MoCo: Urban User Mobile Contact Detection Based on Cellular Signaling Trace

Sijing Duan, *Member, IEEE*, Feng Lyu, *Senior Member, IEEE*, Jing Zhang, *Student Member, IEEE*, Huali Lu, *Student Member, IEEE*, Peng Yang, *Member, IEEE*, Huaqing Wu, *Member, IEEE*, Yaoxue Zhang, *Senior Member, IEEE*, and Xuemin Shen, *Fellow, IEEE*

***Abstract*—Mobile contact exhibits user co-traveling events within the same transportation tool, which is crucial for resident profiling, face-to-face interaction detection, etc. In this paper, we investigate urban user mobile contact detection with cellular signaling traces, which is cost-efficient to enable large-scale detection. Specifically, we develop a data collection platform to collect substantial user signaling traces, covering different types of road scenarios within a city. With the collected traces, we perform systematic data analysis to reveal several technical challenges, which are sparsity of signaling trajectory, remote base station noise, and fuzzy matching difficulties. To address challenges, we propose a mobile contact detection method named MoCo. In MoCo framework, we first conduct data denoising to remove the noise from remote base stations. Then, we devise a spatio-temporal filter to eliminate unlikely mobile contact traces in both spatial and temporal domains, reducing the computational overhead. Finally, we design a detection network that integrates the submodules of data alignment, feature encoder, spatio-temporal representation learner, and user mobile contact detector. Extensive evaluation results demonstrate the superiority of MoCo in comparison with state-of-the-art baselines. Robust experiments show that MoCo can work efficiently in different transportation modes and urban densities.**

***Index Terms*—Mobile contact detection, cellular signaling trace, user correlation sensing.**

Received 8 May 2024; revised 28 November 2024; accepted 19 February 2025. Date of publication 24 February 2025; date of current version 3 July 2025. This work was supported in part by the National Natural Science Foundation of China under Grant 62422216, Grant 62320106006, and Grant 62402279, in part by Central South University Innovation-Driven Research Program under Grant 2023CXQD029, and in part by the 111 Project under Grant B18059. Recommended for acceptance by H. Yao. *(Corresponding author: Feng Lyu.)*

Sijing Duan is with the School of Computer Science and Engineering, Central South University, Changsha, Hunan 410083, China, and also with the Department of Computer Science and Technology, Tsinghua University, Beijing 100190, China (e-mail: duansijing@csu.edu.cn).

Feng Lyu, Jing Zhang, and Huali Lu are with the School of Computer Science and Engineering, Central South University, Changsha, Hunan 410083, China (e-mail: fenglyu@csu.edu.cn; jingzh@csu.edu.cn; huali_lu@csu.edu.cn).

Peng Yang is with the School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan, Hubei 430074, China (e-mail: yangpeng@hust.edu.cn).

Huaqing Wu is with the Department of Electrical and Software Engineering, University of Calgary, Calgary, AB T2N 1N4, Canada (e-mail: huaqing.wu1@ucalgary.ca).

Yaoxue Zhang is with the Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China (e-mail: zhangyx@tsinghua.edu.cn).

Xuemin Shen is with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: sshen@uwaterloo.ca).

Digital Object Identifier 10.1109/TMC.2025.3545437

I. Introduction

MOBILE contact is one of the interactions between individuals, where users exhibit similar spatio-temporal behaviors, such as co-traveling in the same vehicle or being in close physical proximity. Understanding how users engage in mobile contact is essential for various activities and applications, including public health, public security, and urban planning [1], [2]. Particularly, in public health, it aids in contact tracing and exposure notifications during disease outbreaks, allowing health authorities to identify and notify individuals potentially exposed to contagious diseases, helping control infection spread. In public security, it supports surveillance of gatherings and events, enabling authorities to monitor for safety threats, and facilitates emergency response by assisting rescuers in locating and aiding individuals in need. For urban planning and behavioral studies, mobile contact data provides insights into mobility patterns, enabling optimized infrastructure and traffic management, and advances social interaction studies that reveal community dynamics and behaviors [3], [4].

Fig. 1 illustrates an example scenario of urban mobile contact, people who carry mobile devices such as smartphones and travel together in a shared vehicle are considered mobile contact users. During movement, their mobile devices establish connections with base stations (BS) and nearby devices. Detecting mobile user contact behaviors is an effective solution to quickly identify individuals with close physical connections.

Several studies in the literature have explored the detection of human-to-human interaction [5], [6], [7], contact between human objects [8], [9], [10], and contact tracking [1], [11], [12]. For example, Wang et al. [7] proposed a human-to-human interaction detection dataset and benchmark by leveraging a Transformer-based model. Lim et al. [8] designed an efficient and reliable human-object interaction detection network. Zhang et al. [1] developed a contact tracing method for healthcare workers using low-energy Bluetooth beacons. However, these methods primarily focus on detecting interaction or contact from video frames, which may not be effective in detecting mobile contact behaviors between users in natural environments for two reasons. First, video surveillance is not universally deployed in cities. Second, users have extensive spatial movement ranges, making continuous signal collection challenging and limiting the use of video frames for detection. However, existing contact-tracing methods are designed for small-scale scenarios. These limitations motivate us to leverage ubiquitous data, such

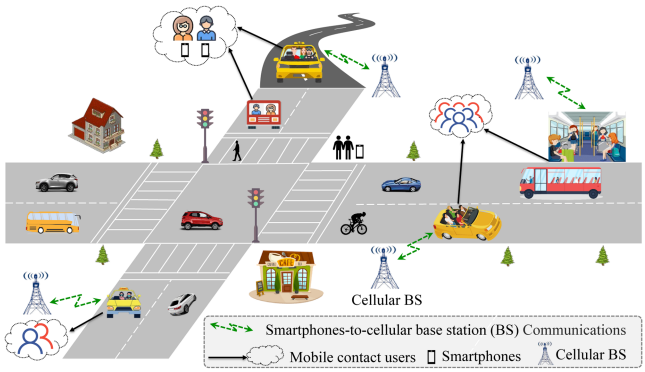


Fig. 1. Illustration of urban user mobile contact.

as wireless signals, to enable large-scale detection of mobile contact between users. In recent years, research has extensively investigated wireless sensing using cellular signals [13], [14], [15], [16], [17], [18], WiFi [19], [20], [21], [22], [23] and mmWave signals [24], [25]. However, most of the work focuses on sensing individual user status and does not adequately consider the correlation between user contacts.

The extensive application of cellular signal data has the potential to facilitate the detection of mobile contacts among a large number of users in a cost-effective manner. Unlike video surveillance and detailed GPS data, cellular signaling traces are collected passively and cover almost all mobile phones [26]. However, there are several challenges in using cellular data for urban user mobile contact detection. First, to ensure the applicability of the detection approach to all cellular networks, we can only rely on cellular signaling traces that record the user's association with a specific BS and the time of association. These traces are universally accessible on all network systems. However, the temporal and spatial distribution of the cellular signaling traces is sparse as a result of the extensive coverage of the BS, making it difficult to characterize the routes taken by users. Second, complex network management and scheduling rules result in variations in BS associated with users and the times at which these associations occur, even when users are traveling along the same route. This disparity makes it challenging to match the asynchronous mobile contact traces. Third, the coarse-grained nature of the cellular signaling traces means that when users travel closely together in different vehicles, their cellular BS trajectories exhibit similar patterns, further complicating the detection of mobile contacts between users.

To address these challenges, one possible approach is to initially map the BS trajectories of users on the road map and then compare their travel times. However, using mapmatching methods for mobile contact detection may not be efficient for several reasons. (i) Map matching algorithms mainly focus on matching cellular BS trajectories of individual users with actual road routes, disregarding the fact that mobile contact events are closely associated with both travel time and movement routes. (ii) The map matching process is laborious and susceptible to significant errors if there is even a slight deviation in mapping users' routes. Therefore, we develop an end-to-end model for the detection of mobile contact using cellular signaling traces of two users. This method can eliminate the need for precise mapping of user routes on the road map, thus avoiding map matching errors.

This paper focuses on the detection of mobile contacts between urban users using cellular signaling traces. The study begins by developing a mobile application to collect cellular signaling data and establishing a data collection platform with 20 Android-based mobile devices. Then a comprehensive data collection campaign is carried out that includes various urban roads and types of vehicles. The collected traces are subjected to extensive data analysis, which reveals several technical challenges related to the scarcity of moving trajectories, interference from remote BS, and challenges in achieving accurate matching.

Based on the insights obtained from the data analysis, we propose a new framework called MoCo, i.e., Mobile Contact detection. This framework consists of three main technical components. First, the *Data Denoising* module aims to eliminate noise from remote BS using the Density-Based Spatial Clustering of Applications (DBSCAN) algorithm. Second, the *Spatio-Temporal Filter* module removes unlikely mobile contact traces in both the spatial and temporal domains to reduce computational overhead. Finally, the *Detection Network* module is responsible for detecting mobile contact behavior by incorporating data alignment, feature encoding, representation learning, and behavior detection. To assess the performance of MoCo, we conduct extensive experiments using data-driven approaches. The results demonstrate the effectiveness of MoCo with an average F1-score of 87.13%, exceeding the performance of the existing baselines. Furthermore, robust experiments validate the efficiency of MoCo across different transportation modes such as cars, subways, and buses. In summary, the main contributions of this paper can be summarized as follows.

- To the best of our knowledge, this is the first work to study the mobile contact detection of urban users. To support our research, we develop a platform to collect a significant amount of cellular signaling traces across a wide range of cities.
- We perform a systematic data analysis on the collected cellular signaling traces. With data-driven insights, we design a novel framework MoCo, which comprises three components for data denoising, spatio-temporal filtering, and mobile contact detection, respectively.
- To evaluate the effectiveness of our proposed MoCo, we collect three different data sets from the real-world cellular network, including cars, subways, and buses. We demonstrate that MoCo outperforms existing state-of-the-art baselines and exhibits robustness in various transportation modes.

The remainder of this paper is organized as follows. We describe the system and define our research problem in Section II. In Section III, we detail our data collection scheme. In Section IV, we perform a data analysis and highlight several challenges. After that, we elaborate on the design of MoCo in Section V. We conduct extensive experiments in Section VI, and discuss the limitations of our work in Section VII. Finally, we review the related work in Section VIII and conclude the paper in Section IX.

II. System Description and Problem Definition

A. System Description

The widespread use of mobile phones has allowed cellular data to be used to detect mobile contacts on a large scale. When a user carries a mobile phone with a SIM card, the cellular networks can record information about the user's network access, including the BS they are connected to, their data usage, and their phone call history. If a user frequently accesses the Internet while traveling, more detailed log information can be collected. However, even without considering the user's specific networking behavior, the network can still collect basic cellular signaling data for each user reliably.

To ensure that the proposed method can be applied to various cellular systems, we utilize the basic cellular signaling trace as the detection evidence for this research. These traces consist of information such as <user ID, BS ID, timestamp>, which are recorded each time a user connects to a new BS. It is important to note that for the cellular service provider, the BS ID can be linked to the geographical location of the BS, represented by longitude and latitude.

