A Regularization Framework for Fingerprint-based Reconstruction of Mobile Trajectories

Duc A. Tran Ting Zhang Siyuan Gong
Computer Science Department,
University of Massachusetts, Boston
Email: {duc.tran, ting.zhang001, siyuan.gong001}@umb.edu

***Abstract*—In many systems, due to the lack of an adequate positioning capability or the need for energy saving, it is infeasible to track the location of a mobile device as it is moving. Its trajectory, however, may be reconstructed from the real-time fingerprint data that are obtained by the sensors built in the device. For this purpose, we investigate a regularization framework aimed to maximize the localization accuracy by taking into account the spatiotemporal properties regarding the fingerprint space in relation to the location space. The viability of this framework is demonstrated in an evaluation using real-world datasets, which shows its potential to outperform conventional approaches to location fingerprinting.**

I. Introduction

Location fingerprinting is a widely adopted approach for GPS-free localization. In location fingerprinting, a fingerprint at a specific location is a vector of location-sensitive measurements observed about the mobile device at this particular location. For indoor environments, such a measurement can be the received signal strength from a nearby Wi-Fi access point [1], a FM broadcasting tower [2], or a cellular tower [3]. For underwater environments, a measurement can be a profile of echo-sounding signals transmitted from the device (e.g., an AUV) to the sea floor or ping signals to the surface buoys. In general, any sensor information that is sensitive to location change, including light [4] and geomagnetic field [5], can be included in the fingerprint vector. Combining different sensor data where they apply can lead to a rich set of discriminative features for the fingerprint information.

Location fingerprinting works on the basis that if fingerprint information is obtained for a set of sample locations, then the device's location given a new fingerprint during the positioning phase can be computed by comparing to these samples. Conventional approaches employ a k-nearest-neighbor (kNN) algorithm [1] or a model-based learning technique such as support vector machines [6], [7] and manifold regularization [8], [9].

This paper is focused the problem of reconstructing the trajectory of a mobile device after its real-time fingerprints have been recorded during this trajectory. We assume that a small portion of these fingerprints are obtained with respective location information and this is the only training data that we have. The fingerprints we need to locate are that of a trajectory and so we conjecture that it is possible to improve over earlier approaches that are designed for non-trajectory fingerprint localization. On the other hand, while trajectory reconstruction is a widely studied problem in the area of computational geometry, our problem involves fingerprint information in the optimization, which does not apply to classic geometrical algorithms.

Our goal is to investigate the effectiveness of a solution framework based on formulating the fingerprint-based trajectory reconstruction problem as a regularization problem where the regularizers are to enforce two properties regarding the fingerprint space in relation to the location space and natural mobility:

- (P1) spatial smoothness: two fingerprints similar in value should correspond to nearby locations.
- (P2) temporal smoothness: the trajectory of a moving device in the real world should exhibit some degree of position smoothness over time.

The usefulness of these two properties have been substantiated, for example, property (P1) for non-trajectory fingerprint localization in the work of Pan et al. [9], [10] and property (P2) for fingerprint-based tracking in our earlier work [11]. The proposed regularization framework unify both (P1) and (P2) in a single framework. We derive an optimal solution to this regularization problem and evaluate its viability in comparison with earlier fingerprint techniques based on kNN and manifold regularization.

Also, in many practical scenarios, not every point in the location space is penetrable. For example, we may not be able to enter certain areas on a floor inside a building, or in an outdoor environment it is not always possible to walk straight from one point to another. Therefore, if a set of valid locations is given, the reconstructed trajectory should be restricted to only these locations. We want to investigate whether the trajectory reconstructed by our algorithm can serve as a good candidate for further enhancements in the presence of this penetrability information.

The remainder of the paper is structured as follows. §II discusses the related work. §III presents the problem formulation. The details of the trajectory reconstruction algorithm are given in §IV. Evaluation results are provided in §V. The paper is concluded in §VI.

II. Related Work

GPS-free localization in wireless networks has been a long-studied problem. A popular approach is to leverage a model correlating received signal strength (RSS) with distance [12].

Given a number of reference points (RPs), e.g., Wi-Fi access points [1] or FM broadcasting towers [2], we can locate a device by estimating its distances to these RPs based on RSS ranging and then using multi-lateration to compute the location. RSS ranging, however, is highly inaccurate due to noise interference [12]. Radio propagates differently in different directions due to obstacles such as walls, people, and furniture. Positioning based on LED lighting [4] has also been proposed with promisingly high accuracy. As visible light does not penetrate through obstacles, this technique is suitable only for short-range applications.

Location fingerprinting is a widely used range-free localization approach. An early adopter of this approach is RADAR [1], the world's first Wi-Fi RSS-based indoor positioning system, which demonstrates the viability of using RSS information to locate a wireless device without ranging. This system relies on a radio map, a lookup table that maps building locations to the corresponding RSS fingerprints empirically observed at these locations. The reference points are the Wi-Fi access points within the user's Wi-Fi range. The radio map is searched to find the closest RSS reading and its corresponding location will be used as the estimate for the user's location. RADAR represents the fingerprint approach where kNN is used to determine the location. One can also employ a model-based learning approach to relate a fingerprint to a location, for example, probabilistically using Bayesian inference [13] or non-probabilistically using an Artificial Neural Network (ANN) [14] or a Support Vector Machine (SVM) [6].

