Crowd-Flow Graph Construction and Identification with Spatio-Temporal Signal Feature Fusion

Suining He and Kang G. Shin
Department of Electrical Engineering and Computer Science,
The University of Michigan, Ann Arbor, MI 48109-2121
Email: {suiningh,kgshin}@umich.edu

***Abstract*—To realize an efficient and flexible urban crowd-flow identification, we propose a new scheme called `CFid` by fusing fine-grained spatio-temporal smartphone signal features. Considering the abundance of ambient WiFi and geomagnetism, we design and validate several of their fine-grained features and similarity metrics to determine if two individuals belong to the same crowd-flow. We formalize a graph stream clustering problem, where co-flow user pairs are opportunistically identified and fed as a dynamic sequence of edges connecting each other. Given such a flexible form, a fast and accurate algorithm processes the edges and identifies the crowd-flows for further upper-level applications, including urban-flow monitoring and shopping-advertisement/recommendation. Using extensive data-driven analytics and 8-month experimental studies upon 50 users (over $2{,}500$ walking traces at 7 different urban sites), we have validated the accuracy (usually $>95\%$), efficiency (only $<5\%$ extra energy-footprint on average than normal usage on mobiles) and flexibility of `CFid` in identifying large-scale crowd-flows.**

I. Introduction

The advent of smart cities accompanied by increasing pervasiveness of IoTs (Internet of Things), provides unprecedented capabilities and opportunities to monitor, model and comprehend the mobility of urban crowds, benefiting both smart-city planners and residents. The resulting crowd analytics market is estimated to grow at a Compound Annual Growth Rate (CAGR) of 24.3%, from USD $\$385.1$M in 2016 to $\$1{,}142.5$M by 2021 [1]. Among the various mobility patterns explored for urban and social sensing, finding the individuals in a certain site of interest moving together on similar paths — called *crowd-flows* — is key to many emerging applications [2]–[6], including event surveillance, urban planning, social analysis, recommendation and consequent commercial promotions.

Conventional crowd-flow studies usually require location estimation and subsequent trajectory mining. Despite their success in vehicle networking or macro migration tracking (demography or zoology), few of them are applicable in crowded urban or indoor environments, where dedicated infrastructures (say, GPS, CCTV/camera or wireless probing transceivers) for localizing devices are likely non-existent or provide poor accuracy (due to crowds or other reasons). Beyond their coarse-grained estimates, a *fine-grained* or *last-mile* augmentation is needed for more pervasive deployment. Mutual proximity between users can be obtained from device pairing (say, Bluetooth), but may cause privacy risks, especially when they are discoverable by other parties.

Furthermore, urban crowd-flows are highly dynamic due to many opportunistically-encountered users. While different signal modalities and their combination have been considered in mobility analytics, few of them considered fine-grained and hybrid signal feature designs for crowd-flows, and provided spatio-temporally adaptive models for their fast identification. In particular, a *scalable* and *efficient* mechanism is required for urban or spacious indoor settings like large malls or airports.

Motivated by ambiance of urban/indoor WLAN infrastructures and geomagnetic anomalies, we propose `CFid`, a **C**rowd-**F**low **id**entification system via fine-grained spatio-temporal signal fusion, with the following design considerations:

- ⋆ By leveraging the spatial diversity (particularly along a certain walking path), we associate the WiFi and magnetic features measured from individuals' smartphones with their sequential/temporal co-presences, or *co-flow*, without explicitly calibrating, pairing devices or tracing the locations.
- ⋆ Closeness or spatio-temporal *similarities* between people (device carriers/users) can be efficiently identified by online comparison of fine-grained signal sequences between users, hence enabling fast detection without extensive localization of devices.
- ⋆ On the signal patterns derived and fine-grained similarity measures, we consider crowd-flows as the *graph stream*, where individuals as vertices in a graph are dynamically connected via correlations of their signals. The stream of the resultant edges can then be fed and processed efficiently.
- ⋆ As these signals can be inertially measured from phones (sanitized and crowdsourced to a central hub or server), we can mitigate individuals' privacy concerns by regulating pairing or communication with unknown peer devices.

This paper makes the following main contributions:

- *Extraction and Application of Spatio-Temporal Signal Features for Co-flow Detection*: Via comprehensive data analytics, we take into account the spatio-temporal signal features, and extract several critical patterns (prototyped with smartphone-based magnetic and WiFi measurements, but integrable with many others) as the basis for detection of co-flow users. To quantify the closeness between users, we have designed hybrid signal metrics fusing these heterogeneous features, which enhance accuracy and robustness under environmental interferences.
- *Efficient Crowd-Flow Identification*: After evaluating the derived features, we have designed automatic learning of decision parameters and fast co-flow detection to quickly determine co-presence of each two users and their mutual edges. Then, by formalizing *graph stream clustering*, `CFid` accurately and efficiently identifies the flow, or dense subgraph, that each individual belongs to. This way, `CFid` can

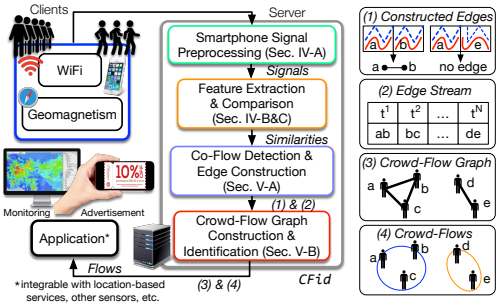

Fig. 1: System framework and information flow of CFid.

achieve flexible and efficient deployment for urban crowd-flow monitoring.

- *Extensive Data-Driven User Studies*: With the aforementioned design, we have conducted an extensive study of three real-world user datasets, including more than 2,500 sampled sequences (walking traces) collected from 50 users at 7 different urban sites during more than 8 months. Our experimental results validated CFid's accuracy, effectiveness and applicability in identifying crowd-flows (say, often with $\geq$95% accuracy and $\leq$1 second overhead), while consuming a negligible amount of energy (only $<$5% extra on average compared to normal usage) upon the mobile devices.

Fig. 1 illustrates the system framework (with information flow) of CFid, which consists of *client* and *server* sides. At the client side, a target scenario app (say, shopping mall promotion or site monitoring) is considered to run transparently in the smartphone background to collect WiFi and magnetometer signals (given user consents). The central monitor (server) then performs the *smartphone signal preprocessing*, derives the co-flow features and assesses the person-to-person similarities (*feature extraction & comparison*). During *co-flow detection & edge construction*, as the crowd-flows are dynamically evolving over time, CFid batches, processes and streams the detected person-to-person closeness as edges in (1). Given the detected co-flow users (edge stream as (2)), the server forms the streamed graph like (3), and efficiently identifies the crowd-flows (dense subgraphs in (4)) they belong to (*crowd-flow graph construction & identification*) for the app.

Note that such flow-related signals can be "crowdsourced" from many individuals [5,7,8], given the social and commercial significance, and practical user incentivization by certain commercial promotions, coupons, recommendations and other monetary/psychological benefits relevant to the sites of interest, and consequently gain the social acceptance. Albeit prototyped under WiFi and geomagnetism, CFid can be easily extended to other emerging signal modalities including channel state information (CSI) and visible light for more advanced applications [4,9,10].

This paper is organized as follows. After reviewing the related work in Sec. II, we present the important concepts, problem and test datasets in Sec. III. Then, we discuss the fine-grained and data-driven signal feature designs in Sec. IV. This is followed by our core crowd-flow identification in CFid in Sec. V. We further validate the prototype of CFid with real-world data in Sec. VI, and conclude the paper in Sec. VII.