B. Problem Definition

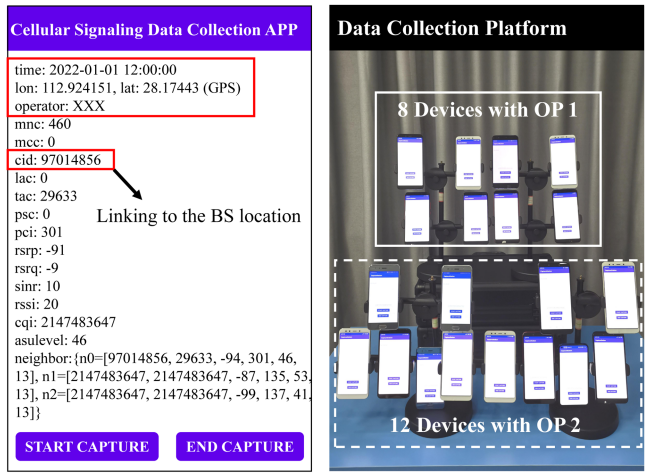
Mobile contact detection can be categorized into human-to-human contact (e.g., face-to-face conversations and fighting), human-to-object contact (e.g., picking up items), proximity-based contact (e.g., detection through wifi and BS connectivity). This paper focuses on proximity-based contact, specifically detecting mobile contacts among urban users who exhibit similar spatio-temporal traveling behaviors. We aim to identify events where users are co-traveling within the same transportation tool. Particularly, we leverage cellular signaling trace data to achieve mobile contact detection, which records the connections between mobile devices and BS at overlapping time intervals as users move through the urban environment. We formulate the problem as follows.

Definition 1 (Cellular Signaling Trace/BS Trajectory): A cellular signaling trace or BS trajectory is a sequence of timestamped BS generated by a mobile user. We denote the traces of two users $u_i, u_j \in \mathcal{N}$ by $S_i = \{(bs_1^i, t_1^i), (bs_2^i, t_2^i), \ldots, (bs_m^i, t_m^i)\}$ and $S_j = \{(bs_1^j, t_1^j), (bs_2^j, t_2^j), \ldots, (bs_m^j, t_m^j)\}$, respectively, where $\mathcal{N}$ is the set of users, bs_m^i is the m-th associated BS by user u_i, t_m^i represents the timestamp of user u_i when connecting to bs_m^i.

Definition 2 (Mobile Contact Detection Problem): Given a set of user cellular signaling trace $\mathcal{S} = \{S_1, S_2, \ldots, S_{|N|}\}$ and a set of users $\mathcal{N} = \{u_1, u_2, \ldots, u_{|N|}\}$, the primary goal of this problem is to detect mobile contact events between users by comparing the similarity scores of their cellular signaling traces. The detection model outputs 1 if a pair of traces (S_u, S_j) belongs to two users with mobile contact; otherwise, it outputs 0.

Solving the problem directly is difficult for several reasons. First, users have asynchronous BS trajectories in both space and time due to different mobility behaviors. Second, even if users have the same routes and encounter mobile contact events, they may connect to different BS due to network scheduling variations caused by user requests and BS loads. This makes it difficult to identify mobile contact behaviors accurately. Third, when users travel closely in different vehicles, their sequential BS trajectories may show similar patterns, making the detection process more complex.

It is important to note that the problem and proposed solutions in this paper can be adapted to address the specific issue of identifying user sets engaged in mobile contact events with a target user. To simulate user mobility events, we provide volunteers with mobile phones during vehicular travel, considering instances where user (device) pairs travel together in the same vehicle as mobile contact behaviors.



(a) (b)

Fig. 2. Cellular signaling data collection platform. (a) is the running interface of the data collection APP. (b) is the platform overview.

III. Data Collection

In this section, we first introduce the data collection platform and then describe our data collection campaigns.

A. Cellular Signaling Data Collection Platform

To support our research, we develop and execute a data collection campaign using an Android app that focuses on collecting cellular signaling data. This app ensures that user privacy is not compromised while providing accurate information on mobile contacts. The app is built on Android 10 and uses the Android.Telephony API to collect details about the network association, the Android.Location API to retrieve user position data, and the I/O API to store the data for offline analysis. The size of the data collection app is 2.64 MB.

Fig. 2(a) illustrates the procedure of running the application, displaying real-time information on the collected data. The user's GPS information, specifically the longitude and latitude, is gathered to facilitate offline analysis. To obtain accurate GPS data as ground truth, we collect GPS data points at one-meter intervals for precise location references. As only data related to BS communication and GPS are collected, the power consumption is minimal during operation. In practical applications,

the data sampling rate can be flexibly adjusted to align with the operator's standards. It can also be adjusted to accommodate specific application requirements with different user mobility patterns, transportation modes, and urban environments.

The cellular signaling data includes the operator-related fields [16], [27], i.e., `mnc`, `mcc`, `cid`, `lac`, `tac`, `psc`, `pci`, `rsrp`, `rsrq`, `sinr`, `rssi`, `cqi`, `asulevel`, and `neighbors`. In particular, the cell identification (`cid`) represents the unique identifier for a cell. The unique device identifier (`tac`) is used to identify a specific device. The cellular signal strength is measured by the received signal strength indicator (`rssi`). The neighboring cellular stations are recorded along with their corresponding field values. When a user connects to a new BS, a cellular signaling sample is collected and assigned a timestamp for temporal analysis. It is important to note that the `cid` field can be used to determine the location of the cellular tower, and this information is obtained through cooperation with the respective operators. After consulting with network operator engineers, it has been verified that the data collection format and granularity align seamlessly with real operator data collection systems. We rely on our own collected data to establish the ground truth for mobile contacts.

B. Data Collection Campaign

We conduct an extensive data collection campaign intending to include all types of urban roads and vehicles.[1] Specifically, we focus on three main road conditions. 1) In urban areas, roads can be bidirectional and range from 4 to 8 lanes wide. These roads experience heavy traffic and have dense BS deployments. Vehicles on these roads tend to move slowly, with speeds ranging from 10 to 30 km/h, due to traffic congestion and traffic lights. 2) In suburban areas, roads are typically bidirectional and range from 4 to 6 lanes wide. Traffic on these roads is lighter and the density of BS is medium. As a result, vehicles can move at faster speeds, ranging from 30 to 60 km/h, without experiencing traffic congestion. 3) On highways, which are bidirectional urban roads with 8 lanes, vehicles can reach speeds of 80 to 110 km/h. Along these highways, BSs are only sparsely deployed. In addition, our data collection campaign includes three transportation modes: cars, subways, and buses.

To reduce the amount of human labor, we install a data collection app on 20 mobile phones. These phones have been distributed to a group of volunteers who travel in 2 to 4 vehicles, maintaining a close distance under normal driving conditions. Within each vehicle, mobile phones can simulate the user's mobile contact events. Fig. 2(b) illustrates the data collection platform, which consists of 20 mobile phones (Huawei Honor 9 or XIAOMI 6X) running the app. Considering that cellular access in a target area is typically provided by multiple operators, we use SIM cards from two different operators for trajectory and signaling data collection to ensure the generalization of the collected data. It should be noted that our proposed method is not limited to the coverage of a single BS. Among these phones, 8 devices are equipped with OP (operator provider) 1 SIM card, while the remaining 12 devices have OP 2 SIM cards. Fig. 3 shows the distribution of data collection areas within a city, taking into account different types of roads. Data are collected while users are traveling in cars. After disregarding devices that exhibited abnormal data collection behaviors, we were able to collect a total of 63, 136, and 42 user (device) trajectories in urban, suburban, and highway environments, respectively. These trajectories accumulated 115, 85, and 11 hours of data, covering distances of 2,112, 3,132, and 960 kilometers, respectively. Furthermore, we collected 874 and 2,303 trajectory pairs under bus and subway transportation modes, which lasted 13.3 hours and 45.6 hours, respectively.

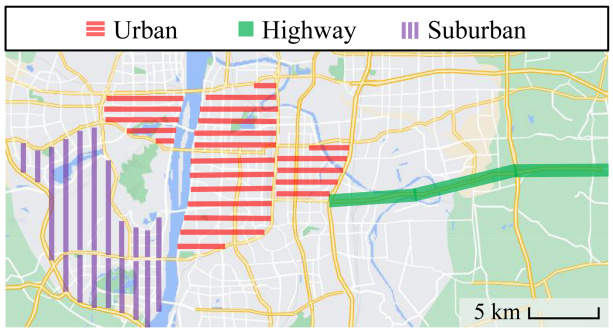


Fig. 3. Data collection in different road types.

C. Data Ethics and Privacy

To address the potential data privacy and security concerns, we take strict measures throughout the data collection and processing stages. In the data collection stage: i) We obtain explicit consent from volunteers. Participants were fully informed of the research objectives, the type of data to be collected, and the purpose of the data. ii) All phone numbers and user identifiers of participating volunteers are encrypted to prevent any association with individual users. iii) The data collection app collects data only when actively in use, and all data collection activities comply with strict security and privacy policies.

In the data processing stage: i) User IDs are hashed into global identifiers, making it impossible to trace back to individual users. ii) Only signaling data from BS switching rather than GPS data are used. The data is sparse and contains significant location noise, thus preventing user precise tracking and protecting user privacy. iii) Data processing is conducted exclusively on the network operator's secure servers, with access restricted to authorized researchers. iv) To obtain crucial information, such as BS IDs and GPS coordinates, we collaborate closely with service providers and obtain explicit permissions. Additionally, our findings do not disclose sensitive or personally identifiable information.

IV. Insights of Data Analytics

A. Challenges

Sparsity of Moving Trajectory: The GPS information has a high level of detail, but BS-based user routes are limited due

[1]The data collection campaign has been discussed with the network operator engineers from the mobile operator to ensure that the campaign aligns with their standards. It is worth noting that the differentiation of RRC states is unnecessary for our data collection in this paper. All collected data are obtained in the RRC Connected State by maintaining device connection with the serving cell.

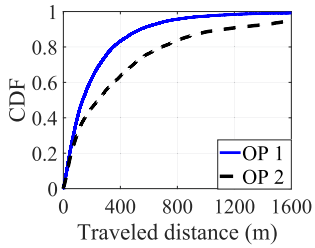


Fig. 4. CDFs of traveled distance between two adjacent samples.

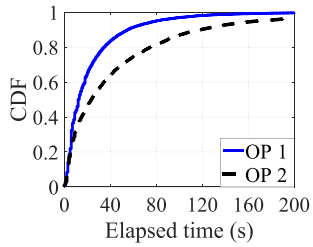


Fig. 5. CDFs of elapsed time between two adjacent samples.

to the density of BS deployed. Fig. 4 illustrates the Cumulative Distribution Functions (CDFs) of the distances traveled by users between two consecutive samples, using GPS data from the collected samples.[2] Users can travel a significant distance from their previous location when a new cellular signaling sample is collected. For OP 1 and OP 2, the median distances traveled are approximately 140 and 250 meters, respectively. Furthermore, more than 20% of the distance values exceed 400 and 800 meters for OP 1 and OP 2, respectively, and some distances even exceed 1,600 meters. Travel distances under OP 1 are generally shorter than those under OP 2, which is expected considering that OP 1 has a higher density of BS in the same geographical areas.

Fig. 5 presents the CDFs of the time elapsed between two consecutive samples for the two operator networks. Similarly, we observe long time intervals between samples, with median elapsed times of up to 13 and 24 seconds for OP 1 and OP 2, respectively. Moreover, more than 20% of the elapsed times exceed 40 and 80 seconds for OP 1 and OP 2, respectively.