The above fingerprint techniques require a rich training set for the learning to be effective. When there are only a small number of sample fingerprints for training, we can utilize non-training fingerprints (those available but without known location), also called "unlabeled" fingerprints as in the area of machine learning. These unlabeled fingerprints can serve as a supplement to the "labeled" fingerprints (those with known location) to obtain a better leaning model, by solving a semi-supervised learning problem. A widely-used method for semi-supervised learning is via the framework of manifold regularization originally proposed by Belkin et al. in [8]. This approach has been applied to location fingerprinting [9]–[11], [15] to learn the location labels of the unlabeled fingerprints based on their weighted similarity with the labeled in a manifold structure. For example, Pan et al. [9] apply manifold regularization with a Laplacian regularization term reflecting the intrinsic manifold structure of the fingerprints; here the manifold is a weighted graph of fingerprints in which the weight of an edge connecting two fingerprints represents their similarity.

In contrast to the above works, which are on non-trajectory localization, our problem is about reconstruction of a trajectory based on the fingerprints recorded when the mobile device moved along this trajectory and so temporal smoothness should be an important factor in the optimization. This is substantiated in our earlier work [11], which, however, investigates only the temporal smoothness property. The kNN and manifold regularization techniques that have been designed for location fingerprinting take into account only spatial smoothness. We can also apply a curve-fitting algorithm to our problem, e.g., De Boor's algorithm [16], in which the trajectory would be a spline curve constructed from the locations of the labeled fingerprints serving as control points. However, this approach is merely geometrical and does not leverage the characteristics of the fingerprint space.

There are already numerous research works on mobile localization and tracking, but they make additional assumptions such that those about special sensors built in the device (e.g., gyroscope, accelerometer, compass, light sensor) [17], those about mobility-specific constraints (e.g, speed, predefined map) [18] and those that are network-specific (e.g., vehicular or wireless sensor networks) [19]. In contrast, we are interested in a framework with universal applicability in the sense that it can work orthogonally with any type of fingerprint space; i.e., applicable where fingerprint information can be of radio signals, acoustic, visible light, or geomagnetic, etc and can contain any other information so long as it is location-sensitive.

III. Preliminaries

Suppose that a mobile device has completed moving along a trajectory, during which we obtained a series of fingerprints, $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_t$; here, time is discretized into time steps 1, 2, ..., t. Each fingerprint is a m-dimensional point, $\mathbf{x}_i \in \mathcal{X} \subset \mathbb{R}^m$, where m is the number of fingerprint features, e.g., RSSI from different Wi-Fi APs, readings from inertial measurement units (accelerometer, gyroscope, magnetometer), and/or any other location-discriminative feature that is obtainable for the device. For ease of presentation, we assume that the location space is 1D; the case for higher dimensions is a trivial extension (where each coordinate is computed seperately).

We assume to know the locations of a small portion of the trajectory fingerprints, which are called "labeled" fingerprints and our goal is to compute the locations of the other fingerprints, the "unlabeled". Let y_i denote the location of $\mathbf{x}_i$ if known. For unlabeled $\mathbf{x}_i$, we simply set $y_i = 0$. We formulate the trajectory estimator as a function $f : \mathcal{X} \rightarrow \mathbb{R}$ so that the estimated location at time t given will be $f(\mathbf{x}_t)$ which, ideally, should equal its ground-truth location.

Our assumption regarding the obtainability of labeled fingerprints during the trajectory is reasonable because (1) it is necessary: localization is impossible otherwise, and (2) it is practically doable: these location labels can be made available, for example, by placing a number of "device-readable" location labels (e.g., RFID tags) in different locations in the area so a mobile device can read when it is traveling nearby. Labeled fingerprints can also be obtained for an AUV during a mission when it surfaces to get GPS fixes or for a smartphone when its GPS is enabled (previously turned off due to energy saving or signal loss in harsh environments).

A. Optimization Objectives

Ideally, the location estimate $f(\mathbf{x}_i)$ when applied to a labeled fingerprint $\mathbf{x}_i$ should match its given ground-truth location y_i. Therefore, our first objective is to minimize the estimation error with respect to the labeled fingerprints. This is quantified by minimizing

$$\min_f \left\{ E(f) = \sum_{i=1}^{t} h_i (f(\mathbf{x}_i) - y_i)^2 \right\}. \quad (1)$$

Here, notation h_i represents whether a fingerprint $\mathbf{x}_i$ is labeled ($h_i = 1$) or unlabeled ($h_i = 0$).

As aforementioned, similar fingerprints should correspond to nearby locations. Our second objective is to enforce this property – which we refer to as the *spatial smoothness* in the fingerprint space. We quantify this by, first, organizing the fingerprints into an undirected weighted graph, where each vertex is a fingerprint and each edge has a weight $w(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{|\mathbf{x}_i - \mathbf{x}_j|^2}{2\sigma^2}\right)$ (for some constant σ) reflecting the similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$, and, second, minimizing the Laplacian quadratic form of this graph:

$$\min_f \left\{ S(f) = \sum_{i=1}^{t} \sum_{j=1}^{i} w(\mathbf{x}_i, \mathbf{x}_j)(f(\mathbf{x}_i) - f(\mathbf{x}_j))^2 \right\}. \quad (2)$$

