Z. Feng and Y. Zhu. A survey on trajectory data mining: Techniques and applications. *IEEE Access*, 4:2056–2067, 2016. 2

[18] S. Gratzl, A. Lex, N. Gehlenborg, H. Pfister, and M. Streit. LineUp: Visual analysis of multi-attribute rankings. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2277–2286, 2013. 6

[19] M. Hameed and F. Naumann. Data preparation: A survey of commercial tools. *ACM SIGMOD Record*, 49(3):18–29, 2020. 3

[20] X. Huang, Y. Zhao, C. Ma, J. Yang, X. Ye, and C. Zhang. TrajGraph: A graph-based visual analytics approach to studying urban network centralities using taxi trajectory data. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):160–169, 2015. 2, 3

[21] Y. Huang, Y. Miao, D. Weng, A. Perer, and Y. Wu. Structvizor: Interactive profiling of semi-structured textual data. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 1–18, 2025. 2

[22] Z. Huang, Y. Zhao, W. Chen, S. Gao, K. Yu, W. Xu, M. Tang, M. Zhu, and M. Xu. A natural-language-based visual query approach of uncertain human trajectories. *IEEE Transactions on Visualization and Computer Graphics*, 26(1):1256–1266, 2019. 3

[23] C. Hurter, O. Ersoy, S. I. Fabrikant, and A. C. Telea. Bundled visualization of dynamic graph and trail data. *IEEE Transactions on Visualization and Computer Graphics*, 20(8):1141–1157, 2014. 2

[24] F. Jin, W. Hua, T. Zhou, J. Xu, M. Francia, M. E. Orlowska, and X. Zhou. Trajectory-based spatiotemporal entity linking. *IEEE Transactions on Knowledge and Data Engineering*, 34(9):4499–4513, 2022. 4

[25] Z. Jin, M. R. Anderson, M. Cafarella, and H. V. Jagadish. Foofah: Transforming data by example. In *Proceedings of the ACM International Conference on Management of Data*, pages 683–698. ACM, 2017. 3

[26] H. Joffe. Thematic analysis. In D. Harper and A. R. Thompson, editors, *Qualitative Research Methods in Mental Health and Psychotherapy: A Guide for Students and Practitioners*, chapter 15, pages 209–223. John Wiley & Sons, 2011. 3

[27] S. Kandel, R. Parikh, A. Paepcke, J. M. Hellerstein, and J. Heer. Profiler: Integrated statistical analysis and visualization for data quality assessment. In *International Working Conference on Advanced Visual Interfaces*, pages 547–554. ACM, 2012. 2, 9

[28] M. Khan, L. Xu, A. Nandi, and J. M. Hellerstein. Data tweening: Incremental visualization of data transforms. *Proceedings of the VLDB Endowment*, 10(6):661–672, 2017. 2

[29] M.-J. Kraak. The space-time cube revisited from a geovisualization perspective. In *Proceedings of the 21st International Cartographic Conference*, pages 1988–1996. International Cartographic Association, 2003. 2

[30] R. Krueger, D. Thom, and T. Ertl. Semantic enrichment of movement behavior with foursquare–a visual analytics approach. *IEEE Transactions on Visualization and Computer Graphics*, 21(8):903–915, 2014. 3

[31] R. Krüger, D. Thom, M. Wörner, H. Bosch, and T. Ertl. TrajectoryLenses- a set-based filtering and exploration technique for long-term trajectory data. In *Proceedings of the 15th Eurographics Conference on Visualization*, pages 451–460. The Eurographs Association, 2013. 2, 3

[32] D. Li, H. Mei, Y. Shen, S. Su, W. Zhang, J. Wang, M. Zu, and W. Chen. ECharts: A declarative framework for rapid construction of web-based visualization. *Visual Informatics*, 2(2):136–146, 2018. 5

[33] L. Li, X. Chen, Q. Liu, and Z. Bao. A data-driven approach for GPS trajectory data cleaning. In *International Conference on Database Systems for Advanced Applications (DASFAA)*, pages 3–19. Springer, 2020. 4

[34] L. Li, T. Peng, and J. Kennedy. A rule based taxonomy of dirty data. In *Proceedings of Annual International Academic Conference on Data Analysis, Data Quality and Metadata Management*, pages 140–148. Global Science and Technology Forum (GSTF), 2011. 3

[35] Y. Liang, H. Wen, Y. Xia, M. Jin, B. Yang, F. Salim, Q. Wen, S. Pan, and G. Cong. Foundation models for spatio-temporal data science: A tutorial and survey. In *Proceedings of the 31st ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD)*, pages 6063–6073. ACM, 2025. 3

[36] D. Liu, D. Weng, Y. Li, J. Bao, Y. Zheng, H. Qu, and Y. Wu. Smartadp: Visual analytics of large-scale taxi trajectories for selecting billboard locations. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):1–10, 2016. 3

[37] H. Liu, X. Chen, Y. Wang, B. Zhang, Y. Chen, Y. Zhao, and F. Zhou. Visualization and visual analysis of vessel trajectory data: A survey. *Visual Informatics*, 5(4):1–10, 2021. 1