Noise of Remote BS: The wireless coverage of BS typically spans from a few hundred meters to several kilometers. Fig. 6 illustrates the distribution of associated BS' locations for 20 mobile phones during a single trip. The black solid line represents the route taken, the red points indicate the associated BS that are close to the route, and the blue points represent the associated BS that are far away from the route. The network connection does not strictly adhere to the principle of connecting to the closest BS. In some cases, users establish connections with BS that are considerably distant. This phenomenon is common in urban areas with challenging terrain or when nearby BS are overloaded, while remote BS offer better service. Consequently, this

[2]It is important to note that our collected dataset presents fine granularity due to our focus on scenarios where vehicles are in close proximity. This presents the most challenging scenario for distinguishing between mobile contact and non-mobile contact events, thereby ensuring the robustness of our model.

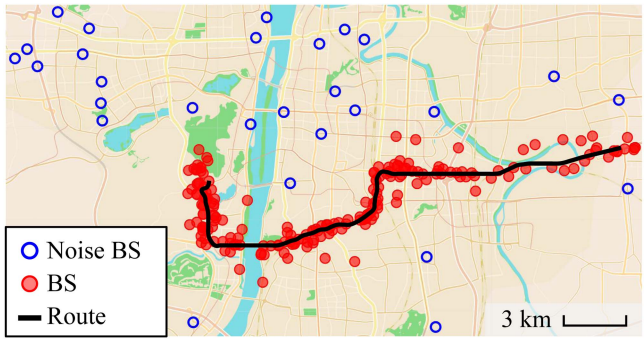


Fig. 6. Location distribution of BS associated by one specific user.

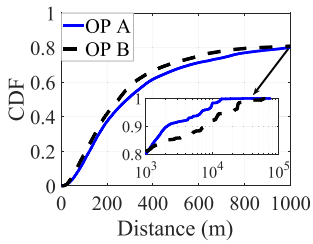


Fig. 7. CDFs of distance between the user and the BS.

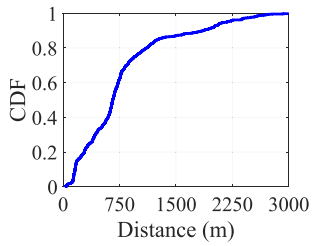


Fig. 8. CDF of separated distance between different vehicles.

introduces significant noise from remote BS when characterizing user-moving routes.

Additionally, Fig. 7 shows the CDFs of the geographical distance between users and the associated BS in two different operator networks, further confirming the presence of noise from the remote BS. It is apparent that for both networks, 80% of the distance samples are less than 1,000 meters. However, the remaining 20% of the samples exhibit a long-tailed distribution that spans a wide range from 1,000 to 10,000 meters. These relatively long-distance samples directly represent the noise from remote BS, which can significantly impact the accuracy of identifying actual moving trajectories and detecting mobile contacts.

Difficulties of Fuzzy Matching: One possible approach to detecting mobile contact events of the user involves matching the trajectories of user motion. However, when there are sparse sequential BS trajectories in both the spatial and temporal domains, fuzzy matching becomes necessary. However, fuzzy matching faces challenges due to complex driving conditions. Fig. 8 illustrates the CDFs of the separated distances between different moving vehicles during trips. It is clear that when

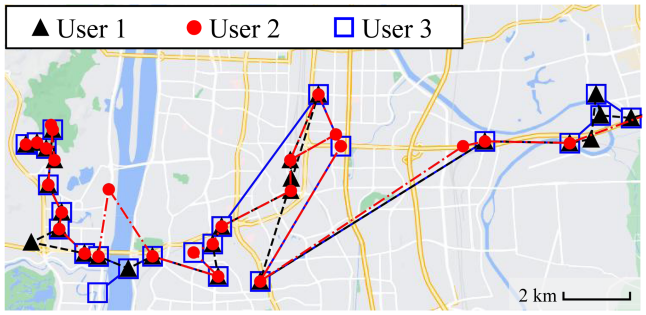


Fig. 9. An example illustrating the difficulty of fuzzy matching.

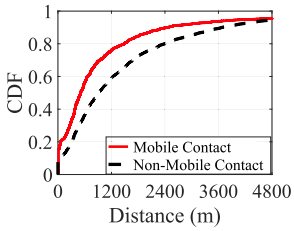


Fig. 10. CDFs of distances between GPS location and associated BS location for two users.

different vehicles follow the same route, the separation distances can be quite small. In particular, the median distance is only 600 meters, with 80% values evenly distributed within a range of 1,200 meters. This observation suggests that it is common for the separated distance between vehicles to be smaller than the granularity of spatial trajectory collection, resulting in very similar sequential BS trajectories. As a result, achieving effective fuzzy matching poses significant challenges.

Fig. 9 illustrates the sequential BS trajectories of three users. In this particular scenario, user 1 and user 2 are traveling together, while user 3 is in a different vehicle. The figure clearly shows that the BS trajectories of the three users appear visually identical, regardless of the specific travel conditions. This highlights the difficulties in achieving accurate fuzzy matching performance, as similar trajectories can arise from closely related motion circumstances, making it challenging to differentiate between them.

B. Implementation Directions

The main idea behind mobile contact events is that users travel together on the same route at the same time. This motivates us to utilize both spatial and temporal data to perform fuzzy matching. In particular, we synchronize the timing to compare the spatial disparities between mobile contact and non-mobile contact events, and vice versa. The differences in spatial distance between the trajectory GPS location and the associated BS location for mobile contact users and non-mobile contact users are shown in Fig. 10, with synchronized time. The figure illustrates the CDFs of the distances of the BS associated with the two types of users. To determine the synchronized BS location, we interpolate using the closest temporal sample. This is done because the user is still within the communication range of the

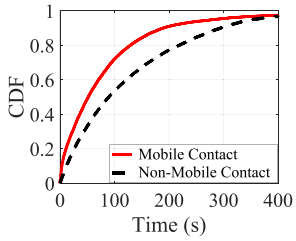


Fig. 11. CDFs of time interval for two users with and without mobile contact arriving in same area.

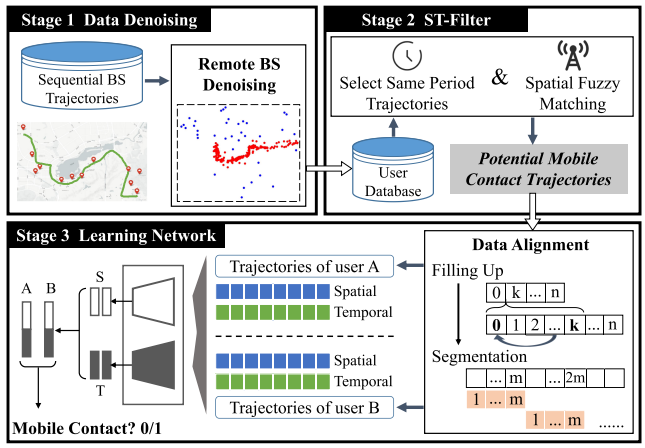


Fig. 12. Design overview of MoCo.

previous BS when no new sample is collected. A clear gap in the distance is observed between the two curves, indicating that the distances for mobile contact events are significantly smaller than those for non-mobile contact events. For example, the median distances for mobile contact and non-mobile contact events are 500 and 900 meters, respectively. Additionally, 80% of distances are smaller than 1,500 and 2,400 meters for mobile contact and non-mobile contact events, respectively.

Similarly, when spatial information is synchronized, we can analyze the temporal differences between the two types of travel. Fig. 11 shows the CDFs of the time intervals for two users: one with mobile contact and one without mobile contact, both arriving in the same area. It can be observed that the time interval samples for users with mobile contact are much smaller than those for users without mobile contact. These noticeable differences in spatial and temporal trajectory patterns serve as compelling evidence for differentiating between mobile contact and non-mobile contact behaviors. This information plays a crucial role in guiding our model design.

V. Design of MoCo

A. Design Overview

In Fig. 12, we present an overview of the design of MoCo, which consists of three main components: data denoising, spatio-temporal filter (ST-Filter) and the learning network. Specifically, the data denoising component utilizes the DBSCAN algorithm to reduce the impact of distant BS. Due to

the computational requirements of detecting mobile contacts on a large scale, we devise the ST-filter component to eliminate unlikely mobile contact trajectories by using spatial fuzzy matching for trajectories within the same period. For the final detection, we integrate four sub-modules into the learning network: data alignment, feature encoder, representation learner, and detector. These modules are responsible for aligning trajectories, encoding spatio-temporal features, learning feature embeddings, and calculating similarity between users' traces, respectively. We then provide detailed explanations for each component.

B. Data Denoising

As mentioned above, BS noise is commonly present in the sequential trajectories of users, which can obscure the underlying movement patterns of the users. To address this issue, we denoize data by eliminating remote BS using the DBSCAN algorithm, which is a clustering algorithm known for its effectiveness in outlier detection [28]. Based on the insights obtained from the data in Fig. 7, where more than 80% of the associated distances are less than 1,000 meters, we empirically set this distance as the threshold. This allows us to cluster the associated BS into two sets: close BS that are in proximity to the user's actual geographical locations and remote BS that are considered as noise because they are far away from the user's routes. It is important to note that the association durations with remote BS are short and account for only a small proportion of the total sequential BS trajectories. Therefore, filtering out these remote BS noises does not affect the characterization of the user's movement routes. The workflow of the data denoising algorithm is summarized in Algorithm 1, where ϵ represents the radius of the neighborhood and $MinPts$ is the threshold for the number of core points. The selection of $MinPts$ is determined by the density of the BS deployment, with a larger value chosen for denser deployments. In this paper, we set the value of $MinPts$ as 5 based on data-driven analysis and experiments. The value of $MinPts$ should be flexibly adjusted based on the distribution of the spatial BS, to ensure the effectiveness of data denoising.

It is worth noting that the data filtering process is an optional component, and that operators have the flexibility to decide whether to employ it for remote BS detection. Since the data format of our collected data is consistent with that of operator-collected dataset. The data denoising step can seamlessly extend to operator-collected datasets as well. Specifically, in cases where remote BS noise affects the characterization of user trajectories, operators may opt to utilize the filtering process. Conversely, operators may choose not to use this process in situations where user trajectories can be identified without any interference from remote BS noise.

C. Spatio-Temporal Filter

To identify mobile contacts on a large scale, a commonly used approach involves making $\frac{n(n-1)}{2}$ comparisons to find pairs of trajectories among n users. This means that the time complexity is $O(n^2)$, which indicates that the computational workload increases exponentially as the number of users in the system grows. To solve this problem, we suggest using an ST-Filter. This filter identifies pairs of user trajectories that have high spatio-temporal similarity and directs them to the final detection network. As a result, we can significantly reduce the computation time required to detect mobile contacts in a large-scale scenario.

Algorithm 1: Data Denoising.