II. Related Work

Crowd status can be obtained from pairing-based [9,11] and location-based sensing techniques [12]. Trajectory mining for group/community discovery [13] has also been studied. By deriving spatio-temporal features from inertially-measured smartphone signals, CFid is *amendable* to these studies or applications, and can serve as a plug-in (Fig. 1) for their more adaptive and pervasive deployment.

Recently, researchers started associating signal modalities to infer the mobility, social or demographic patterns of crowds [3,4]. Kjærgaard *et al.* [13] considered temporal user clustering with different sensor readings. GruMon [6] detects groups mainly based on users' temporal movement correlation. While most pilot studies focus on single correlation measure in terms of group mobility or signal modalities [6,14], we design several comprehensive metrics *jointly* on spatio-temporal features to detect crowds more effectively. Furthermore, instead of their computationally-expensive static clustering [15,16] and supervised learning [6,13] that requires *a priori* training, we design a fast graph streaming and clustering framework without extensive model or parameter calibration.

WiFi and geomagnetism sensing, due to their ubiquity, have triggered a myriad of mobile apps [10,15], including location-based service [14,17] and smartphone sensing [12]. However, few of these studies systematically investigated their *fusion* potential for crowd-flow study. To fill this gap, we propose CFid which is built on several novel signal-processing and crowd-related feature extraction techniques to unfold their potential for fast and accurate crowd-flow analytics.

III. Concepts, Problem Formulation & Data Sets

A. Important Concepts & Problem Formulation

We first briefly introduce the important concepts related to the design of CFid.

Definition 1: Crowd-Flow: In an urban/indoor environment, a *crowd-flow* consists of multiple ($\geq$ 2) pedestrians who are moving along a similar path and direction. □

To quantify and model a crowd-flow (CF), we define a measure for each pair of users in the same flow (or *co-flow*):

Definition 2: Person-to-Person Co-Flow Similarity: The co-flow similarity, $\Phi(i,j)$, captures the likelihood that two users, i and j, belong to the same CF. In other words, the higher the $\Phi(i,j)$, the more likely two users belong to the same CF. Specifically, we find the co-flow signal features (say, WiFi and geomagnetism) showing the sequential (spatio-temporal) potential of the two users in the same moving crowd. □

For ease of its formulation and description, we consider $\Phi(i,j) \in [0,1]$ for CFid. The co-flow pedestrians sharing the contexts are likely to have strong mutual similarity in measured signals and thus form a virtual *edge*, while those in different CFs possess weak or even no correlation, resulting in no edges (or edge pruning) between them. Based on this observation, we can construct:

Definition 3: Crowd-Flow Graph: Each user i is considered as a vertex u_i, while her/his detected co-flow states related to other users j's are considered as edges e_{ij}'s (if not pruned).

CFid searches the thus-formed graph $\mathcal{G} = (\mathbf{V}, \mathbf{E})$ where $\mathbf{V} = \{u_i\}$ and $\mathbf{E} = \{e_{ij}\}$ to find the CFs $\mathbf{F}$'s underneath.

Let δ_i be the degree of u_i. We assume there is no self-loop in $\mathcal{G}$, *i.e.*, a user needs no similarity comparison with himself. In the context of graph clustering [18], vertices (users) u_i's within a CF $\mathbf{F}$ (dense subgraph [18]) tend to be connected more (of higher degrees) than across different flows. □

Due to user mobility and dynamics, the formed graph in Def. 3 is dynamically changing and updated over time. Meanwhile, crowdsourced signals are usually streamed in. Thus, to balance between identification timeliness and robustness against noisy signals, we further consider the setting of:

Definition 4: Graph Streaming: Graph streaming considers a stream (or batched) — an order sequence (that can be random) of identified closenesses or *edges* — as

$$\mathbf{S} = \{e^1, \ldots, e^n\}, \tag{1}$$

where $e^t \in \mathbf{E}$ $(1 \leq t \leq n)$ represents the t-th edge arriving at the central monitor of CFid. Let A_{ij} $(A_{ij} \geq 0)$ be an element of an augmented adjacency matrix [18] representing the number of edges between users i and j. Note that each edge $e_{ij} \in \mathbf{E}$ (co-flow users i and j) can be spotted A_{ij} times within the stream $\mathbf{S}$. □

To quantify the formation of CFs, we leverage the concept of *modularity* for graph partitioning (clustering) [18]:

Definition 5: Crowd-Flow Graph Modularity: In CFid, the modularity of a crowd-flow graph is defined as the fraction of the edges that fall within the given CFs $\mathbf{F}$'s minus the expected fraction if edges were distributed randomly [18]. Specifically, let Q be the *modularity* of $\mathcal{G}$, *i.e.*,

$$Q = \frac{1}{2|\mathbf{E}|} \sum\nolimits_i \sum\nolimits_j \left(A_{ij} - \frac{\delta_i \delta_j}{2|\mathbf{E}|} \right) \mathbf{I}_{ij}, \tag{2}$$

describes the *division strength* of graph partitioning [18]. $\mathbf{I}_{ij} = 1(0)$ if i and j are (not) in the same flow. □

Q can be interpreted as the probability that an edge lies within its assigned subgraph minus $\delta_i \delta_j / (2|\mathbf{E}|)$, the probability that an edge, if chosen proportionally to the vertex degrees, lies in the subgraph (a randomized version of the original $\mathcal{G}$, or *null model*) [18,19]. In general, the higher the Q, the more credible or confident that a graph partitioning has led to crowd-flows $\mathbf{F}$'s (cluster). Note that direct optimization of graph modularity is NP-hard [18], and hence various heuristics have been proposed to approximate and push the limit (e.g., see [18] for details which are omitted due to space limit).

Based on the aforementioned concepts, we present the problem formulation in CFid as:

Definition 6: Crowd-Flow Identification (CFI): Let b_i^l be the decision variable that a user i belongs to flow l, *i.e.*, $b_i^l = 1$ if $i \in l$ and $b_i^l = 0$ otherwise. From user i's smartphone sensors, we get the WiFi and magnetic readings, denoted as $[\mathbf{W}_i, \mathbf{M}_i]$. Let $f(i, j, \mathbf{W}_i, \mathbf{W}_j, \mathbf{M}_i, \mathbf{M}_j)$ be the co-flow similarity function between users i and j given $\{i, j, \mathbf{W}_i, \mathbf{W}_j, \mathbf{M}_i, \mathbf{M}_j\}$. With the batched user measurements, their mutual $\Phi(i, j)$'s and a decision function $\Gamma(\Phi(i, j), l)$, the CFI in CFid is to find the flows $\mathbf{F}$'s of users, or dense subgraphs (clusters) in $\mathcal{G}$, such that the modularity Q of the streamed graph partitioning is maximized, *i.e.*,

$$\begin{aligned} \arg\max_{\{\mathbf{F}'s\}} \quad & \text{Objective (2)}, \qquad (3) \\ \text{s.t.} \quad & \Phi(i, j) = f(i, j, \mathbf{W}_i, \mathbf{W}_j, \mathbf{M}_i, \mathbf{M}_j), \quad b_i^l \in \{0, 1\}, \\ & (b_i^l, b_j^l) = \Gamma(\Phi(i, j), l), \quad \forall i, j, \quad \sum\nolimits_l b_i^l = 1. \end{aligned}$$ □

B. Datasets for Crowd-Flow Studies

We study the CFI with the following 3 real-world datasets.

Dataset A: We recruited 50 volunteers for data collection at 7 different urban sites (including our university campus, recreation center, student dormitory apartment, academic building and premium shopping mall) during more than 8 months. A wide spectrum of smartphones, including Samsung Note 7, S6 Edge+, Note 5, Note II, LG G3, Google Pixel, Pixel XL, Nexus 6P and Nexus S, are involved. A total of 2,516 walking sequences/traces (each is of average size 55kB) have been collected for our data analysis. A total of 442 WiFi access points (APs) are detected (covering over 3,700m^2 in total; see Sec. VI-A for details).