In a study of tracking the trajectory of a single device, Rallapalli et al. [20] confirmed that real-world mobility often exhibits moving at a constant velocity for a long period of time before changing speed. Consequently, the quantity $|(f(\mathbf{x}_i) - f(\mathbf{x}_{i-1})) - (f(\mathbf{x}_{i-1}) - f(\mathbf{x}_{i-2}))| = |(f(\mathbf{x}_i) + f(\mathbf{x}_{i-2})) - 2f(\mathbf{x}_{i-1})|$ for most i should be close to zero. We refer to this this temporal stability as the *temporal smoothness* in the fingerprint space. Our third objective is to enforce this property, by minimizing the following quadratic functional:

$$\min_f \left\{ T(f) = \sum_{i=3}^{t} (f(\mathbf{x}_i) + f(\mathbf{x}_{i-2}) - 2f(\mathbf{x}_{i-1}))^2 \right\}. \quad (3)$$

In practice, other optimization goals may be introduced to better satisfy the kinetic properties of the deployed application. In this paper, since we want to devise a general framework, we focus only on the three objectives above.

B. Regularization Framework

To optimize the above objectives, we combine them into a single objective function using a regularization framework [8]. Specifically, we are seeking the trajectory estimator f as a function in a reproducing kernel Hilbert space (RKHS) whose inner product is implemented by a positive definite kernel function $K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\frac{|\mathbf{x}_i - \mathbf{x}_j|^2}{2\gamma^2}\right)$ (for some constant γ, given as a system parameter). The objective is to minimize the following regularized risk:

$$\min_f \left\{ J(f) = E(f) + \lambda_S S(f) + \lambda_T T(f) + \lambda_K \|f\|_K^2 \right\}, \quad (4)$$

where there are three regularization terms:

- $\lambda_S S(f)$: a regularizer to enforce spatial smoothness
- $\lambda_T T(f)$: a regularizer to enforce temporal smoothness
- $\lambda_K \|f\|_K^2$: a regularizer added to make the minimization problem well-posed; here, f is preferred to be smooth with respect to kernel K ($\|.\|_K$ denotes the inner product in the RKHS).

Our regularization framework is a generalized combination of the manifold regularization framework of Belkin et al. [8] (by setting $\lambda_T = 0$) and the regularization framework proposed in our earlier research [11] (by setting $\lambda_S = 0$). The coefficients $\lambda_S, \lambda_T, \lambda_K \in [0, \infty)$ are the weights to control the importance of minimizing the smoothness terms, respectively. In [11], we showed that temporal smoothness is more effective than spatial smoothness for trajectory construction but we did not investigate the combined effect of both of these properties.

IV. Trajectory Reconstruction Algorithm

In this section, we derive a solution for the risk in (4). First, we will express this risk in matrix form. Let us denote the following matrices: $\mathbf{f} = [f(\mathbf{x}_1), f(\mathbf{x}_2), ..., f(\mathbf{x}_t)]^\intercal$, $\mathbf{y} = [y_1, y_2, ..., y_t]^\intercal$ (y_i is set to zero by default for unlabeled $\mathbf{x}_i$), $\mathbf{H} = diag(h_1, h_2, ..., h_t)$, the identity matrix $\mathbf{I} = diag(\underbrace{1, 1, ..., 1}_{t})$, the kernel matrix $\mathbf{K} = [k_{ij} = K(\mathbf{x}_j, \mathbf{x}_i)]_{t \times t}$, the Laplacian matrix of the fingerprint graph

$$\mathbf{L} = \left[l_{ij} = \begin{cases} -w(\mathbf{x}_i, \mathbf{x}_j) & \text{if } i \neq j \\ \sum_{k=1}^{t} w(\mathbf{x}_i, \mathbf{x}_k) & \text{otherwise} \end{cases} \right]_{t \times t}$$

and $\mathbf{B} = [b_{ij}]_{t \times t} = \mathbf{D}^\intercal \mathbf{D}$ where $\mathbf{D}$ is the second-order difference matrix,

$$\mathbf{D} = \left[d_{ij} = \begin{cases} 1 & \text{if } (i \geq 3) \wedge (j = i \vee j = i-2) \\ -2 & \text{if } (i \geq 3) \wedge (j = i-1) \\ 0 & \text{otherwise} \end{cases} \right]_{t \times t}$$

More specifically, $\mathbf{B}$ is the following pentadiagonal matrix

$$\begin{bmatrix} 1 & -2 & 1 & 0 & ... & ... & ... & ... & 0 \\ -2 & 5 & -4 & 1 & 0 & ... & ... & ... & 0 \\ 1 & -4 & 6 & -4 & 1 & 0 & ... & ... & 0 \\ 0 & 1 & -4 & 6 & -4 & 1 & 0 & ... & 0 \\ 0 & 0 & 1 & -4 & 6 & -4 & 1 & ... & 0 \\ ... & ... & ... & ... & ... & ... & ... & ... & ... \\ 0 & ... & ... & 0 & 1 & -4 & 6 & -4 & 1 \\ 0 & ... & ... & ... & 0 & 1 & -4 & 5 & -2 \\ 0 & ... & ... & ... & ... & 0 & 1 & -2 & 1 \end{bmatrix}_{t \times t}$$