Input: Sequential BS trajectories S of all users.
Neighborhood parameters (ϵ, $MinPts$).
Output: Sequential BS trajectories $\hat{S}$ of all users.
1: Initialize $\hat{S}$ to be $\emptyset$.
2: **for** $i = 1, 2, \ldots, |N|$ **do**
3: Calculate the cluster C=DBSCAN((S_i), $(\epsilon, MinPts)$).
4: **for** $j = 1, 2, \ldots, |C|$ **do**
5: **if** C_j is the cluster that contains core object. **then**
6: $\hat{S}_i = \hat{S}_i \cap C_j$
7: **end if**
8: **end for**
9: **end for**

To begin with, we extract pairs of user sub-trajectories that have overlapping periods. We discard trajectory pairs if the duration of their overlap is less than five minutes. This is because we assume that mobile contact events typically last more than five minutes and we do not consider shorter events in our system. Next, we propose a spatial fuzzy matching method for spatial filtering. Instead of directly matching BS identifiers, we calculate the spatial similarity based on the number of similar pairs of BS using the longest common sub-sequence (LCSS) algorithm [29]. Through data-driven analysis, we determine a distance threshold of 1,500 meters for the LCSS algorithm. If two BS are within this predefined distance threshold, they are considered to be a pair of similar BS. Let $A = \{a_1, \ldots, a_i, \ldots, a_n\}$ and $B = \{b_1, \ldots, b_j, \ldots, b_m\}$ represent the trajectories of users A and B during overlap time periods. Here, a_i and b_j denote the GPS locations of the BS at the indexes i and j, respectively. Let A_i and B_j refer to the prefixes of A and B, respectively, and let $LCS(A_i, B_j)$ denote the number of longest common subsequences (LCS) between A_i and B_j. Therefore, the number of similar pairs of BS can be calculated as

$$LCS(A_i, B_j) = \begin{cases} 0, & \text{if } A = \phi \text{ or } B = \phi \\ 1 + LCS(A_{i-1}, B_{j-1}), & \text{if dist } (a_i, b_j) < \gamma \\ \max(LCS(A_{i-1}, B_j), \\ \quad LCS(A_i, B_{j-1})), & \text{otherwise;} \end{cases} \tag{1}$$

where dist(a_i, b_j) is the spherical distance between a_i and b_j, and γ is the distance threshold used to determine similar BS. The empirical value of γ is set to 1,500 meters, based on the data-driven findings shown in Fig. 7. Using (1), we can calculate the spatial similarity between the trajectories A and B, which were both collected within the same time frame.

$$simi(A, B) = \frac{LCS(A_n, B_m)}{\min(n, m)}, \tag{2}$$

Algorithm 2: ST-Filter Component.

Input: Sequential BS trajectories S of all users. The spatial similarity threshold λ.
Output: A collection of all potential user pairs S'.
1: Initialize S' to be null.
2: **for** S_A in S and S_B in S and S_A != S_B **do**
3: Calculate the overlap time period Δ_W.
4: Calculate the spatial similarity $simi(A, B)$ in the same time period Δ_W using (2).
5: **if** $\Delta_W > 5$ minutes and $simi(A, B) > \lambda$ **then**
6: Add (S_A, S_B) to S'.
7: **end if**
8: **end for**

where n and m represent the length of trajectories A and B, respectively.

Fig. 14 plots the CDFs of spatial similarity scores of all pairs of trajectories for mobile contact and non-mobile contact events. We have two main observations. First, the spatial similarity scores of mobile contact events are notably larger than those of non-mobile contact events, indicating that the spatial similarity score is an efficient indicator to differentiate two types of events roughly. Second, we can observe that all the similarity scores under mobile contact events are larger than 60%, which can be selected as the spatial similarity threshold to determine whether further detection is required for the candidate pairs of trajectories. This threshold value enables the exclusion of 90% proportions of non-mobile contact trajectories. The ST-Filter workflow is summarized in Algorithm 2.

Setting an appropriate temporal and spatial threshold is essential for accurately identifying co-traveling events, meanwhile achieving a trade-off between detection accuracy and computational efficiency. *(i) Temporal Threshold.* Shorter temporal thresholds might include transient interactions that do not truly represent meaningful contact events. For instance, vehicle encounters within one or two minutes may not reflect sustained interactions necessary for defining co-traveling behavior. Conversely, longer temporal thresholds may overlook some short-distance trips with mobile contact events, thus reducing the robustness of our findings. To balance these considerations, we set the temporal threshold to 5 minutes to filter out trajectory pairs with duration overlap below this value. This threshold effectively filters out inconsequential interactions while still capturing meaningful contact events indicative of co-traveling behavior. *(i) Spatial Threshold.* A smaller threshold might lead to false positives by misclassifying unrelated interactions as mobile contacts, increasing computational load and processing time. While a larger threshold may lead to false negatives, where actual interactions are overlooked. With these considerations and data-driven insights, we set the spatial threshold to 60% to exclude 90% of non-mobile contact trajectories.

D. Learning Network

The ST-Filter is used to obtain the potential mobile contact trajectories, which are then used by the learning network for final detection. The architecture of the learning network, as shown in Fig. 13, comprises four sub-modules: data alignment, feature encoder, spatio-temporal representation learner, and detector.

Data Alignment: Before learning the underlying trajectory information, we perform data filling and segmentation to align the sparse trajectory data and synchronize it with the granularity of the system time. Specifically, since users maintain their connection to the previous BS until switching to a new one, we customize the trajectory data at the second level and fill in the temporal-empty locations with the positions of the previous BS. Subsequently, we align the BS trajectories of all user pairs based on the timestamp and divide them into multiple sub-segments with a uniform length of ω, which represents the time granularity of system detection. This granularity can be adjusted to meet the requirements of the upper-layer application. Finally, we obtain a set of trajectory pairs that are ready to be inputted into the learning model.

Feature Encoder: The one-hot embedding encoding technique is conventionally applied to encode spatial feature vectors, where the geographical location of BS is encoded to a high-dimensional binary vector. However, the dataset used in this paper contains a substantial number of BS (3,261 in total), which can lead to sparsity issues when feeding binary vectors encoded by BS into the model directly. As a result, the model cannot learn the relationship between BS. To deal with the problem, we use the Bing Maps Tile system to embed the exact position of BS, where we first map the latitude and longitude into a grid and then encode the unique ID (quadkey) [30]. In this system, the entire world is projected onto a flat plane via the Mercator projection. The map gridding technique is not constrained by travel distance and remains applicable across different cities. To improve the performance of map retrieval and display, the plane is cut into tiles (i.e., grids) of 256×256 pixels each. We consider that latitude α and longitude β are in the WGS (World Geodetic System) 84 datum when mapping the GPS coordinates into grids. In this way, the GPS coordinate (α, β) can be converted into the Cartesian coordinate (x, y) in the spherical-projected plane, which is calculated as

$$x = \frac{\beta + 180}{360} \times 256 \times 2^l,$$
$$y = \left(\frac{1}{2} - \frac{1}{4\pi} \log \frac{1 + \sin(\alpha \times \pi/180)}{1 - \sin(\alpha \times \pi/180)}\right) \times 256 \times 2^l, \quad (3)$$

where l is the scale level of the map.

Therefore, the Cartesian coordinate (x, y) can be converted to a grid tile coordinate (X, Y) by dividing them by 256. The grid division operates similarly to a quadtree, where each grid can be identified by a quadtree key (or quadkey for short). The quadkey is a base-4 number with a length equal to the scale level of the map. For instance, at the 17th level of the map, a GPS coordinate (28.174688, 112.923602) can be mapped to the quadkey "13212022010231212". The advantage of using a quadkey is that it can maintain the proximity of coordinates within a grid. Different locations within a grid share the same quadkey, which helps alleviate the sparsity problem. In addition to spatial features, temporal features also play a crucial role in the learning model. Based on data-driven analytics, the patterns

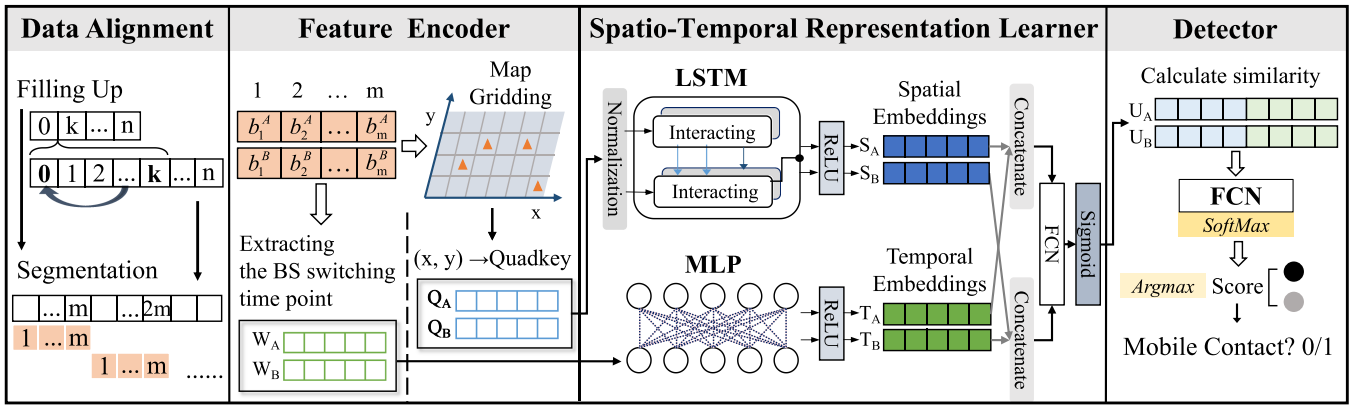


Fig. 13. Learning network framework of MoCo.

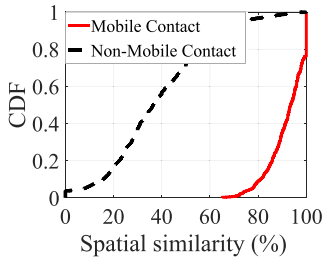


Fig. 14. CDFs of spatial similarity scores.

of BS switching differ significantly between mobile contact and non-mobile contact events. To encode the temporal sequences of BS switching, we employ the one-hot embedding approach. This approach assigns a value of "1" to switching positions and "0" to non-switching positions.

Spatio-Temporal Representation Learner: To understand the spatio-temporal representations of encoded features, we integrate two neural networks into our learning model. In particular, we use long-short-term memory (LSTM) [31] and multilayer perceptron (MLP) for spatial and temporal embedding feature learning, respectively. The reasons for choosing this architecture are listed below. First, LSTM shows good performance in sequence modeling due to its powerful ability to retain long-term dependencies and capture patterns across multiple time steps [32]. We choose LSTM to learn features from spatial BS signals due to the inherent sequential nature of the BS signals [33], [34]. Second, although the BS signals are spatial features, it is essentially also caused by the temporal BS switching sequences, with a strong temporal correlation, rather than just spatial features, so LSTM can work well. Third, recent works also demonstrate the advantages of MLP in learning time-series data due to time-step-dependent characteristics [35]. This time-step-dependent linear model, despite its simplicity, proves to be highly effective in modeling temporal patterns. Conversely, even though recurrent or attention architectures have high representational capacity, achieving time-step independence is challenging for them [36]. As shown in Fig. 13, the LSTM network is used to capture spatial correlations among candidate trajectories. The inputs to the LSTM network are the quadkey sequences of two trajectories, where each element is a quadkey tuple. The LSTM cell can be calculated at each step t as follows:

$$\begin{aligned} f_t &= \sigma(W_f x_t + W_f h_{t-1} + b_f), \\ i_t &= \sigma(W_i x_t + W_i h_{t-1} + b_i), \\ \tilde{C}_t &= \tanh(W_C x_t + W_C h_{t-1} + b_C), \\ o_t &= \sigma(W_o x_t + W_o h_{t-1} + b_o), \end{aligned} \tag{4}$$

where f_t, i_t, o_t, $\tilde{C}_t$ are forget gate, input gate, output gate, and modulated input, respectively. x_t is the spatial embedding vector. W_x and b_x are the weights and biases of each neuron, respectively. $\sigma(\cdot)$ is the Sigmoid function. Then, the memory cell C_t and hidden state h_t are updated as

$$\begin{aligned} C_t &= f_t \odot C_{t-1} + i_t \odot \tilde{C}_t, \\ h_t &= o_t + \tanh(C_t), \end{aligned} \tag{5}$$

where h_t is the output of LSTM at step t, $\odot$ is the Hadamard product operator. Here, we use a two-layer LSTM, where the outputs of each time step in the first layer are used as the inputs of the second layer. Thus, for each quadkey sequence Q_i, the LSTM cell takes the embedding of the Q_i, the memory state, and hidden state at step $i-1$ as inputs, i.e.,

$$\begin{aligned} h_i^1, C_i^1 &= LSTM(Q_i, h_{i-1}^1, C_{i-1}^1), \\ h_i^2, C_i^2 &= LSTM(h_i^1, h_{i-1}^2, C_{i-1}^2). \end{aligned} \tag{6}$$

To learn the temporal features, we utilize a one-layer MLP and input the one-hot encoding vectors of a pair of user trajectories. Consequently, the representation of the output from the MLP-based component can be expressed as follows.

$$\begin{aligned} H &= \phi(X_i W_H + b_H), \\ T_i &= W_T H + b_T, \end{aligned} \tag{7}$$

where H and ϕ represent the hidden layer and the activate function, respectively. In addition, W_i and T_i denote the input and output of the user i, respectively.