Dataset B [20]: This is a public dataset collected in an academic building (of hall ways and corridors) with Sony Xperia M2 and LG G Watch R (W110), which contains over 648 user walking traces with WiFi/geomagnetic readings. A total of 127 APs are detected in sites covering over 180m^2.

Dataset C [21]: A public WiFi/geomagnetism dataset collected from 6 different indoor areas of an office building (including labs, offices and lounges), which contains over 461 user walking traces (details are referred to [21]). A total of 219 APs are detected in sites covering over 70m^2.

For each of the datasets, we have the timestamps, WiFi MAC addresses (basic service set identifier or BSSID), names of networks (service set identifier or SSID, if available), received signal strength indicators (RSSI; in dBm), geomagnetic readings (three dimensions; in μT) and user/crowd IDs for data analytics in the following sections.

IV. Co-Flow Feature Extraction & Similarities

To characterize $\Phi(i, j)$, the feature extraction in CFid needs to (1) define *similar patterns*, and (2) design their *similarity measures*. Given the collected smartphone signals (Sec. IV-A), Fig. 2 illustrates the co-flow features explored in CFid, where the WiFi (APs set and signal sequence in Sec. IV-B) and geomagnetic features (spectral/temporal, as in Sec. IV-C) are characterized and compared. Finally, we present hybrid similarity and feature refinement (Sec. IV-D).

A. Smartphone Signal Collection & Preprocessing

Via the existing mobile APIs (e.g., Android), each WiFi-tuple measurement from an access point (AP) can be easily obtained, and is represented as [MAC, RSSI, SSID, Timestamp]. SSID is the service set identifier, if any, which is usually the name of WiFi network shown in the user interface (like the well-known "eduroam"). For each user i, let $W_i^k(t)$ be the RSSI from an AP indexed by k at time t. Mobile APs tethered by third parties are filtered out based on the vendor-dependent organizationally unique identifiers (OUIs) inside their MAC addresses. RSSI values are converted from dBm to mW to differentiate strong and weak signals.

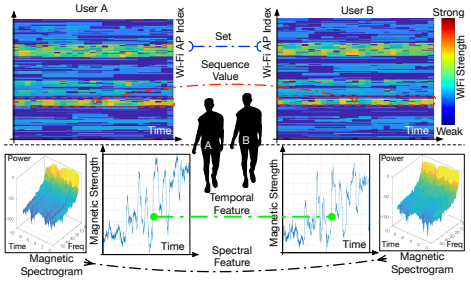

Fig. 2: Spatio-temporal co-flow features (WiFi heatmaps, magnetic spectrograms & sequences).

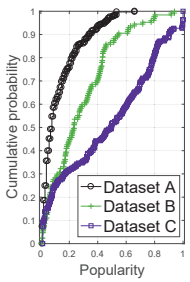

Fig. 3: AP popularity in our datasets.

To measure the magnetic signal, for Android platforms we utilize *calibrated magnetic field* (`TYPE_MAGNETIC_FIELD`), as the hard iron bias [12] is removed from the given sensor readings (self-calibration due to distortions that arise from magnetized metal or permanent magnets on the device). Then, a temporal sequence of size ω (a predefined sliding window) is defined as $\mathbf{M}_i = [M(1), M(2), \ldots, M(\omega)]$. We *normalize* the readings to cope with device heterogeneity, apply a 5-order Butterworth low-pass filter upon them, and focus on the signals below 12Hz as they are more likely associated with normal human movements (including walking and turning) [12].

B. WiFi Co-Flow Feature Extraction & Comparison

To illustrate similar WiFi patterns of two co-flow users, we plot in Fig. 2 their sequentially-scanned heatmaps (with spatio-temporally alike strengths within a 10s window) *w.r.t.* all APs (vertical axis) along the same path (horizontal axis). Such correlations can be leveraged to detect the co-presence. So, co-flow feature extraction with WiFi is mainly based on comparison in *AP sets* (*spatial*) and *AP signal sequences* (*temporal*). Due to a certain coverage area of the shared APs, set comparison can semantically characterize the co-presence of users A and B, while signal sequence comparison upon these shared APs further differentiates them. Fig. 2 illustrates these two perspectives, where the spatio-temporal WiFi diagrams of two users are compared. Given a sliding window T of latest WiFi readings, we will detail these two design perspectives.

(a) **Semantic AP Set Comparison**: In terms of detected AP sets (union of APs within the sliding window T), we design the following spatial metrics assessing two users' closeness:

- *Adapted Tanimoto Similarity*: For each user i, we consider a bit vector $\mathbf{I}_i$ representing a set of all candidate APs (union of all available MACs), where each element or bit $\mathbf{I}_i[k]$ is 1 if s/he detects this AP k, and 0 otherwise. Then, the adapted *Tanimoto* similarity [22] between them, $\psi^T(i,j)$, is
$$\psi^T(i,j) \triangleq \frac{1 + \mathbf{I}_i \cdot \mathbf{I}_j}{1 + |\mathbf{I}_i|^2 + |\mathbf{I}_j|^2 - \mathbf{I}_i \cdot \mathbf{I}_j}. \quad (4)$$
In other words, the more APs of the detected are shared by two users, the higher $\psi^T(i,j)$ and the more likely they belong to the same crowd-flow. Note that such an adaptation is to ensure $\psi^T(i,j) \in (0,1]$.
- *Adapted Adamic-Adar Similarity*: Apart from the ratio of shared APs, our adapted Adamic-Adar similarity [23] evaluates the closeness in terms of shared APs from another vantage, where the co-presence of more unpopular APs is more likely to indicate the closeness of two users. Popular APs, on the other hand, can be less informative in set comparison due to their large coverage.
Specifically, let P^k be the popularity of AP k ($P^k \in (0,1]$), *i.e.*, ratio or proportion of those detecting AP k within all the involved users. Formally, the adapted Adamic-Adar similarity $\psi^A(i,j)$ ($\psi^A(i,j) \in (0,1)$) for all shared APs that are detected by both users i and j is given by
$$\psi^A(i,j) \triangleq \sum_{k=1}^{|\mathbf{I}_i \cap \mathbf{I}_j|} \frac{1}{1 + \log(1 + P^k)}, \quad (5)$$
where $|\mathbf{I}_i \cap \mathbf{I}_j|$ is the set cardinality of their shared APs.

From the analysis of our datasets (CDFs of P^k's in Fig. 3), we have observed in Datasets A and B that more than 80% of the detected APs have popularities P^k's lower than 45%, while in Dataset C P^k's are higher due to denser APs, which may account in part for performance differences (Sec. VI).

(b) **Efficient AP Signal Sequence Comparison**: Beyond the detected AP set, we also consider AP signal sequence comparison to account for the users' dynamic patterns, thus incorporating user co-flow mobility in practice.

To facilitate the WiFi sequence comparison, we leverage the LB_Keogh bound (*LB_K*) [24], which efficiently returns a *lower-bound* of the dynamic time warping (DTW) distance between two temporal sequences in linear time (illustrated in Algo. 1). Specifically, for each user i, let $\mathbf{W}_i^k = [W_i^k(t-T), \ldots, W_i^k(t)]$ be her/his measured WiFi sequence from AP k within a sliding window of T. When comparing two given sequences $\mathbf{W}_i^k$ and $\mathbf{W}_j^k$ of users i and j from AP k (Line 1), *LB_K* takes in the global path constraints, *i.e.*,
$$id_j - p \le id_i \le id_j + p, \quad p > 0, \quad (6)$$
where id_i and id_j are indices of the warping path *w.r.t.* two series, and p is a predefined path constraint. Then, *LB_K* finds the upper and lower values along the path, u and l (Lines 6 to 7), for the difference accumulation, and a lower-bounding measure lb_sum of difference is later returned (Lines 8 to 12).