Proposition IV.1. *The minimizer of risk $J(f)$ admits the following solution, in matrix form,*

$$\mathbf{f} = (\lambda_K \mathbf{I} + \mathbf{K}(\mathbf{H} + \lambda_S \mathbf{L} + \lambda_T \mathbf{B}))^{-1} \mathbf{K}\mathbf{H}\mathbf{y}. \quad (5)$$

Proof. Since f is a function in the reproducing kernel Hilber space associated with the kernel function K, mimicking the derivations based on the extended Representer Theorem as in [8], we are looking for f as a finite linear combination of kernel products evaluated on the input points $\mathbf{x}_1$, $\mathbf{x}_2$, ..., $\mathbf{x}_t$; i.e., $f(\mathbf{x}) = \sum_{i=1}^{t} \alpha_i K(\mathbf{x}_i, \mathbf{x})$. In matrix form, $\mathbf{f} = \mathbf{K}\boldsymbol{\alpha}$, where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, ..., \alpha_t]^\intercal$ (unknown).

Express the functionals in Eqs. (1), (2), and (3) in matrix form as follows: $E(f) = (\mathbf{f} - \mathbf{y})^\intercal \mathbf{H}(\mathbf{f} - \mathbf{y})$, $S(f) = \mathbf{f}^\intercal \mathbf{L}\mathbf{f}$,

$T(f) = \mathbf{f}^\intercal\mathbf{B}\mathbf{f}$. In the matrix form, $\|\mathbf{f}\|_K^2 = \boldsymbol{\alpha}^\intercal\mathbf{K}\boldsymbol{\alpha}$. Thus, the risk $J(f)$ in Eq. (4) can be expressed in matrix form (note that $\mathbf{K}$ is symmetric; hence, $\mathbf{K} = \mathbf{K}^\intercal$):

$$\begin{aligned}J(f) &= (\mathbf{f}-\mathbf{y})^\intercal\mathbf{H}(\mathbf{f}-\mathbf{y}) + \lambda_S\mathbf{f}^\intercal\mathbf{L}\mathbf{f} + \lambda_T\mathbf{f}^\intercal\mathbf{B}\mathbf{f} + \lambda_K\boldsymbol{\alpha}^\intercal\mathbf{K}\boldsymbol{\alpha}\\ &= \mathbf{f}^\intercal(\mathbf{H}+\lambda_S\mathbf{L}+\lambda_T\mathbf{B})\mathbf{f} - 2\mathbf{y}^\intercal\mathbf{H}\mathbf{f} + \mathbf{y}^\intercal\mathbf{H}\mathbf{y} + \lambda_K\boldsymbol{\alpha}^\intercal\mathbf{K}\boldsymbol{\alpha}\\ &= \boldsymbol{\alpha}^\intercal\mathbf{K}(\mathbf{H}+\lambda_S\mathbf{L}+\lambda_T\mathbf{B})\mathbf{K}\boldsymbol{\alpha} - \\ &\quad 2\mathbf{y}^\intercal\mathbf{H}\mathbf{K}\boldsymbol{\alpha} + \mathbf{y}^\intercal\mathbf{H}\mathbf{y} + \lambda_K\boldsymbol{\alpha}^\intercal\mathbf{K}\boldsymbol{\alpha}\\ &= \boldsymbol{\alpha}^\intercal\mathbf{K}\left(\lambda_K\mathbf{I} + (\mathbf{H}+\lambda_S\mathbf{L}+\lambda_T\mathbf{B})\mathbf{K}\right)\boldsymbol{\alpha} - \\ &\quad 2\mathbf{y}^\intercal\mathbf{H}\mathbf{K}\boldsymbol{\alpha} + \mathbf{y}^\intercal\mathbf{H}\mathbf{y}.\end{aligned}$$

Let $\mathbf{Q} = \mathbf{K}\left(\lambda_K\mathbf{I} + (\mathbf{H}+\lambda_S\mathbf{L}+\lambda_T\mathbf{B})\mathbf{K}\right)$, we have

$$J(f) = \boldsymbol{\alpha}^\intercal\mathbf{Q}\boldsymbol{\alpha} - 2\mathbf{y}^\intercal\mathbf{H}\mathbf{K}\boldsymbol{\alpha} + \mathbf{y}^\intercal\mathbf{H}\mathbf{y}. \tag{6}$$

To minimize J, set its derivative with respect to $\boldsymbol{\alpha}$ to zero,

$$\frac{\partial J}{\partial \boldsymbol{\alpha}} = (\mathbf{Q}+\mathbf{Q}^\intercal)\boldsymbol{\alpha} - 2\mathbf{K}\mathbf{H}\mathbf{y} = \mathbf{0}. \tag{7}$$

Because of the symmetry of matrices $\mathbf{I}$, $\mathbf{K}$, $\mathbf{H}$, $\mathbf{B}$, and $\mathbf{L}$,

$$\begin{aligned}\mathbf{Q}^\intercal &= \left(\lambda_K\mathbf{I} + \mathbf{K}(\mathbf{H}+\lambda_S\mathbf{L}+\lambda_T\mathbf{B})\right)\mathbf{K}\\ &= \lambda_K\mathbf{K} + \mathbf{K}(\mathbf{H}+\lambda_S\mathbf{L}+\lambda_T\mathbf{B})\mathbf{K}\\ &= \mathbf{K}\left(\lambda_K\mathbf{I} + (\mathbf{H}+\lambda_S\mathbf{L}+\lambda_T\mathbf{B})\mathbf{K}\right) = \mathbf{Q}\\ &\Rightarrow \mathbf{Q}+\mathbf{Q}^\intercal = 2\left(\lambda_K\mathbf{I} + \mathbf{K}(\mathbf{H}+\lambda_S\mathbf{L}+\lambda_T\mathbf{B})\right)\mathbf{K}.\end{aligned}$$