Detector: After the fully connected network (FCN) and softmax layer, the similarity scores between the user trajectories can be computed. The total score, which is the sum of the

mobile contact score and non-mobile contact score, is equal to 1. The cross-entropy loss function [37] is used to train the model. It should be noted that, for the classification problem, there is limited prior knowledge about the correlation between categories. Therefore, the cross-entropy loss function, which focuses solely on the category of the sample, is more suitable than the biased mean square error (MSE) loss function. The calculation of the cross-entropy loss function is as follows:

$$L = -\frac{1}{N}\sum_{i}[y_i * \log(p_i) + (1 - y_i) * \log(1 - p_i)], \quad (8)$$

where p_i is the predicted probability of user pair i having a mobile contact, $y_i \in \{0, 1\}$ is a mobile contact indicator, with $y_i = 1$ if the pair of users has a mobile contact, otherwise, $y_i = 0$. N represents the overall count of the training user pair samples. In the inference phase, we identify the mobile contact status based on their score values, where the highest score is chosen as the ultimate detection outcome.

VI. Performance Evaluation

A. Evaluation Methodology

Experiment Setup: To evaluate performance, we use the data set we collected, which comprises 241 user trajectories and spans a total duration of 211 hours, covering a distance of 6204 kilometers within the city. Specifically, we employ the ST-filter component to identify trajectory pairs with a high probability of mobile contact, resulting in 1,627 trajectory pairs. Subsequently, we populate the trajectory data and divide it into multiple sub-segments, each with a time granularity of 10 minutes. This process yields a total of 15,208 data samples, consisting of 6,438 labels indicating mobile contact and 8,770 labels indicating non-mobile contact. We evenly distribute the extracted samples, assigning 80% for model training and the remaining 20% for testing. The error in the testing set represents the generalization error of the final model in realistic scenarios. Importantly, the distribution of positive and negative samples in the training and testing datasets aligns with that of the complete dataset, ensuring more reliable results for assessing the model's capability.

We implement MoCo using Python 3.11.3 and PyTorch 2.0.1, an open-source machine learning framework. The experiments are carried out on a CPU server equipped with 8 cores and 16 GB of memory. The proposed model is trained for 160 epochs using a batch size of 64. We utilize the Adam optimizer with a learning rate of 0.0001.

Baselines: To demonstrate the superiority of the proposed MoCo, we design and implement the following baselines, including traditional algorithms and deep learning models.

1) Classic Models:

- *Spatial Longest Common Sub-sequence (S-LCSS).* S-LCSS is a trajectory similarity-matching approach based on the LCSS algorithm. It calculates the trajectory similarity of mobile contact training samples, and the similarity scores achieved are typically around 88.9%. Thus, trajectory pairs with similarity scores larger than this threshold are classified as mobile contact trajectory pairs.

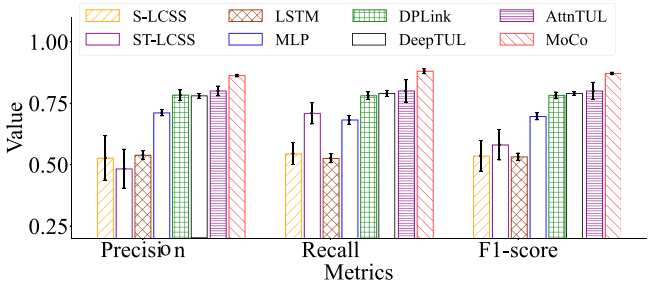


Fig. 15. Overall performance comparison.

- *Spatio-Temporal LCSS (ST-LCSS).* ST-LCSS is a classification algorithm that considers both spatial and temporal thresholds. The spatial threshold is derived from the S-LCSS, while the temporal threshold is determined by our observation that mobile contact users usually connect to BS located in a similar area within 93 seconds. Trajectory pairs that meet the spatial similarity and temporal threshold are considered mobile contact trajectory pairs.

2) Deep Learning Models:

- *LSTM* [38]. It is a variant of the recurrent neural network widely used to capture long- and short-term temporal correlations.
- *MLP*. MLP is a basic class of feedforward neural networks, and we use a hidden layer for this baseline.
- *DPLink* [39]. It is an end-to-end deep learning-based framework that aims to match the user identity from heterogeneous mobility data.
- *DeepTUL* [40]. An attentive recurrent network model is used to solve the trajectory-user link task. It combines multiple features and learns the multi-periodic nature of user mobility from trajectories.
- *AttnTUL* [41]. A hierarchical spatio-temporal attention neural network, which jointly encodes the local trajectory transitional patterns and global spatial dependencies for trajectory-user linking.

Performance Metrics: We calculate four basic conditions, i.e., true positive (TP), false positive (FP), false negative (TN), and false negative (FN), from which the subsequent three metrics are derived to assess the model's performance.

- *Precision*: Precision measures the proportion of correctly detected mobile contact trajectory pairs of all identified user pairs, that is, $\frac{TP}{TP+FP}$.
- *Recall*: Recall measures the proportion of correctly identified mobile contact trajectory pairs from the entire set of ground-truth mobile contact pairs, i.e., $\frac{TP}{TP+FN}$.
- *F1-Score*: The F1-score reconciles the precision and recall metrics, which can be calculated by $2 \times \frac{Precison \times Recall}{Precision + Recall}$.

B. Overall Performance Comparison

We conduct comprehensive comparisons of the overall performance between MoCo and other baselines. Fig. 15 presents the metric scores obtained by different models. The test samples are divided equally into 10 groups, and error bars are calculated to

ensure statistical reliability. We have the following two main observations. First, MoCo consistently outperforms other baselines significantly across all metrics. For example, the average F1-scores for MoCo, AttnTUL, DeepTUL, DPLink, MLP, LSTM, ST-LCSS, and S-LCSS are approximately 0.87, 0.80, 0.79, 0.78, 0.69, 0.53, 0.58, and 0.53, respectively. This indicates that MoCo can improve performance by 8.04%, 10.12%, 12.98%, 15.94%, 64.15%, 50%, and 64.15% compared to the respective baselines.

We analyze the limitations of different models as follows. S-LCSS exhibits the lowest precision scores since it relies on a simple spatial threshold and does not consider temporal dynamics to detect mobile contact events. While ST-LCSS incorporates temporal information, it does not fully leverage the potential of machine learning algorithms, thereby lacking in-depth feature learning capabilities. Although LSTM and MLP have deep learning structures to handle spatial and temporal dynamics, respectively, their independent focus on either spatial or temporal aspects often hinders accurate modeling of complex spatio-temporal interactions between users. DPLink uses trajectory features for user linkage but depends on predefined features, which limits its adaptability in detecting mobile contact under varying real-world conditions. DeepTUL incorporates an attention mechanism to handle the periodic nature of mobility better, its performance can degrade in scenarios with sparse data or irregular user movement patterns in diverse environments. AttnTUL builds over a graph neural network (GNN) to achieve trajectory-user linking. While GNN is powerful in handling structured data like graphs, the sparse and intermittent nature of BS trajectory data poses challenges. The BS trajectory data is less structured and discontinuous compared to the dense, well-defined graph connections. As a result, graph models may struggle to capture meaningful contact events from sparse data.

The advantages of MoCo are summarized as follows: MoCo is specifically designed to handle sparse BS trajectory data, which makes it highly effective in environments where user mobility data is sparse, such as in mobile contact detection for co-traveling in vehicles. Moreover, with adaptive spatio-temporal learning capabilities, MoCo excels in capturing diverse mobility patterns and urban environments.

C. Performance Under Different Road Types

Next, we study the performance of all models on various road types, namely urban, highway, and suburban. To facilitate understanding, Fig. 16 visualizes the mobility route and BS association example of users under different road types within a city, where various points of interest are distributed. The black line, red circle, and pink circle denote the route, associated BS, and actual BS deployment, respectively. We can observe that the BS distribution is uneven in different road types, thereby leading to irregular BS switches under such a complex road environment. Particularly, the switching frequency in urban areas is more frequent than that in suburban and highway.

As shown in Fig. 17, the proposed MoCo outperforms all baselines with a significant margin across different road types. For example, in the context of the highway environment, the average F1-Scores for AttnTUL, DeepTUL, DPLink, MLP, LSTM,

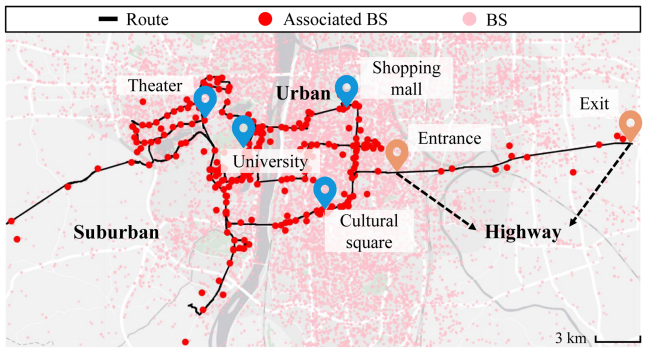


Fig. 16. Mobility route and BS association example of users in different road types, including urban, suburban, and highway.

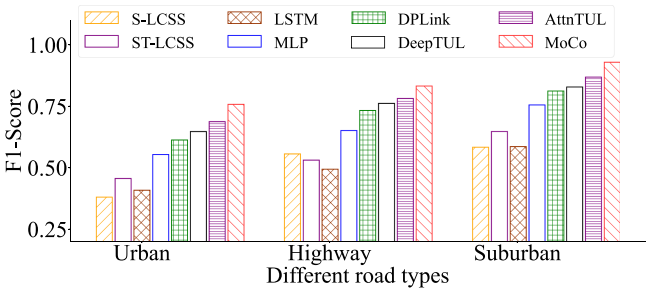


Fig. 17. Performance under different road types.