Finally, we define if *LB_K* distance, $LB\_K\left(\mathbf{W}_i^k, \mathbf{W}_j^k\right)$, divided by the mean length of the two sequences (Line 15), is less than a positive threshold γ, or equivalently,
$$LB\_K\left(\mathbf{W}_i^k, \mathbf{W}_j^k\right) \le \frac{1}{2}\left(|\mathbf{W}_i^k| + |\mathbf{W}_j^k|\right) \cdot \gamma, \quad \gamma > 0, \quad (7)$$

Algorithm 1: Fast WiFi sequence comparison.

Input: $\mathbf{W}_i$, $\mathbf{W}_j$: two sequences; γ: threshold; p: window size.
Output: *LB_K* difference divided by the mean length.

```
1  lb_sum←0; W_i←shorter(W_i, W_j); W_j←longer(W_i, W_j);
       /* Enumerate shorter W_i for efficiency */
2  for id, e in enumerate(W_i) do
3      if id > len(W_j) then
4          break;
5      end
       /* Upper & lower values: u & l */
6      u←max(W_j[(id-p if id-p ≥ 0 else 0): (id+p)]);
7      l←min(W_j[(id-p if id-p ≥ 0 else 0): (id+p)]);
8      if e > u then
9          lb_sum←lb_sum + (e-u)^2;
10         else if e < l then
11             lb_sum←lb_sum + (e-l)^2;
12         end
13     end
14 end
15 return sqrt(lb_sum)/mean([len(W_i),len(W_j)]);
```

then we conclude that the input pair may be similar (we empirically find optimal γ as 0.002mW in our experiment), or dissimilar otherwise. Note that complexity of computing *LB_K* is linear with the input WiFi sequence length [24]. Via this fast regional decision, we can quickly prune some unnecessary computation for the later magnetic field (Sec. IV-C) when two users are not likely co-present.

If the average *LB_K* is sufficiently small, we further form a metric and find the similarity between WiFi sequences of users i and j, denoted as $\psi^K(i,j)$ ($\psi^K(i,j) \in (0,1)$), *i.e.*,

$$\psi^K(i,j) \triangleq \frac{1}{1+\log\left(1+\sum_k \textit{LB\_K}\left(\mathbf{W}_i^k, \mathbf{W}_j^k\right)\right)}. \quad (8)$$

We select the three strongest APs (in terms of RSSI) which are shared by two users for the above comparison, as they are more likely to be detected and lead to longer measurement sequences for better differentiation.

Finally, given the above components of Eqs. (4), (5) and (8), the hybrid similarity $\Phi^W(i,j)$ between users i and j in WiFi signal space is defined as

$$\Phi^W(i,j) \triangleq \psi^T(i,j) \cdot \psi^A(i,j) \cdot \psi^K(i,j). \quad (9)$$

For ease of interpretation and integration, we normalize all $\Phi^W(i,j)$'s into the range $\Phi^W(i,j) \in (0,1)$.

C. Geomagnetic Co-Flow Feature Extraction & Comparison

Besides WLAN, urban/indoor sites are usually cluttered with diverse geomagnetic anomalies due to the steel-reinforced buildings. As in Fig. 2, when users A and B move along the same path with magnetometers on their phones, similar magnetic sequences are captured, implying the crowds' spatio-temporal co-presence.

Instead of focusing on individual magnetic reading, `CFid` takes in much less ambiguous patterns based on spatio-temporally measured sequence. Further, fusing geomagnetism's profound mobility-related features with the WiFi's *regional* indication enhances `CFid`'s fine-grained accuracy, and robustness against external and environmental interferences (validated in Sec. VI). In what follows, we present the spectral and temporal features considered in `CFid`.

(a) **Spectral features**: Fig. 2 shows similar spectrograms of magnetic readings when users A and B are walking in the same flow. Due to their similar step speed, human body movement and turning on the same walking path, the geomagnetism may show similar profound mobility features.

To quantify such a subtle spectral similarity, we perform the Fast Fourier Transform over the magnetic sequence (we empirically set a window of $\omega = 5$s), and extract a total of 11 spectral features s^n (indexed by $n \in \{1,\ldots,11\}$), *i.e.*, mean, standard deviation (STD), entropy, minimum nonzero energy, weighted average of frequency (based on energy), domain frequency ratio, coefficient of variation, skewness, kurtosis, flatness, and spread (i.e., dispersion of spectrum around the centroid). Details of these features have been well documented in the literature [22], and are thus omitted. The thus-formed vectors ($\mathbf{S} = [s^1, s^2, \ldots, s^{11}]$) of spectral features, $\mathbf{S}_i$ and $\mathbf{S}_j$ *w.r.t.* users i and j are compared with the Euclidean distance (l^2-norm). Then, $dist^S(i,j) = \|\mathbf{S}_i - \mathbf{S}_j\|_2$ is returned.

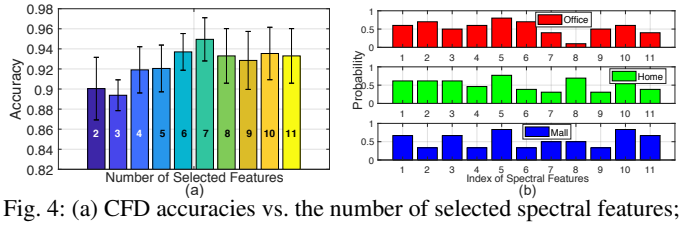

Fig. 4: (a) CFD accuracies vs. the number of selected spectral features; (b) Probability of spectral features to be selected (Dataset A).

(b) **Temporal features**: We also take into account the temporal sequence differences. Unlike applying *LB_K* bound upon the coarse-grained WiFi signals, magnetic readings of high sampling rates (often 100Hz vs. 1Hz of WiFi) and more features need a fast but fine-grained similarity measure. Specifically, we design in `CFid` a fast dynamic time warping (DTW) [22] augmented with *Z-normalization* (normalized to zero mean and unit of energy) and *early comparison abandoning*. Z-normalization makes `CFid` focus on the temporal trend instead of offsets and noises. Specifically, each reading $M(t)$ is subtracted by μ and then normalized by σ, *i.e.*,

$$\mu = \frac{1}{m}\sum_{t=1}^{m} M(t), \quad \sigma = \left(\frac{1}{m}\sum_{t=1}^{m}\left(M(t)\right)^2 - \mu^2\right)^{\frac{1}{2}}. \quad (10)$$

When comparing each pair of elements from two sequences, we track the cumulative *LB_K* lower bound [24] so far, and abandon the following comparison early if it is larger than the calculated DTW up to this moment. Further, if the final DTW result is larger than a predefined threshold (0.24 in our setting), we conclude that they are not co-flow (the edge is pruned). Hence, the efficiency of temporal comparison is significantly improved (often $20\times$ faster). Finally, the temporal DTW distance $dist^T(i,j)$, if not pruned, is returned.