[38] H. Liu, S. Jin, Y. Yan, Y. Tao, and H. Lin. Visual analytics of taxi trajectory data via topical sub-trajectories. *Visual Informatics*, 3(3):140–149, 2019. 1

[39] S. Liu, G. Andrienko, Y. Wu, N. Cao, L. Jiang, C. Shi, Y.-S. Wang, and S. Hong. Steering data quality with visual analytics: The complexity

challenge. *Visual Informatics*, 2(4):191–197, 2018. 2

[40] S. Liu, Y. Liu, J. Li, Y. Huang, Y. Shangguan, Z. Deng, D. Weng, and Y. Wu. RidgeBuilder: Interactive authoring of expressive ridgeline plots. In *Proc. CHI Conference on Human Factors in Computing Systems (CHI)*, pages 392:1–392:18. ACM, 2025. 9

[41] M. Lu, J. Liang, Z. Wang, and X. Yuan. Exploring OD patterns of interested region based on taxi trajectories. *Journal of Visualization*, 19(4):811–821, 2016. 3

[42] Mapbox. Mapbox GL JS. https://www.mapbox.com/mapbox-gl-js. Accessed: January 5, 2026. 5

[43] J. Muckell, P. W. Olsen, Jr., J.-H. Hwang, C. T. Lawson, and S. S. Ravi. Compression of trajectory data: A comprehensive evaluation and new approach. *GeoInformatica*, 18(3):435–460, 2014. 4

[44] QGIS Development Team. QGIS geographic information system. https://www.qgis.org. Accessed: January 5, 2026. 9

[45] T. Qin and S. Shen. Online temporal calibration for monocular visual-inertial systems. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3662–3669. IEEE, 2018. 4

[46] B. Russig, D. Groß, R. Dachselt, and S. Gumhold. On-tube attribute visualization for multivariate trajectory data. *IEEE Transactions on Visualization and Computer Graphics*, 29(1):1288–1298, 2023. 2

[47] R. Scheepens, N. Willems, H. Van de Wetering, and J. J. Van Wijk. Composite density maps for multivariate trajectories. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2518–2527, 2011. 2

[48] D. Shin, J. Jo, B. Kim, H. Song, S.-H. Cho, and J. Seo. RCMVis: A visual analytics system for route choice modeling. *IEEE Transactions on Visualization and Computer Graphics*, 29(3):1799–1817, 2021. 3

[49] Tableau Software. *Tableau Public (Version 2024.3)*, 2024. Accessed: January 5, 2026. 7

[50] J. Tang, H. Ren, Z. Deng, D. Weng, T. Tang, L. Yu, J. Bao, Y. Zheng, and Y. Wu. Towards better illegal chemical facility detection with hazardous chemicals transportation trajectories. *Journal of Visualization*, 28:535–558, 2025. 2

[51] C. Tominski, H. Schumann, G. Andrienko, and N. Andrienko. Stacking-based visualization of trajectory attribute data. *IEEE Transactions on visualization and Computer Graphics*, 18(12):2565–2574, 2012. 2, 3

[52] J. W. Tukey. *Exploratory Data Analysis*. Addison-Wesley Pub. Co., 1977. 6

[53] D. Weng, H. Zhu, J. Bao, Y. Zheng, and Y. Wu. Homefinder revisited: Finding ideal homes with reachability-centric multi-criteria decision making. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pages 1–12, 2018. 1

[54] W. Wu, J. Xu, H. Zeng, Y. Zheng, H. Qu, B. Ni, M. Yuan, and L. M. Ni. TelCoVis: Visual exploration of co-occurrence in urban human mobility based on telco data. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):935–944, 2015. 1

[55] K. Xiong, X. Xu, S. Fu, D. Weng, Y. Wang, and Y. Wu. JsonCurer: Data quality management for JSON based on an aggregated schema. *IEEE Transactions on Visualization and Computer Graphics*, 30(6):3008–3021, 2024. 3

[56] J. Yuan, Y. Zheng, C. Zhang, W. Xie, X. Xie, G. Sun, and Y. Huang. T-drive: Driving directions based on taxi trajectories. In *ACM SIGSPATIAL International Symposium on Advances in Geographic Information Systems*, pages 99–108. ACM, 2010. 7

[57] K. E. Zannat, C. F. Choudhury, S. Hess, and D. Watling. Investigating the relative accuracy of GPS, GSM and CDR data for inferring spatiotemporal travel trajectories. *IET Intelligent Transport Systems*, 18(S1):3013–3033, 2024. 4

[58] Y. Zheng. Trajectory data mining: An overview. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 6(3):29:1–29:41, 2015. 2

[59] M. Zhou, T. Ai, G. Zhou, and W. Hu. A visualization method for mining colocation patterns constrained by a road network. *IEEE Access*, 8:51933–51944, 2020. 3

[60] J. Zhu, T. Wang, D. Hu, Z. Fang, L. Chen, Y. Gao, T. Li, and C. S. Jensen. T-Assess: An efficient data quality assessment system tailored for trajectory data. *Proceedings of the VLDB Endowment*, 18(3):666–674, 2024. 1, 3, 4