ST-LCSS, and S-LCSS are approximately 0.77, 0.76, 0.74, 0.65, 0.49, 0.56, and 0.53, respectively, while MoCo achieves a score of 0.83. Moreover, for all models, the best performance can be achieved in the suburban areas while urban areas yield the worst results. The reason lies in that the BS density in urban areas is larger than that in highway and suburban environments, which can result in more BS switching cases. As a result, there is a greater disparity in the associated BS for mobile contact users, while non-mobile contact users are closer geographically due to slow traffic, resulting in more similar associated BS. This blurs the distinction between mobile contact and non-mobile contact events, leading to a degradation in detection performance. However, MoCo demonstrates significantly better performance than other baselines across all types of roads.

We further provide some insights into specific challenges of different road types as well as how MoCo effectively addresses these variations. i) *Urban:* Urban areas often present challenges such as high population density, intricate road layouts, and fluctuations in mobility velocity caused by traffic congestion. MoCo demonstrates its effectiveness by utilizing a spatio-temporal filtering module to efficiently eliminate noise and irrelevant mobile contact traces. ii) *Highway.* Highways are generally characterized by high velocities and extended, uninterrupted roadways. The effective data denoising techniques and feature alignment of MoCo facilitate precise detection of user interactions even in fast-moving highway scenarios. iii) *Suburban:* Suburban regions typically have a lower population density. The design of spatio-temporal filtering and feature encoding in MoCo can effectively capture mobile contact events in the sparsely populated suburban.

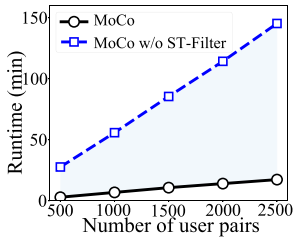


Fig. 18. Effectiveness of the ST-Filter.

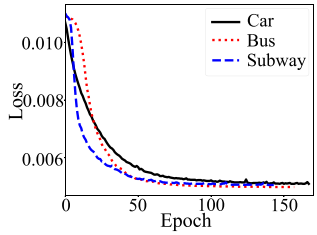


Fig. 19. Loss curves under different transportation modes.

TABLE I
ABLATION EXPERIMENT RESULTS

Methods	Precision	Recall	F1-Score
MoCo	0.8624	0.8804	0.8713
w/o LSTM	0.7742	0.8121	0.7927
w/o MLP	0.8371	0.8059	0.8212

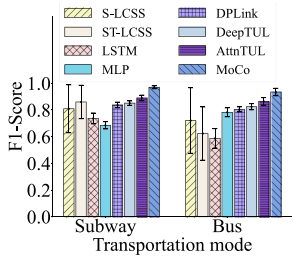


Fig. 20. F1-Score under different transportation modes.

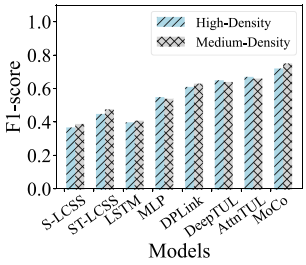


Fig. 21. Results under different density of areas.

D. Ablation Experiment

In this subsection, we perform several ablation experiments to validate the efficacy of each component in MoCo. We first evaluate the effectiveness of *ST-Filter*, and Fig. 18 shows the running time with different data sizes of user pairs. We can observe that *ST-Filter* significantly reduces running time regardless of the data size increment. Specifically, the running time of MoCo with *ST-Filter* is about 10.8 minutes with 1,500 user pairs, which is 85.6 minutes for MoCo without *ST-Filter*. Furthermore, the gap becomes more obvious as the number of user pairs increases, demonstrating the stability of MoCo.

The ablation experimental results of all metrics after removing the LSTM and MLP components are presented in Table I. It shows that the detection performance decreases rapidly after removing these components. Specifically, the F1-Score decreases from 0.8713 to 0.7927 and 0.8212 after removing the LSTM and MLP components, respectively. Moreover, when the LSTM component for spatial feature extraction is absent, the model's performance decreases more significantly, as indicated by these performance metrics. This suggests that spatial differences can better capture mobile contact behaviors than temporal information. The ablation experiments demonstrate that spatial and temporal feature learning components have significant impacts on mobile contact detection.

E. Robustness Analysis

1) Performance Under Different Transportation Mode: We then investigate the robustness of the MoCo model when users travel with other transportation modes, such as the subway and bus. We collect two additional datasets specifically for subways and buses to evaluate their performance in these scenarios. The bus dataset consists of 874 trajectory pairs, which last 13.3 hours, while the subway dataset includes 2,303 trajectory pairs, lasting 45.6 hours. The same experimental setup as before is used for these datasets. Fig. 19 illustrates the loss curves for three transportation modes, the convergence trend of bus and subway results is similar to that of the car. Fig. 20 further illustrates the average F1-Score achieved by different models for the two transportation modes. Similar trends and observations can be made as in the previous experiments. In particular, MoCo demonstrates superior performance compared to other baselines, indicating a clear advantage. It is worth noting that the performance of MoCo exhibits improvement when applied to the subway and bus dataset, with the F1-score increasing from 0.87 to 0.96 and 0.93, respectively. This enhancement is expected due to the regular schedules typically followed by subways and buses, resulting in longer time intervals between non-mobile contact trajectories (e.g., the time gap can range from 2 to 8 minutes in our datasets).

2) Performance Under Different Area Density: To further evaluate MoCo's performance in areas with varying densities of users and BS, we categorize the urban dataset into high-density and medium-density areas. High-density areas include regions like shopping malls and cultural squares, with high concentrations of users and densely deployed BS. Medium-density areas such as universities and theaters, exhibit moderate user and BS densities. User mobility in these areas is less congested compared to high-density areas. Fig. 21 shows the F1-score results of MoCo under these two types of areas. MoCo consistently outperforms other baselines in both high-density and medium-density areas. This highlights MoCo's robustness and scalability in larger urban environments.

TABLE II
PERFORMANCE UNDER DIFFERENT TIME GRANULARITIES

Methods	Recall					F1-Score				
	ω=5	ω=10	ω=15	ω=20	ω=25	ω=5	ω=10	ω=15	ω=20	ω=25
S-LCSS	0.5608	0.5119	0.5451	0.5667	0.5495	0.5319	0.5075	0.5361	0.5558	0.5279
ST-LCSS	0.6231	0.6833	0.7359	0.7402	0.7023	0.6050	0.5441	0.5911	0.5984	0.5639
LSTM [38]	0.4712	0.4362	0.5264	0.4901	0.5983	0.4933	0.4201	0.5328	0.5024	0.6032
MLP	0.5586	0.7158	0.6825	0.7106	0.7559	0.5504	0.7104	0.6965	0.7114	0.7574
DPLink [39]	0.6223	0.7018	0.7808	0.7902	0.8029	0.6322	0.7383	0.7820	0.7850	0.8119
DeepTUL [40]	0.6021	0.7233	0.7952	0.8124	0.8257	0.6201	0.7323	0.7943	0.8052	0.8324
AttnTUL [41]	0.6478	0.7306	0.8011	0.8375	0.8526	0.6531	0.7522	0.8025	0.8111	0.8537
MoCo	**0.7015**	**0.7766**	**0.8804**	**0.9010**	**0.9128**	**0.7091**	**0.8036**	**0.8713**	**0.8730**	**0.9160**

F. Impact of Time Granularity ω

In this subsection, we assess the impact of time granularity ω on the performance of mobile contact detection. The results of Recall and F1-Score achieved by different schemes with varying time granularity ω are presented in Table II. We observe that MoCo consistently outperforms the other baselines with different values of ω. Moreover, most schemes exhibit improved performance as ω increases. This can be explained by the fact that a larger ω enables the model to capture more spatio-temporal features within the specified time window, thereby enhancing the efficacy of the detection process.

The scalability of MoCo with different ω values offers opportunities for optimizing real-time mobile contact detection. By adjusting ω based on specific use cases and computational constraints, MoCo can strike a balance between detection accuracy and computational efficiency. For scenarios prioritizing real-time responsiveness, smaller ω values can be used to reduce computational overhead without significantly comprising detection accuracy. Conversely, for applications requiring fine-grained temporal analysis, larger ω values can be employed to capture fine-grained events with higher precision.

VII. DISCUSSION

A. Data Collection Impacting Factors

We discuss some factors that can arise during data collection and their impact on model performance.

- *Road Condition*. During data collection, if users are stuck in congestion, MoCo can still determine whether mobile users are traveling together based on the distance between users and the BS, as well as the time interval between connections to the BS.
- *Signal Strength*. In MoCo, GPS signals are collected as ground truth for training purposes. Therefore, in GPS-denied environments, the performance of MoCo remains unaffected as it is designed to exclusively rely on signals from BS. However, in weak BS signal environments, where the BS signal may be too weak to be reliably detected or captured by the data collection system, there is a risk of missing or incomplete data records that can potentially affect detection performance.
- *Transportation Mode*. Essentially, MoCo is versatile and can be applied to different transportation modes, including low-speed mobility scenarios involving motorcycles and electric bicycles. In addition, as high-speed train settings are characterized by fixed driving tracks and large time gaps between trains, MoCo can exhibit superior performance in accurately identifying co-traveling users.
- *Device Restriction*. Current restrictions on background applications of mobile devices running Android or iOS primarily include limitations on background services, broadcast activities, and network bandwidth. Specifically, device constraints on background services and broadcast activities do not greatly impact the essential communication process. Operators can still collect BS signals to ensure that the quality of MoCo remains unaffected. However, if the available bandwidth is restricted, data transmission between sampled data from BS to the back-end servers may become slow or intermittent, thereby impacting the overall sampling process.
- *Battery Consumption*. In MoCo, the battery consumption of mobile devices mainly depends on user preference for device setting, BS deployment density, and mobility mode. User device settings such as high screen brightness and frequent network connection result in more battery consumption. While dense urban environments might have more BS to improve signal strength and reduce energy-intensive handovers, network congestion, and signal interference may increase energy consumption. High mobility and frequent BS handovers will increase battery consumption due to the energy required for maintaining a stable connection.

B. Limitations

To the best of our knowledge, there is currently no comparable dataset available that includes ground truths for mobile contact detection, necessitating the collection of our dataset. We conduct a thorough data collection campaign to gather an adequate amount of cellular signaling traces. However, manually collected data are still limited compared to the extensive coverage and various travel scenarios present in urban environments. We have considered using the cellular operator's signaling traces to obtain a larger amount of data. A significant challenge arises as the operator data does not include the ground truth for mobile and non-mobile contacts. This means that ground-truth data have to be collected manually. Therefore, we have developed a data collection platform to collect research data, which allows us to circumvent the constraints imposed by the operator's data system.

We have discussed with network operator engineers about the data collection campaign to ensure that the data format in

our platform aligns with that of the cellular operator system to guarantee the applicability of our methodology. We exclusively utilize the signaling trace, which has a granularity that allows data collection across all cellular networks. Recognizing potential limitations in our manually collected data scale due to cost and time constraints, we have implemented several measures to mitigate the impact of the limited data scale. First, we consider multiple transportation modes, various road structures, and different BS densities. Second, our data collection efforts span various periods. Thus, while still limited in scale, our collected data captures a wide range of scenarios and provides a diverse representation of real-world situations for effective evaluation of mobile contact detection.

However, MoCo may have potential limitations in generalizing to all urban mobility patterns and environmental conditions. First, the current dataset may not fully capture areas with extreme user densities, such as large-scale public gathering events or sparsely populated regions, which could impact MoCo's accuracy and adaptability. Second, certain urban environments may have signal obstructions that were not fully represented in the collected data, potentially affecting MoCo's performance under such untested conditions. To address these limitations, we plan to collaborate with network operators to fully evaluate our model using the extensive volume of privacy-free cellular association traces. This collaboration will facilitate further optimization of our model to enhance its generalization and suitability for real-world deployment.