Thus, Eq. (7) becomes

$$2\left(\lambda_K\mathbf{I} + \mathbf{K}(\mathbf{H}+\lambda_S\mathbf{L}+\lambda_T\mathbf{B})\right)\mathbf{K}\boldsymbol{\alpha} - 2\mathbf{K}\mathbf{H}\mathbf{y} = \mathbf{0}$$

which leads to the following solution

$$\mathbf{f} = \mathbf{K}\boldsymbol{\alpha} = \left(\lambda_K\mathbf{I} + \mathbf{K}(\mathbf{H}+\lambda_S\mathbf{L}+\lambda_T\mathbf{B})\right)^{-1}\mathbf{K}\mathbf{H}\mathbf{y}.$$

□

A. Choice of Parameters

Our framework is a parametrized framework with the following parameters: λ_S, λ_T, λ_K, σ, γ. How to choose the values for all parameters is beyond the scope of this paper; we assume that they are given. In practice, the more often we see a labeled fingerprint, the less smoothing is needed, hence smaller values for the smoothing parameters, λ_S and λ_T. The kernel smoothing parameter, λ_K, should be set to a small value sufficient to make the regularization problem well-posed.

B. Incorporating Penetrability Constraints

The proposed framework does not factor in any physical location space constraint and, as such, the trajectory estimator f may return an estimate that does not correspond to a penetrable location. In many practical cases, for example where a floor plan is available, it is not difficult to obtain the coordinates of penetrable points on the floor; we call these points the "valid" points. The reconstructed trajectory should only contain valid points.

Suppose that the set of valid points is $\{v_1, v_2, ..., v_N\}$. If this set is known, the algorithm to reconstruct the trajectory can simply be revised as follows:

1) Compute $\mathbf{f} = [f_1, f_2, ..., f_t]$ according to Eq. (5).
2) Location of each unlabeled fingerprint $\mathbf{x}_i$ will be v_{i^*} where $i^* = \arg\min_j (v_j - f_i)^2$.

V. Evaluation

The evaluation was conducted using the Wi-Fi RSSI data obtained at our university in three case studies: (1) umbccul: upper-level floor of the Campus Center (124 locations, average 39 APs per location, Figure 1(a)), (2) umbcs: floor of the Computer Science department (185 locations, average 24 APs per location, Figure 1(c)), and (3) umbwheatley: first floor of the Wheatley building (189 locations, average 39 APs per location, Figure 1(e)). These three floors demonstrate different layouts, two with many narrow corridors (umbcs, umbwheatley) and one with much larger shared open space (umbccul). The corresponding RSSI from unreachable access points is set to -100db by default. At each sample location, the corresponding fingerprint is the average of the RSSIs observed at this location. RSSI was measured by a person carrying an Android phone in no particular heading direction.

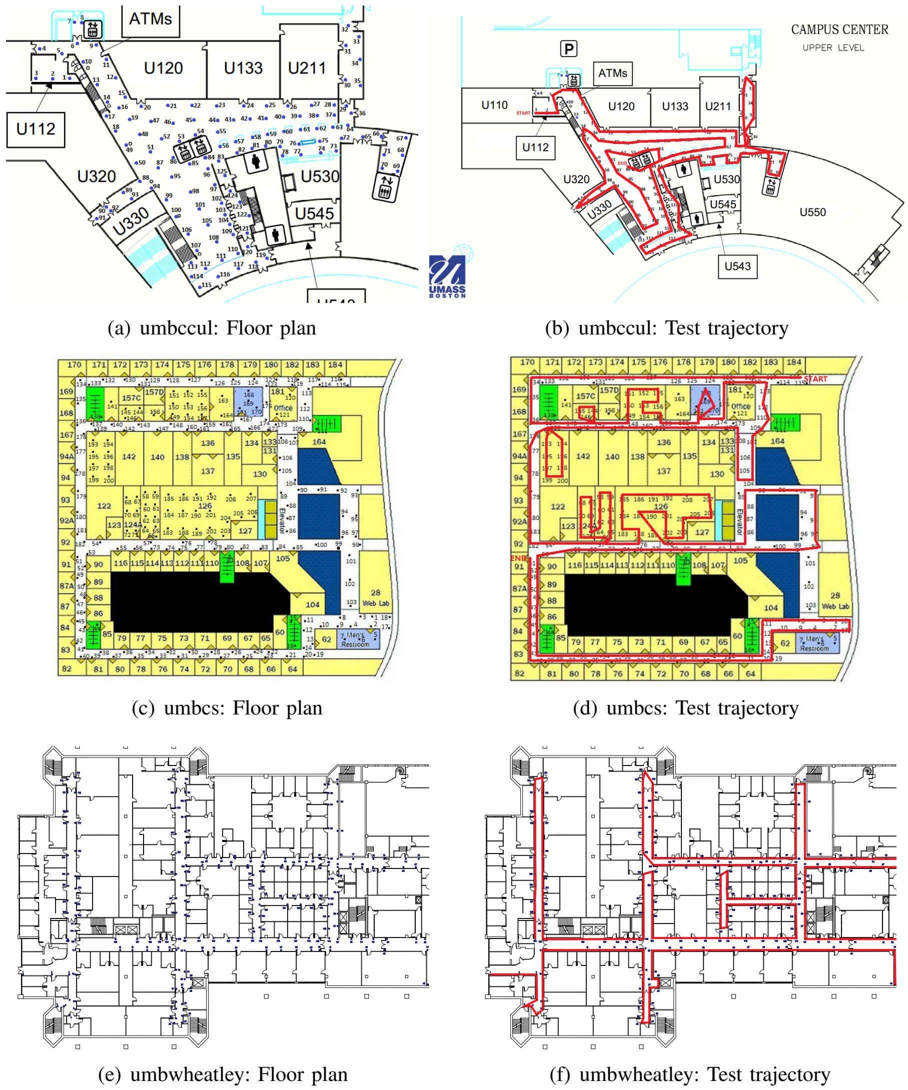
The evaluation was conducted with three trajectories, one for each case study; shown in Figures 1(d), 1(f), and 1(b). In each trajectory under consideration, the availability of fingerprints with known location is parameterized by a probability, called the label rate, $p_l \in \{0.1, 0.3, 0.5, 0.7\}$. For example, if $p_l = 0.1$, roughly 10% of the fingerprints on the trajectory are labeled and the rest unlabeled. For each choice of p_l, the results are averaged over five random runs. Parameter λ_K is set to 10^{-6}, which is small enough to make our minimization problem solvable in all case studies. After cross-validation for reasonable accuracy, we chose to set the Gaussian parameters $\gamma = 1$ and $\sigma = 0.1$ for the kernel function and weight function, respectively, and used this choice for all simulation runs. The range of values for the regularization coefficients is $\lambda_S, \lambda_T \in \{0, 10^{-8}, 10^{-7}, ..., 10^{-1}, 1\}$, representing ten different scales.