D. Hybrid Similarity & Feature Refinement

In summary, given the above temporal and spectral distances, *i.e.*, $dist^S(i,j)$ and $dist^T(i,j)$, inspired by [25] we design the similarity $\Phi^G(i,j) \in (0,1)$ between users i and j in geomagnetic signal space, which is formally given by

$$\Phi^G(i,j) \triangleq \frac{1}{1+\log\left(1+dist^S(i,j)\cdot dist^T(i,j)\right)}. \quad (11)$$

Then, taking Eqs. (9) and (11) into account, the hybrid similarity $\Phi(i,j)$ between users i and j is defined as

$$\Phi(i,j) \triangleq \beta\cdot\Phi^W(i,j) + (1-\beta)\cdot\Phi^G(i,j), \quad \beta \in [0,1], \quad (12)$$

where β is a customizable parameter characterizing and balancing the weight of importance. The similarity metric $\Phi(i,j)$ in Eq. (12) enables us to include many other signals [4,9,10] if available, interpret performance, customize for and extend to other crowd-related applications.

To support efficient and large-scale deployment, one may also utilize the FEAST toolbox [26] and choose the *Joint Mutual Information* (JMI) criterion for offline selection of more useful features, reducing unnecessary online vector computation. We choose JMI for its advantages in balancing among accuracy, stability, and flexibility [26], and select features offline *w.r.t.* each application site or scenario.

Fig. 4(a) shows a case study (Dataset A) with the mean co-flow detection (CFD) accuracy vs. the number of selected geomagnetic features (with standard deviations). For each

case, we conduct 5 different test trials to find its mean and variance. When the number of features is small, accuracy is not high due to little recognizable information. As more features are selected, accuracy improves and then degrades as some noisy signals are also included. To reflect this tradeoff, we select the top 7 features ranked by JMI by default.

Fig. 4(b) shows the spectral features selected based on JMI, with the probability of each feature being selected at 3 typical sites from Dataset A. For each site, we conduct 10 permutations and feature selections to evaluate the probability of selected features. We can see that magnetic features from statistics of higher order (say, kurtosis of index 9) may vary with sites, because user activity freedom as well as the spatial magnetic diversity may be different.

V. Crowd-Flow Construction & Identification

Due to the crowd-flow dynamics in practice, we transform the identification to a *graph stream clustering* problem, which consists of *fast detection and edge construction* (Sec. V-A), and *dynamic graph streaming and clustering* (Sec. V-B).

A. Fast Co-Flow Detection & Flexible Edge Construction

A co-flow edge is formed if the similarity $\Phi(i,j)$ of two users is found to be sufficiently high. We design automatic self-learning to adaptively parameterize this decision rule, and then conduct fast co-flow edge detection accordingly.

(a) Automatic self-learning of decision parameters:

When `CFid` is initialized, *affinity propagation clustering* (APC) [27] is done upon the users, given their pairwise similarities $\Phi(i,j)$'s, to learn the decision parameters. Unlike conventional k-means clustering, the APC reveals the inner discrepancies among the input user data without explicit designation of partition/cluster number or extensive calibration.

Two sets of information are alternatively updated and exchanged within APC, *i.e.*, *responsibility* $r(\cdot,\cdot)$ and *availability* $a(\cdot,\cdot)$. Specifically, the responsibility updates are broadcast, quantifying how suitable a node k is to serve as an exemplar:

$$r(i,k) \triangleq \Phi(i,k) - \max_{k' \neq k} a(i,k') + \Phi(i,k'). \tag{13}$$

The availability is then updated to valuate the appropriateness of picking k for i as exemplar when taking into account others' preferences, *i.e.*,

$$a(i,k) \triangleq \min\big\{0, r(k,k) + \sum_{i' \notin \{i,k\}} \max\left(0, r(i',k)\right)\big\}, \tag{14}$$

if $i \neq k$, and otherwise

$$a(k,k) \triangleq \sum_{i' \neq k} \max\left(0, r(i',k)\right). \tag{15}$$

The iterations are to maximize the net similarity ρ^i of each user i [27] with others until crowd partitioning converges, *i.e.*,

$$\rho^i = \max_j \big\{a(i,j) + r(i,j)\big\}, \quad \forall i. \tag{16}$$

Given the preliminary partitions, we find the latent statistics of intra-flow similarities to acquire a big picture of crowd-flows without explicit labeling. Specifically, we find the mean $\overline{\Phi}$ and standard deviation σ^{Φ} of all the mutual similarities $\Phi(i,j)$'s from those who are partitioned into the same flow. Note that this stage is done only once, and hence the self-learning does not incur much overhead to the system.

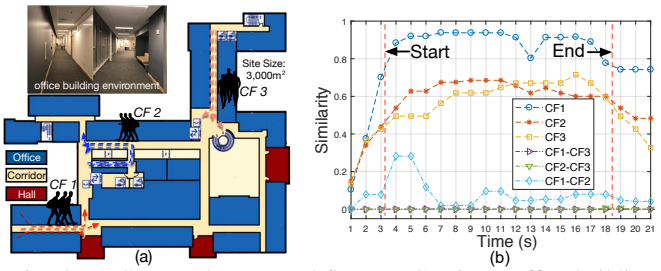

Fig. 5: (a) Case study: 3 crowd-flows or CFs in an office building (Dataset A); (b) temporal similarity dynamics of the CFs in (a).

(b) Fast co-flow edge detection:

Given the aforementioned $\overline{\Phi}$ and σ^{Φ} via APC, we design a fast edge-detection rule (for the decision function $\Gamma(\cdot)$ in Eq. (3)). Specifically, an edge is detected between a pair of users i and j (indicated by $\mathbf{A}_{ij}$'s in the augmented adjacency matrix) if the mutual similarity $\Phi(i,j)$ satisfies $\Phi(i,j) \geq \overline{\Phi} - \alpha \cdot \sigma^{\Phi}$, or otherwise $\mathbf{A}_{ij}$ remains constant, *i.e.*,

$$\mathbf{A}_{ij} \triangleq \begin{cases} \mathbf{A}_{ij} + 1, & \text{if} \quad \Phi(i,j) \geq \overline{\Phi} - \alpha \cdot \sigma^{\Phi}; \\ \mathbf{A}_{ij}, & \text{otherwise}, \end{cases} \tag{17}$$

where $\alpha > 0$ is a predefined parameter (empirically evaluated in Sec. VI), and all $\mathbf{A}_{ij}$'s are initially to 0.

Recall that for efficient deployment, given the detected WiFi set, `CFid` first partitions the incoming users before making more computation-intensive mutual comparisons. Specifically, if two users share too few APs (say, fewer than three in our setting), or the average *LB_K* is not sufficiently small (Sec. IV-B) in spite of some shared APs, we do not compare the rest of their similarities (Secs. IV-C & IV-D) and quickly determine $\mathbf{A}_{ij} = 0$.

Fig. 5(a) shows a typical crowd analytics scenario in a $3{,}000\text{m}^2$ office building (from Dataset A; each CF consists of 3 users' walking traces), where different CFs are identified and labeled. Fig. 5(b) further shows the temporal similarities from each other. Specifically, for each data point we find the mean similarities $\Phi(i,j)$'s between the users within and across the crowd-flows. One can see that at the beginning (before around 3s) when the flows are not formed, the intra- and inter-flow similarities are low. When the flows are formed gradually (after around 5s), the inter-flow similarities become notably higher than the intra ones. The 3 crowds split at about 18s. Note that the intra similarities are not always 0, and may show less noticeable temporal variations.

B. Efficient Crowd-Flow Graph Construction & Identification

The basic idea of our CFI works as follows. Since the incoming edges arrive and evolve in a sequential and random order due to the dynamics of opportunistically-encountered users, we consider the setting of *dynamic graph streaming* such that each edge is examined only once. Recall that many intra-flow edges are expected to be compared with inter-flow ones (Def. 5). So, an intra-flow edge is more likely to increase the degrees of identified crowd-flows before an inter one likely splits the flows due to excessive degrees. Merging smaller crowds with the larger ones strengthens such an increment. The edges are then examined sequentially to check the degree updates and reduce the entire complexity [19].