C. Cross-Operator Detection

Due to the commercial issue, each country generally has different cellular operator companies that implement their own BS, and users can select different operator services according to package benefits. Therefore, it is necessary to detect mobile contact events when the user signaling traces are from different operator data systems. In this paper, to achieve convincing data analysis results, we have collected data with SIM cards from the two largest operators in our country. For performance evaluation, to examine the adaptability of MoCo, we do not differentiate the operator traces, and the results are calculated based on the mixed traces. The result demonstrates that when the operator system data is available, our proposed MoCo can work with satisfactory detection performance. However, due to privacy or commercial concerns, cellular operators are often reluctant to share data.

To this end, federated learning can be adopted to empower data infusion in this case without exchanging raw data [42]. To achieve this goal, the federated learning framework should be designed to enable secure data infusion in the components of the spatio-temporal filter and detection learning network. Furthermore, parameter aggregation algorithms in federated learning should be carefully investigated to avoid degradation of detection accuracy compared to raw data exchange results, which remains an open problem.

VIII. Related Work

In this section, we review the related work in three categories, i.e., human contact detection and tracing, mobile sensing with wireless networks, and trajectory mining.

A. Human Contact Detection and Tracing

Detecting and tracing human contact or interaction is important for a variety of services, such as public security and pandemic-related services [1], [5], [7], [11], [43], [44]. Wang et al. [7] proposed a human-to-human interaction detection dataset and benchmark by leveraging a Transformer-based model. Lim et al. [8] designed an efficient and reliable human-object interaction detection network. Wang et al. [43] proposed a consistency-sensitive graph network to understand human interactions. These existing works mainly focus on human interaction or contact detection from video frames. However, these techniques cannot work effectively to detect mobile contact behaviors between users due to the limited availability of video surveillance and the challenge of detecting mobile contact with limited video frames.

For contact tracing, Yi et al. [11] proposed a COVID-19 contact tracing system based on measurements of the state information of the cellular network channel. Zhang et al. [1] proposed a healthcare worker contact tracing method utilizing low-energy Bluetooth beacons. The authors of [44] discussed some privacy issues with a mobile phone app, which is designed to help health officials track exposures after an infected individual is identified. However, these works focus on small-scale contact tracing and have privacy concerns, with difficulties in large-scale urban user mobile contact detection.

B. Mobile Sensing With Wireless Networks

In the literature, some studies have focused on wireless sensing, for example, using cellular signal data [13], [15], [16], [17], [18], WiFi signals data [19], [20], [21], [22], [23], [45] and mmWave data [24], [25]. For individual user sensing, Qi et al. [13] proposed a user trajectory tracking system using LTE signaling data, which integrates filtering techniques and a map matching algorithm to locate a user. Zhao et al. [14] proposed a framework to infer user transport modes in terms of public transportation or private cars on an urban scale with cellular data. Fang et al. [15] proposed a human mobility sensing system to recover mobility traces from the scarce cellular billing data. To identify the most likely route given a sequence of cell towers, Shen et al. [16] presented a DL-based map-matching framework.

For urban scale detection, the authors [17] designed a framework to delineate trade areas for urban commercial districts with cellular networks. Wang et al. [18] used mobile crowd sensing to build the city-wide mobile network signal map. In addition, many existing works focus on mobile sensing with WiFi data, such as indoor localization [19], human activities recognition [21], and object tracking [23]. However, most of these efforts focus on sensing individual user states or urban-scale attributes, while rarely considering the correlations among users, e.g., detecting the user mobile contact behaviors as studied in this paper.

C. Trajectory Mining

In recent years, some trajectory mining and similarity computation approaches have been applied to various applications. Trajectory similarity computation can be divided into two

categories, i.e., traditional distance-based methods and learning-based methods. Traditional methods include dynamic time warping (DTW), longest common subsequences (LCS) [29], edit distance with real penalty (ERP), etc. However, these methods come with several significant limitations, such as high computational complexity due to point-wise matching and inapplicable to data with low sampling rates and noise. To this end, some advanced learning-based methods have been proposed in recent years [46], [47], [48], [49], [50], [51]. For example, Zhou et al. [49] designed a graph-based residual LSTM network structure for trajectory similarity computation. Cao et al. [50] proposed a novel hash learning method to encode the trajectories into binary hash codes and compute trajectory similarities by Hamming distances. However, these works are mainly based on fine-grained GPS trajectories, which cannot be applied directly to sparse cellular trajectories.

IX. Conclusion

In this paper, we have proposed MoCo for urban user mobile contact detection of users based on cellular signaling traces. Specifically, we have integrated three major technical components in MoCo, i.e., data denoising, spatio-temporal filter, and detection learning network, to remove the noise of remote BS, eliminate unlikely mobile contact trajectories in both spatial and temporal domains, and learn spatio-temporal mobile contact behaviors for final detection, respectively. Extensive experiments on the real-world data have demonstrated the efficiency and robustness of MoCo. Our future work lies in two major folds. First, we will collect more data traces in different environments to evaluate the performance of MoCo, and leverage the insights to further optimize its performance. Second, we will cooperate with mobile operators to fit and implement MoCo on a Big Data platform in a production environment to perform specific mobile contact detection tasks.

References

[1] J. Zhang et al., "Contact tracing for healthcare workers in an intensive care unit," in *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 7, no. 3, pp. 1–23, 2023.

[2] J. Chen, H. Wu, P. Yang, F. Lyu, and X. Shen, "Cooperative edge caching with location-based and popular contents for vehicular networks," *IEEE Trans. Veh. Technol*, vol. 69, no. 9, pp. 10 291–10 305, Sep. 2020.

[3] K. K. Ho, D. K. Chiu, and K. L. Sayama, "When privacy, distrust, and misinformation cause worry about using COVID-19 contact-tracing apps," *IEEE Internet Comput.*, vol. 27, no. 2, pp. 7–12, Mar./Apr. 2023.

[4] H. Wu, M. He, X. Shen, W. Zhaung, N.-D. U. Đào, and W. Shi, "Network performance analysis of satellite-terrestrial vehicular network," *IEEE Internet Things J.*, vol. 11, no. 9, pp. 16 829–16 844, May 2024.

[5] U. Haroon et al., "A multi-stream sequence learning framework for human interaction recognition," *IEEE Trans. Human-Mach. Syst.*, vol. 52, no. 3, pp. 435–444, Jun. 2022.

[6] L. Xu et al., "Inter-X: Towards versatile human-human interaction analysis," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2024, pp. 22 260–22 271.

[7] Z. Wang, K. Ying, J. Meng, J. Ning, and C. Bai, "Human-to-human interaction detection," 2023, *arXiv:2307.00464*.

[8] J. Lim, V. M. Baskaran, J. M.-Y. Lim, K. Wong, J. See, and M. Tistarelli, "ERNet: An efficient and reliable human-object interaction detection network," *IEEE Trans. Image Process.*, vol. 32, pp. 964–979, 2023.

[9] Y. Chen, S. K. Dwivedi, M. J. Black, and D. Tzionas, "Detecting human-object contact in images," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 17 100–17 110.

[10] J. Park, J.-W. Park, and J.-S. Lee, "ViPLO: Vision transformer based pose-conditioned self-loop graph for human-object interaction detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2023, pp. 17 152–17 162.

[11] F. Yi, Y. Xie, and K. Jamieson, "Cellular-assisted, deep learning based COVID-19 contact tracing," in *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 6, no. 3, pp. 1–27, 2022.

[12] A. Aleta et al., "Modelling the impact of testing, contact tracing and household quarantine on second waves of COVID-19," *Nature Hum. Behav.*, vol. 4, no. 9, pp. 964–971, 2020.

[13] H. Qi, Y. Shen, and B. Yin, "Intelligent trajectory inference through cellular signaling data," *IEEE Trans. Cogn. Commun. Netw.*, vol. 6, no. 2, pp. 586–596, Jun. 2020.

[14] Y. Zhao, X. Wang, J. Li, D. Zhang, and Z. Yang, "CellTrans: Private car or public transportation? Infer users' main transportation modes at urban scale with cellular data," in *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 3, no. 3, pp. 1–26, 2019.

[15] Z. Fang, Y. Yang, G. Yang, Y. Xian, F. Zhang, and D. Zhang, "CellSense: Human mobility recovery via cellular network data enhancement," in *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 5, no. 3, pp. 1–22, 2021.

[16] Z. Shen, K. Yang, X. Zhao, J. Zou, W. Du, and J. Wu, "DMM: A deep reinforcement learning based map matching framework for cellular data," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 10, pp. 5120–5137, Oct. 2024.

[17] Y. Zhao, Z. Zhou, X. Wang, T. Liu, Y. Liu, and Z. Yang, "CellTradeMap: Delineating trade areas for urban commercial districts with cellular networks," in *Proc. IEEE Conf. Comput. Commun.*, 2019, pp. 937–945.

[18] H. Wang, B. Guo, S. Wang, T. He, and D. Zhang, "CSMC: Cellular signal map construction via mobile crowdsensing," in *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 5, no. 4, pp. 1–22, 2021.

[19] G. Wang, D. Zhang, T. Zhang, S. Yang, Q. Sun, and Y. Chen, "Learning domain-invariant model for WiFi-based indoor localization," *IEEE Trans. Mobile Comput.*, vol. 23, no. 12, pp. 13898–13913, Dec. 2024.

[20] C. Wu, X. Huang, J. Huang, and G. Xing, "Enabling ubiquitous WiFi sensing with beamforming reports," in *Proc. ACM SIGCOMM Conf.*, 2023, pp. 20–32.

[21] M. G. Moghaddam, A. A. N. Shirehjini, and S. Shirmohammadi, "A WiFi-based method for recognizing fine-grained multiple-subject human activities," *IEEE Trans. Instrum. Meas.*, vol. 72, 2023, Art. no. 2520313.

[22] W. Li et al., "WiFi-CSI difference paradigm: Achieving efficient doppler speed estimation for passive tracking," in *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 8, no. 2, pp. 1–29, 2024.

[23] X. Zhang, H. Xu, J. Liu, and J. Han, "TomFi: Small object tracking using commodity WiFi," *ACM Trans. Sensor Netw.*, vol. 20, no. 4, pp. 1–15, 2024.

[24] A. Blanco, P. J. Mateo, F. Gringoli, and J. Widmer, "Augmenting mmWave localization accuracy through sub-6 GHz on off-the-shelf devices," in *Proc. 20th Annu. Int. Conf. Mobile Syst. Appl. Serv.*, 2022, pp. 477–490.

[25] W. Xu et al., "Mask does not matter: Anti-spoofing face authentication using mmWave without on-site registration," in *Proc. 28th Annu. Int. Conf. Mobile Comput. Netw.*, 2022, pp. 310–323.

[26] S. Duan et al., "Distributed artificial intelligence empowered by end-edge-cloud computing: A survey," *IEEE Commun. Surveys Tut.*, vol. 25, no. 1, pp. 591–624, First Quarter 2023.

[27] Y. Song, D. Jiang, Y. Liu, Z. Qin, C. Tan, and D. Zhang, "HERMAS: A human mobility embedding framework with large-scale cellular signaling data," in *Proc. ACM Interactive Mobile Wearable Ubiquitous Technol.*, vol. 5, no. 3, pp. 1–21, 2021.