We compared the proposed regularization approach to the kNN approach and the manifold regularization approach. The manifold approach is simply a special case of our approach where λ_T is set to zero. The metric for comparison is "reconstruction error" computed as the average pairwise Euclidean distance between the location of each fingerprint on the reconstructed trajectory with the corresponding ground-truth location.

A. Effect of Spatio-Temporal Smoothness

Figure 2 and Figure 3 show the heatmap of reconstruction error when the label rate is low ($p_l = 0.1, 0.3$) and higher ($p_l = 0.5, 0.7$), respectively, for all choices of λ_S and λ_T. Obviously from these figure, the manifold regularization approach (i.e., by setting $\lambda_T = 0$ to ignore temporal smoothness) is far from the best. Temporal smoothness is far more dominant to affect the accuracy; choosing the right value for λ_T is crucial to obtain a good accuracy, especially when the label rate is low. For example, when only 10% of the fingerprints is labeled, over enforcing the temporal smoothness leads to worse accuracy (Figure 2(a,b,c)). For all cases, the best λ_T is either 10^{-4} or 10^{-5}.

The importance of temporal smoothness is understandable because the device's trajectory should be smooth over time. In contrast, spatial smoothness only plays a significant role when



(a) umbccul: Floor plan
(b) umbccul: Test trajectory
(c) umbcs: Floor plan
(d) umbcs: Test trajectory
(e) umbwheatley: Floor plan
(f) umbwheatley: Test trajectory