Algorithm 2: Crowd-Flow Identification in `CFid`.

Input: Stream of detected edges (pairs of users who are similar) and the maximum crowd-flow size $C_{\max} \geq 1$;
Output: $\mathbf{F}$: identified crowd-flows (CFs); $\mathbf{FID}$: index set of discovered crowd-flows *w.r.t.* each user;

1 $\boldsymbol{\delta} \leftarrow$ zeros$(1, N)$; $\mathbf{C} \leftarrow \{\}$; /* Initial degree $\boldsymbol{\delta}$ & flow size $\mathbf{C}$ of the N users */
2 $\mathbf{FID} \leftarrow$ minus_ones$(1, N)$; $l \leftarrow 0$; /* l: total index */
3 **for** $e_{ij} \in \mathbf{S}$ **do**
4 **if** $\mathbf{FID}[i] == -1$ **then**
5 $\mathbf{FID}[i] \leftarrow l$; $l \leftarrow l + 1$; /* New CF id for i */
6 **end**
7 **if** $\mathbf{FID}[j] == -1$ **then**
8 $\mathbf{FID}[j] \leftarrow l$; $l \leftarrow l + 1$; /* New CF id for j */
9 **end**
10 $\boldsymbol{\delta}[i] \leftarrow \boldsymbol{\delta}[i] + 1$; $\boldsymbol{\delta}[j] \leftarrow \boldsymbol{\delta}[j] + 1$; /* Degree update */ /* Size updates of crowd-flows i and j */
11 $\mathbf{C}[\mathbf{FID}[i]] \leftarrow \mathbf{C}[\mathbf{FID}[i]] + 1$; $\mathbf{C}[\mathbf{FID}[j]] \leftarrow \mathbf{C}[\mathbf{FID}[j]] + 1$;
12 **if** $\mathbf{C}[\mathbf{FID}[i]] \leq C_{\max}$ && $\mathbf{C}[\mathbf{FID}[j]] \leq C_{\max}$ **then**
13 **if** $\mathbf{C}[\mathbf{FID}[j]] \geq \mathbf{C}[\mathbf{FID}[i]]$ **then**
14 $\mathbf{C}[\mathbf{FID}[j]] \leftarrow \mathbf{C}[\mathbf{FID}[j]] + \boldsymbol{\delta}[i]$; /* Merge i */
15 $\mathbf{C}[\mathbf{FID}[i]] \leftarrow \mathbf{C}[\mathbf{FID}[i]] - \boldsymbol{\delta}[i]$; $\mathbf{FID}[i] \leftarrow \mathbf{FID}[j]$;
16 **else**
17 $\mathbf{C}[\mathbf{FID}[i]] \leftarrow \mathbf{C}[\mathbf{FID}[i]] + \boldsymbol{\delta}[j]$; /* Merge j */
18 $\mathbf{C}[\mathbf{FID}[j]] \leftarrow \mathbf{C}[\mathbf{FID}[j]] - \boldsymbol{\delta}[j]$; $\mathbf{FID}[j] \leftarrow \mathbf{FID}[i]$;
19 **end**
20 **end**
21 **end**
22 return $\mathbf{F} \leftarrow$ unique$(\mathbf{FID})$ and $\mathbf{FID}$;

Algo. 2 illustrates the clustering process upon the streamed edges of the crowd-flows. Specifically, given each arrival of a detected edge (co-flow features), we first initialize the crowds for each user if s/he has not yet been assigned (Lines 5–8). Then, `CFid` updates the degrees $\boldsymbol{\delta}$ and flow sizes $\mathbf{C}$, respectively, which are two critical measures of flows (Lines 10–11). If both flows are not yet sufficiently large (say, smaller than a customizable $C_{\max}$), `CFid` merges the user from the smaller flow into the larger one (Lines 12–20). If both flows (subgraphs) have the same number of users (vertices), *i.e.*, $\mathbf{C}[\mathbf{FID}[i]] = \mathbf{C}[\mathbf{FID}[j]]$, we assign one user to the other's flow with the edges of newer timestamps. After all edges are processed, the final CF set (identified flows $\mathbf{F}$ and index set $\mathbf{FID}$) is returned to the central monitor. Then, in Eq. (3), decision variable $b_i^{\mathbf{FID}[i]} = 1$ while all other $b_i^l | l \neq \mathbf{FID}[i] = 0$.

Our design benefits flow identification in three aspects.

Flexibility: We transform the dynamic flow identification to a graph streaming problem [18,19], allowing the incoming detected edges to arrive one after another in a random order. This way, closeness detection can be conducted by multiple processors/machines and then the resulting edges can be fed dynamically, making `CFid` flexible in dynamic flow analysis.

Efficiency: As shown in Algo. 2, each streamed edge is processed and examined only once, and hence the computation complexity is linear in the number of edges.

Identifiability: Given the incoming edge, the intra-cluster degrees tend to get larger than the inter-cluster ones [19]. This way, the metric in Eq. (3), the modularity of graph $\mathcal{G}$, is dynamically enhanced [19], making the formed subgraphs (crowds) denser inside and more separated mutually.

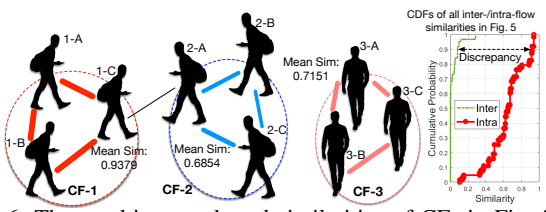

Fig. 6: The resulting graph and similarities of CFs in Fig. 5(a).

Fig. 6 shows a snapshot of a constructed crowd-flow graph (with the link width representing correlation) for Fig. 5(a). Despite some rather weak links across different CFs (due likely to similar WiFi coverage in Fig. 5(a)), other strong links are still detected and the CFs are identified via Algo. 2.

The complexity of `CFid` is summarized briefly. Given $\mathcal{O}(L)$ APs and $\mathcal{O}(N)$ users, each of whom collects $\mathcal{O}(G)$ magnetic readings, the complexity in edge detection is asymptotically $\mathcal{O}(LN^2 + G^2N^2)$. The graph stream clustering is linear in constructed edges, *i.e.*, $\mathcal{O}(N^2)$. So, the overall complexity of `CFid` is asymptotically $\mathcal{O}(LN^2 + G^2N^2)$. Recall that with coarse partitioning via WiFi AP set and *LB_K* sequence check (Sec. IV), the users can be divided before more sophisticated computation is done within each partition.

VI. Experimental Evaluation

A. Experimental Settings

Data & system preparation. To prepare our Dataset A (Sec. III-B), we collected data from total 50 volunteers (age: 23∼30; height: 155∼180cm; 47 males and 3 females) along the walking traces. Each trace collection ranges from 15s (apartment) to 30min (shopping mall), depending on the test sites. For Datasets B and C, we emulate the crowd-flow scenarios where 16 users are walking randomly over the time, forming different crowds which follow the random waypoint mobility model with resting [14]. For all datasets, we do not assume a constant walking speed, and participants walk at their own usual pace (device holding postures may vary) at the site. Volunteers either label the crowd-flow ground-truths (IDs) explicitly by themselves, or implicitly via the mutual Bluetooth Low Energy detection [10,11].