[28] M. Ester et al., "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. Knowl. Discov. Data Mining*, 1996, pp. 226–231.

[29] M. Paterson and V. Dančík, "Longest common subsequences," in *Proc. Int. Symp. Math. Found. Comput. Sci.*, Springer, 1994, pp. 127–142.

[30] Bing maps tile system, 2022. [Online]. Available: https://docs.microsoft.com/en-us/bingmaps/articles/bing-maps-tile-system

[31] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[32] J. Lin, Y. Chen, H. Zheng, M. Ding, P. Cheng, and L. Hanzo, "A data-driven base station sleeping strategy based on traffic prediction," *IEEE Trans. Netw. Sci. Eng.*, vol. 11, no. 6, pp. 5627–5643, Nov./Dec. 2024.

[33] Y. Gao, J. Chen, Z. Liu, L. Liu, and N. Hu, "Deep learning based location prediction with multiple features in communication network," in *Proc. IEEE Wireless Commun. Netw. Conf.*, 2021, pp. 1–5.
[34] S. H. A. Shah and S. Rangan, "Multi-cell multi-beam prediction using auto-encoder LSTM for mmWave systems," *IEEE Trans. Wireless Commun.*, vol. 21, no. 12, pp. 10 366–10 380, Dec. 2022.
[35] V. Ekambaram, A. Jati, N. Nguyen, P. Sinthong, and J. Kalagnanam, "TSMixer: Lightweight MLP-mixer model for multivariate time series forecasting," in *Proc. 29th ACM SIGKDD Conf. Knowl. Discov. Data Mining*, 2023, pp. 459–469.
[36] S.-A. Chen, C.-L. Li, N. Yoder, S. O. Arik, and T. Pfister, "TSMixer: An all-MLP architecture for time series forecasting," 2023, *arXiv:2303.06053*.
[37] P.-T. De Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Ann. Operations Res.*, vol. 134, no. 1, pp. 19–67, 2005.
[38] A. Graves and A. Graves, "Long short-term memory," in *Supervised Sequence Labelling With Recurrent Neural Networks*. Berlin, Germany: Springer, 2012, pp. 37–45.
[39] J. Feng et al., "User identity linkage via co-attentional neural network from heterogeneous mobility data," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 2, pp. 954–968, Feb. 2022.
[40] C. Miao, J. Wang, H. Yu, W. Zhang, and Y. Qi, "Trajectory-user linking with attentive recurrent network," in *Proc. 19th Int. Conf. Auton. Agents Multiagent Syst.*, 2020, pp. 878–886.
[41] W. Chen, C. Huang, Y. Yu, Y. Jiang, and J. Dong, "Trajectory-user linking via hierarchical spatio-temporal attention networks," *ACM Trans. Knowl. Discov. Data*, vol. 18, no. 4, pp. 1–22, 2024.
[42] H. Lu et al., "FL-AMM: Federated learning augmented map matching with heterogeneous cellular moving trajectories," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 12, pp. 3878–3892, Dec. 2023.
[43] Z. Wang, J. Meng, D. Guo, J. Zhang, J. Q. Shi, and S. Chen, "Consistency-aware graph network for human interaction understanding," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 13 369–13 378.
[44] H. Cho, D. Ippolito, and Y. W. Yu, "Contact tracing mobile apps for COVID-19: Privacy considerations and related trade-offs," 2020, *arXiv: 2003.11511*.
[45] S. Duan et al., "MOTO: Mobility-aware online task offloading with adaptive load balancing in small-cell MEC," *IEEE Trans. Mobile Comput.*, vol. 23, no. 1, pp. 645–659, Jan. 2024.
[46] H. He et al., "TraSS: Efficient trajectory similarity search based on key-value data stores," in *Proc. IEEE Int. Conf. Data Eng.*, 2022, pp. 2306–2318.
[47] Y. Zhang, Y. Li, and W. Ji, "A trajectory-based user movement pattern similarity measure for user identification," *IEEE Trans. Netw. Sci. Eng.*, vol. 10, no. 6, pp. 3834–3845, Nov./Dec. 2023.
[48] S. Duan et al., "VeLP: Vehicle loading plan learning from human behavior in nationwide logistics system," in *Proc. VLDB Endowment*, vol. 17, no. 2, pp. 241–249, 2023.
[49] S. Zhou, J. Li, H. Wang, S. Shang, and P. Han, "GRLSTM: Trajectory similarity computation with graph-based residual LSTM," in *Proc. AAAI Conf. Artif. Intell.*, 2023, pp. 4972–4980.
[50] Y. Cao, L. Li, X. Chen, X. Xu, Z. Huang, and Y. Yu, "Hypergraph hash learning for efficient trajectory similarity computation," in *Proc. 33rd ACM Int. Conf. Inf. Knowl. Manage.*, 2024, pp. 175–186.
[51] C. Wang et al., "A deep spatiotemporal trajectory representation learning framework for clustering," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 7, pp. 7687–7700, Jul. 2024.

Sijing Duan (Member, IEEE) received the PhD degree from the School of Computer Science and Engineering, Central South University, Changsha, China, in 2023. She is also a postdoctoral fellow with the Department of Computer Science and Technology, Tsinghua University, Beijing, China. Her research interests include Big Data analytics, mobile computing, and human mobility modeling.

Feng Lyu (Senior Member, IEEE) received the PhD degree from the Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai, China, in 2018. From 2018 to 2020, he was a postdoctoral fellow with the Broadband Communications Research Group, Department of Electrical and Computer Engineering, University of Waterloo, Canada. He is currently a full professor with the School of Computer Science and Engineering, Central South University, Changsha, China. His research interests include mobile networks, beyond 5G networks, Big Data measurement and application design, and cloud/edge computing. He has published more than 100 scientific articles in leading journals and top conferences, including the *Proceedings of the IEEE*, *IEEE Journal on Selected Areas in Communications*, *IEEE/ACM Transactions on Networking*, *IEEE Transactions on Mobile Computing*, *IEEE Transactions on Parallel and Distributed Systems*, and IEEE INFOCOM, VLDB, ACM SenSys, etc. He is the recipient of IEEE Technical Committee on Hyper-Intelligence (IEEE HITC) 2023 Early Career Researcher, and 2024-2025 IEEE ComSoc Distinguished Lecturer. He currently serves as associate editors of *IEEE Transactions on Cognitive Communications and Networking*, *IEEE Internet of Things Journal*, *IEEE Systems Journal*, and *Peer-to-Peer Networking and Applications*, and serves as TPC members for many international conferences.

Jing Zhang (Student Member, IEEE) received the MS degree from the School of Computer Science and Engineering, Central South University, Changsha, China, in 2023. Her research interests include mobile Big Data mining, human mobility, and intelligent applications.

Huali Lu (Student Member, IEEE) received the BSc and MSc degrees from the College of Computer Science and Electronic Engineering, Hunan University, in 2017 and 2020, respectively, and the PhD degree from the School of Computer Science and Engineering, Central South University, in 2024. She is a postdoctoral fellow with the Department of Computer Science, City University of Hong Kong. Her research interests include mobile computing, Big Data, and network security.

Peng Yang (Member, IEEE) received the BE degree in communication engineering and the PhD degree in information and communication engineering from the Huazhong University of Science and Technology (HUST), Wuhan, China, in 2013 and 2018, respectively. He was with the Department of Electrical and Computer Engineering, University of Waterloo, Canada, as a visiting PhD student from 2015 to 2017, and a postdoctoral fellow from 2018 to 2019. Since 2020, he has been an associate professor with the School of Electronic Information and Communications, HUST. His current research focuses on mobile edge computing, video analytics, and virtual reality.

Huaqing Wu (Member, IEEE) received the BE and ME degrees from the Beijing University of Posts and Telecommunications, Beijing, China, in 2014 and 2017, respectively, and the PhD degree from the University of Waterloo, Ontario, Canada, in 2021. She received the prestigious Natural Sciences and Engineering Research Council of Canada (NSERC) Postdoctoral Fellowship Award in 2021 and worked as a postdoctoral fellow with the Department of Electrical and Computer Engineering, McMaster University, from 2021 to 2022. She is currently an assistant professor with the Department of Electrical and Software Engineering, University of Calgary, Alberta, Canada. Her current research interests include B5G/6 G, space-air-ground integrated networks, Internet of vehicles, mobile/edge computing/caching, artificial intelligence (AI) for future networking. She received the N2Women: Rising Stars in Computer Networking and Communications Award in 2024. She received the Best Paper Awards at IEEE GLOBECOM 2018, Chinese Journal on Internet of Things 2020, IEEE GLOBECOM 2022, and IEEE GLOBECOM 2024. She serves as an associate editor of the *IEEE Communications Surveys & Tutorials* since 2024, the *IEEE Network* since 2023, and the *Security and Safety Journal* since 2021.

Yaoxue Zhang (Senior Member, IEEE) received the BSc degree from the Northwest Institute of Telecommunication Engineering, China, in 1982, and the PhD degree in computer networking from Tohoku University, Japan, in 1989. He is currently a professor with the Department of Computer Science and Technology, Tsinghua University, Beijing, China. He has published more than 200 papers. His research interests include computer networking, operating systems, and transparent computing. He is a fellow of the Chinese Academy of Engineering. He is the editor-in-chief of the *Chinese Journal of Electronics*.

Xuemin (Sherman) Shen (Fellow, IEEE) received the PhD degree in electrical engineering from Rutgers University, New Brunswick, NJ, USA, in 1990. He is a University professor with the Department of Electrical and Computer Engineering, University of Waterloo, Canada. His research focuses on network resource management, wireless network security, Internet of Things, 5G and beyond, and vehicular networks. He is a registered professional engineer of Ontario, Canada, an Engineering Institute of Canada fellow, a Canadian Academy of Engineering fellow, a Royal Society of Canada fellow, a Chinese Academy of Engineering foreign member, and a distinguished lecturer of the IEEE Vehicular Technology Society and Communications Society. He received "West Lake Friendship Award" from Zhejiang Province in 2023, President's Excellence in Research from the University of Waterloo in 2022, the Canadian Award for Telecommunications Research from the Canadian Society of Information Theory (CSIT) in 2021, the R.A. Fessenden Award in 2019 from IEEE, Canada, Award of Merit from the Federation of Chinese Canadian Professionals (Ontario) in 2019, James Evans Avant Garde Award in 2018 from the IEEE Vehicular Technology Society, Joseph LoCicero Award in 2015 and Education Award in 2017 from the IEEE Communications Society (ComSoc), and Technical Recognition Award from Wireless Communications Technical Committee (2019) and AHSN Technical Committee (2013). He has also received the Excellent Graduate Supervision Award in 2006 from the University of Waterloo and the Premier's Research Excellence Award (PREA) in 2003 from the Province of Ontario, Canada. He serves/served as the general chair for the 6 G Global Conference'23, and ACM Mobihoc'15, Technical Program Committee chair/co-chair for IEEE Globecom'24, 16 and 07, IEEE Infocom'14, IEEE VTC'10 Fall, and the chair for the IEEE ComSoc Technical Committee on Wireless Communications. He is the president of the IEEE ComSoc. He was the vice president for Technical & Educational Activities, vice president for Publications, member-at-large on the Board of Governors, chair of the Distinguished Lecturer Selection Committee, and member of IEEE fellow Selection Committee of the ComSoc. He served as the editor-in-chief of the *IEEE Internet of Things Journal*, *IEEE Network*, and *IET Communications*.