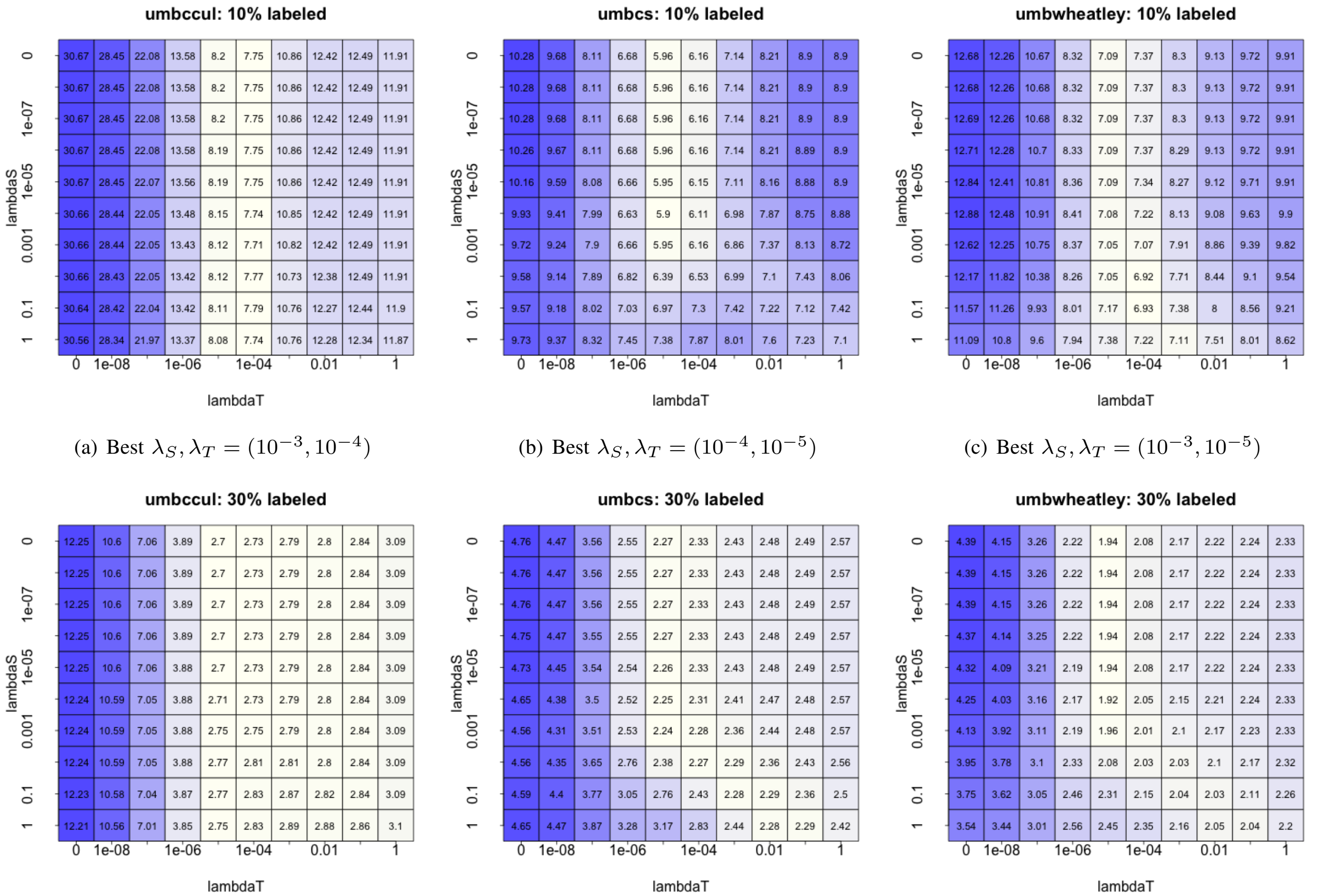
Fig. 1. Floor plans with sample locations (shown as dots) where WiFi RSSI data were obtained and test trajectories (shown in bold red): (a-b) umbccul dataset (box dimension 185m × 113m, 125 sample locations); (c-d) umbcs dataset (box dimension 68m × 63m, 208 sample locations); and (e-f) umbwheatley dataset (box dimension 111m × 68m, 189 sample locations)

temporal smoothness is not well-regularized. For example, with 10% fingerprints labeled in the umbcs dataset (Figure 2(b)), when $\lambda_T = 1$ (which is over-regularizing), increasing λ_S from 0 to 1 reduces the error from 8.9m to 7.1m, which is 20% better. On the other hand, when temporal smoothness is properly regularized, spatial smoothness can be ignored.

The above study validates (1) the importance of the regularization on temporal smoothness and (2) the superiority of the proposed framework to the original manifold regularization framework. Regarding spatial smoothness, although it is shown less influential, we do not suggest that it be removed from the regularization. We believe that its low influence is due to the particularly high fingerprint sampling rate on the trajectory in our datasets. For trajectories with lower fingerprint sampling rate, temporal smoothness should be less relevant and so spatial smoothness will play a larger role. Evaluation with more datasets will be part of our future work.

B. Comparison to kNN

We now compare the proposed regularization framework with the best choice of λ_S and λ_T with kNN. For the kNN approach, after trying with different k, we found 1NN to be the most accurate for the three datasets in the evaluation. In 1NN, the location estimate of each unlabeled fingerprint is the known location of the nearest fingerprint. The comparison results are plotted in Figure 4 for all three datasets and various label rates. It is consistently shown that our framework offers better trajectory reconstruction quality. When the label rate is low, 10% or 30%, the error is roughly 4m less for the umbccul dataset, 2m less for the umbcs dataset, and 1m less for the umbwheatley dataset. When the label rate is higher, although the error reduction in meter is less, but the reduction as a percentage is larger (more 50% improved). A visualization of



(a) Best $\lambda_S, \lambda_T = (10^{-3}, 10^{-4})$

(b) Best $\lambda_S, \lambda_T = (10^{-4}, 10^{-5})$

(c) Best $\lambda_S, \lambda_T = (10^{-3}, 10^{-5})$

(d) Best $\lambda_S, \lambda_T = (0, 10^{-5})$

(e) Best $\lambda_S, \lambda_T = (10^{-3}, 10^{-5})$

(f) Best $\lambda_S, \lambda_T = (10^{-4}, 10^{-5})$

Fig. 2. Low label availability (10%, 30% labeled): reconstruction error for various degrees of spatial smoothness (λ_S) and temporal smoothness (λ_T).

the trajectory, for the case of 10% label rate, reconstructed by 1NN is provided in Figure 6(a, b, c) and by the proposed framework in Figure 6(d, e, f).