For a thorough sensitivity/deployment evaluation, we further utilize additional 332 walking traces from the 50 volunteers. In particular, 200 of them form multiple crowds (maximum 10; minimum 4) over the time for sensitivity analysis, while the rest are for device dependency evaluation. For Algo. 2, we empirically set $C_{\max} = 10$. To balance between accuracy and responsiveness (Dataset A), sampling frequency is set to 1Hz and window size $T = 5$s for WiFi, and $\omega = 5$s for magnetic reading. While some preliminary signal feature processing and filtering can be done on at their smartphones (Sec. III-B), the `CFid` core algorithm is running on a PC server with Intel i7-8700K 3.7GHz, 16GB RAM and Windows 10.

Performance metrics. (a) *Co-flow detection* (CFD): Considering whether each two users are co-flow or not, we measure the detection performance with *F1-score*, *true positive rate* (TPR) and *true negative rate* (TNR), where the notion of positive represents the co-flow state (two volunteers are in the same CF). (b) *Crowd-flow identification* (CFI): With the ground-truth labels (multi-class labels), we can find the *accuracy* of identified CFs for every involved user (whether the estimated ID matches the groundtruth one). We also evaluate

Dataset	A	B	C
CFD F1-Score	95.31%	92.27%	91.64%
CFD TPR	96.83%	94.44%	91.36%
CFD TNR	93.65%	92.71%	94.72%
CFI Accuracy	95.67%	95.29%	93.54%

Table I: Overall performance (CFD and CFI) on different datasets (Datasets A, B & C).

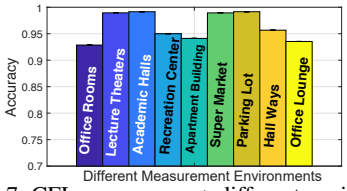

Fig. 7: CFI accuracy *w.r.t.* different environments or interferences (Datasets A, B & C).

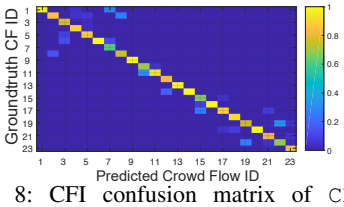

Fig. 8: CFI confusion matrix of CFid (Dataset A).

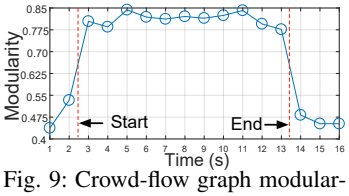

Fig. 9: Crowd-flow graph modularity (Eq. (2)) *w.r.t.* time (Dataset A).

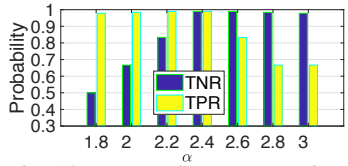

Fig. 10: TPR and TNR vs. α in the CFD decision (Eq. (17)).

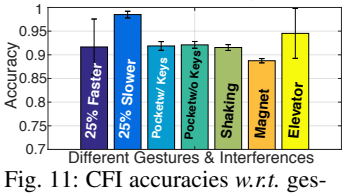

Fig. 11: CFI accuracies *w.r.t.* gestures & external interferences.

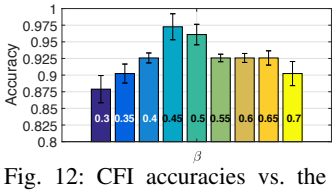

Fig. 12: CFI accuracies vs. the weight β (Eq. (12)).

the *computation overhead*, and measure *energy consumption* upon smartphones using a mobile app called PowerTutor [28].

B. Evaluation Results

(1) Overall performance. We first show the overall performance of CFid in terms of CFD and CFI.

• *Performance on different datasets*. With different datasets (A, B and C in Sec. III-B), we show in Table I the performance of CFid. CFid is found to be able to accurately detect if two users are walking in the same flow, and determine the correct crowd-flow they belong to. Due mainly to more differentiable and stronger signal features (Sec. IV-B & IV-C) in the sites of Datasets A and B (including higher popularities of WiFi APs as in Fig. 3), CFid may experience slightly higher F1-scores, TPRs/TNRs and accuracies upon them than the Dataset C.

• *Measurement environments*. By classifying different measurement scenarios of the 3 datasets, we also show the performance *w.r.t.* environments in Fig. 7. We also observe higher degradation in the apartment buildings and office rooms, as the mobility of users is more constrained there than at the other sites, yielding weaker patterns for derivation. Despite these variations, CFid achieves, in general, high accuracies across different measurement environments (often by more than 92.5%). Fig. 8 provides further details of the CFid's confusion matrix upon Dataset A, validating its accuracy.

• *Temporal dynamics of modularity*. We also illustrate the modularity dynamics [18] in Fig. 9 using Dataset A. 24 users are selected to evaluate the modularity (Eq. (2)) of their crowd-flow graph. One can see that the modularity of all involved users gradually increases and then decreases, showing their temporal dynamics that the crowd-flows form (at around 3s) and later disappear (at about 13s).

The results for all three datasets are qualitatively similar, so we will focus on Dataset A for more comprehensive analyses.

(2) Sensitivity evaluation. We also study CFid's sensitivity to various settings.

• *CFD decision boundary*: Fig. 10 shows the CFD performance vs. the decision parameter α (Eq. (17)). As α increases, TNR increases and TPR decreases, showing the changes in CFid's sensitivity and adaptivity. A smaller α indicates a stricter standard, leading to lower TNR and higher TPR, while a larger α implies tolerance at the cost of higher TNR but lower TNR. To balance these two, we chose $\alpha = 2.4$.

• *Different gestures & external interferences*: Fig. 11 shows the effects of measurement postures and external interferences on the smartphone magnetometer readings. We test traditional device-holding postures, varying walking speed (25% faster or slower than normal), shaking the mobile device with hand, inside pants pocket without or with keys. For test cases with the magnet, we attach the device body with a smartwatch charging dock of a strong magnet inside (magnitude grows by $6.94\times$) to simulate scenarios when a user accidentally brings a magnet (or magnetized object). We also conduct a similar validation (same sample size) near a moving elevator to evaluate the effect of external magnetic interference.

Clearly, different postures/gestures are likely to introduce more diverse readings. Traditional device holding achieves a CFI accuracy of $98.53\pm1.05\%$. With keys/magnets or moving elevators nearby, the magnetometer sensitivity may change and result in accuracy difference. Despite some offsets, we find the overall shape of entire magnetic magnitude sequence is preserved. CFid retrieves normalized and robust spatio-temporal patterns for identification, which are free from value differences. Faster collection speed may lead to some feature loss in some measurements. Nevertheless, CFid focuses on fusing those features which make the largest differences, hence still accurately identifying the input sequences (often $> 90\%$).

• *Weight in WiFi & geomagnetism*: Fig. 12 shows the CFI accuracy (with standard deviation) vs. the weight β in Eq. (12). As β increases, the accuracy improves and then degrades, showing a tradeoff between $\Phi^W(i,j)$ and $\Phi^G(i,j)$. A slightly small β implies modestly higher importance of geomagnetism due to its more fine-grained granularity, but too small β with too much an emphasis on magnetometer leads to over-sensitive and fluctuating identification. Considering the aforementioned tradeoff, we select $\beta = 0.45$ in our general setting.

• *Number of detected APs*: Fig. 13 shows the effects of discovered APs on CFI accuracy (with standard deviations). We randomly remove some of the APs to assess the performance degradation of CFid. Clearly, the fewer APs detected, the lower accuracy CFid may have. However, as CFid is based upon the joint decision of WiFi and geomagnetism, it can still achieve robust performance.

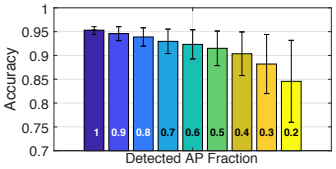

Fig. 13: CFI accuracies vs. detected AP fraction.

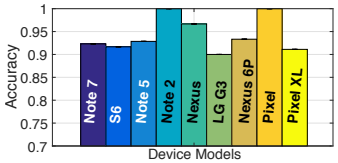

Fig. 14: CFI accuracies *w.r.t.* 9 different mobile devices.

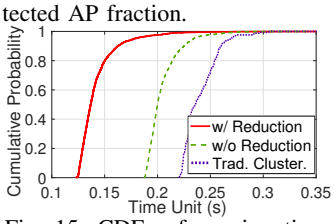

Fig. 15: CDFs of running time (compared with trad. clustering).

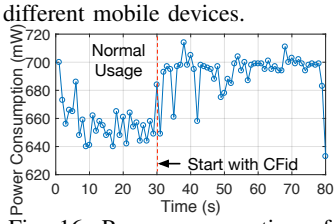

Fig. 16: Power consumption of CFid (Samsung Note 5).

(3) Deployment evaluation. Besides the aforementioned general performance and sensitivity, we evaluate the deployability of CFid.

• *Device dependency*: We have recruited the volunteers to study 9 different devices' dependency for CFI accuracy. Specifically, we evaluate mutual user similarities based on a total of 108 WiFi/geomagnetic traces in the 9 same crowd formations (on the same walking paths), where each device model contributes 12 traces for each flow evaluation. As shown in Fig. 14, we select several smartphone models. Different identification accuracies mainly come from sensitivity differences in their chipset hardware, and hence accuracy varies with the phone model. However, CFid shows generally good device (including backward) compatibility.

• *Computation overhead*: Fig. 15 shows the computation overhead of CFid. Thanks to the computation reduction and fast CFI (Sec. V), we achieved more than 30% more efficiency improvement than CFid without facilitation (w/o Reduction) and the traditional spectral clustering (Trad. Cluster.) [18]. In general, CFid incurs overall low computation overhead, hence adapting to large-scale CFI scenarios.

• *Energy efficiency*: Fig. 16 shows that CFid takes average power consumption of only 27.75mW (Samsung Note 5 as an example), and the total consumption of the mobile device is increased by only 4.21% on average. Our real-world test traces showed the longest activation duration (among the 50 participants) to be 30min. Its corresponding energy consumption is approximately 13.8mWh, which is less than 0.12% of Note 5's total battery capacity (11.1Wh). So, CFid is energy-efficient, leaving a small power consumption footprint upon mobiles. Other mobile platforms show qualitatively similar results (which are thus omitted). In summary, on average, CFid consumes only 25.1347±4.1758mW, and overall 4.37±1.8924% additional energy (above normal usage) on the Android platforms we used.

VII. Conclusion

We have proposed CFid, an efficient crowd-flow identification system based on spatio-temporal signal fusion. It takes into account the hybrid similarities in WiFi and geomagnetic measurements when the target users are walking together with smartphones. Without reliance upon location/pairing, we have designed fine-grained features and identified their similarities for co-flow detection. A graph stream clustering problem is then formulated, where each of the users is considered as a vertex and their closeness forms dynamically incoming edges. Then, CFid efficiently identifies flows which are dense subgraphs within the constructed graph. Our experimental studies with 3 different datasets have validated accuracy, efficiency and flexibility of CFid in identifying crowd-flows.

References

[1] "Crowd analytics market worth $1,142.5 million by 2021," https://www.marketsandmarkets.com/PressReleases/crowd-analytics.asp, 2018.

[2] K. K. Rachuri, C. Mascolo *et al.*, "SociableSense: Exploring the trade-offs of adaptive sampling and computation offloading for social sensing," in *Proc. ACM MobiCom*, 2011.

[3] H. Du, Z. Yu *et al.*, "Recognition of group mobility level and group structure with mobile devices," *IEEE TMC*, 2018.

[4] H. Zhang, W. Du *et al.*, "An acoustic-based encounter profiling system," *IEEE TMC*, 2018.

[5] H. Shao, S. Yao *et al.*, "A constrained maximum likelihood estimator for unguided social sensing," in *Proc. IEEE INFOCOM*, 2018.

[6] R. Sen, Y. Lee *et al.*, "GruMon: Fast and accurate group monitoring for heterogeneous urban spaces," in *Proc. ACM SenSys*, 2014.

[7] Q. Zhu, M. Y. S. Uddin *et al.*, "Spatiotemporal scheduling for crowd augmented urban sensing," in *Proc. IEEE INFOCOM*, 2018.

[8] S. He and K. G. Shin, "Steering crowdsourced signal map construction via Bayesian compressive sensing," in *Proc. IEEE INFOCOM*, 2018.

[9] A. Montanari, S. Nawaz *et al.*, "A study of Bluetooth low energy performance for human proximity detection in the workplace," in *Proc. IEEE PerCom*, 2017.

[10] P. Sapiezynski, A. Stopczynski *et al.*, "Inferring person-to-person proximity using WiFi signals," *Proc. ACM IMWUT*, 2017.

[11] S. Liu, Y. Jiang *et al.*, "Face-to-face proximity estimation using Bluetooth on smartphones," *IEEE TMC*, 2014.

[12] S. He and K. G. Shin, "Geomagnetism for smartphone-based indoor localization: Challenges, advances, and comparisons," *ACM CSUR*, 2018.

[13] M. B. Kjærgaard, M. Wirz *et al.*, "Detecting pedestrian flocks by fusion of multi-modal sensors in mobile phones," in *Proc. ACM UbiComp*, 2012.

[14] J. Jun, Y. Gu *et al.*, "Social-Loc: Improving indoor localization with social sensing," in *Proc. ACM SenSys*, 2013.

[15] H. Hong, C. Luo *et al.*, "SocialProbe: Understanding social interaction through passive WiFi monitoring," in *Proc. MobiQuitous*, 2016.

[16] S. P. Borgatti and M. G. Everett, "A graph-theoretic perspective on centrality," *Social Networks*, vol. 28, no. 4, pp. 466–484, 2006.

[17] Y. Shu, K. G. Shin *et al.*, "Last-mile navigation using smartphones," in *Proc. ACM MobiCom*, 2015.

[18] S. Fortunato, "Community detection in graphs," *Physics Reports*, 2010.

[19] A. Hollocou, J. Maudet *et al.*, "A streaming algorithm for graph clustering," in *Proc. NIPS Wokshop*, 2017.

[20] P. Barsocchi, A. Crivello *et al.*, "A multisource and multivariate dataset for indoor localization methods based on WLAN and geo-magnetic field fingerprinting," in *Proc. IPIN*, 2016.

[21] P. Lazik, N. Rajagopal *et al.*, "ALPS: A Bluetooth and ultrasound platform for mapping and localization," in *Proc. ACM SenSys*, 2015.

[22] C. Sammut and G. I. Webb, *Encyclopedia of Machine Learning*. Springer Science & Business Media, 2011.

[23] L. A. Adamic and E. Adar, "Friends and neighbors on the web," *Social Networks*, 2003.

[24] E. J. Keogh, "Exact indexing of dynamic time warping," in *Proc. VLDB*, 2002.

[25] J. Tang, M. Qu *et al.*, "LINE: Large-scale information network embedding," in *Proc. WWW*, 2015.

[26] G. Brown, A. Pocock *et al.*, "Conditional likelihood maximisation: A unifying framework for information theoretic feature selection," *JMLR*, 2012.

[27] B. J. Frey and D. Dueck, "Clustering by passing messages between data points," *Science*, 2007.

[28] L. Zhang, B. Tiwana *et al.*, "Accurate online power estimation and automatic battery behavior based power model generation for smartphones," in *Proc. IEEE/ACM/IFIP CODES+ISSS*, 2010.