C. Usefulness of Penetrability Information

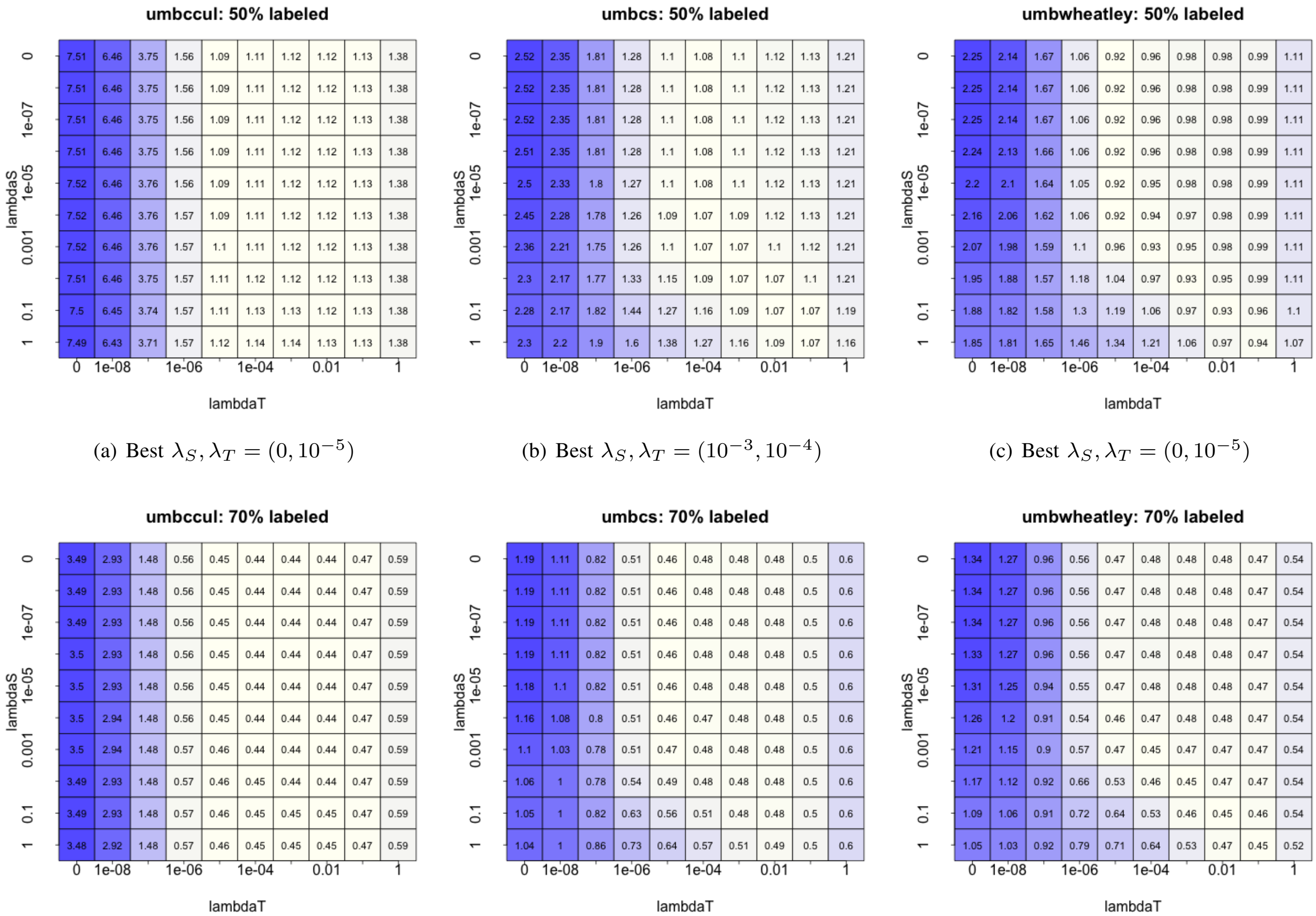
When the set of valid locations is known, as we discussed in §IV-B, the trajectory reconstruction algorithm is revised to take an addition step in which the estimated location of each fingerprint on the trajectory will be rounded to the nearest valid location. Figure 5 plots the reconstruction error of the revised algorithm in comparison to the original algorithm. Here, both algorithms use the best choice of λ_S and λ_T for the regularization. As shown in this figure, there is indeed an improvement on the error, which is not surprising. This improvement in terms of percentage is summarized in Table I. The improvement is 6% or less when the label rate is 10%, but more significant gain is achieved when more fingerprints are labeled. A visualization of the trajectory reconstructed by the revised algorithm is provided in Figure 6(g, h, i) for the case of 10% label rate. This study implies that the proposed algorithm can offer a good trajectory candidate for further enhancement if information about penetrability is available.

TABLE I
PERCENTAGE IMPROVEMENT IF VALID LOCATIONS ARE KNOWN

label rate:	10%	30%	50%	70%
umbccul	6%	13%	28%	39%
umbcs	2%	8%	15%	24%
umbwheatley	4%	14%	29%	46%

VI. Conclusions

We have investigated a regularization framework for fingerprint-based reconstruction of mobile trajectories. Its regularization enforces two important properties involving the fingerprint space and location space: spatial smoothness and temporal smoothness. This framework with a proper choice of regularization coefficients should offer better reconstruction quality compared to kNN and manifold regularization if they are applied to reconstruct the trajectory. The trajectory reconstructed by our algorithm can serve as a good starting point for further improvement if a set of valid locations is known. That said, our work remains merely theoretical and there exist challenges when applying it in practice, one of which is how to find these best coefficients. However, the proposed framework



(a) Best $\lambda_S, \lambda_T = (0, 10^{-5})$ (b) Best $\lambda_S, \lambda_T = (10^{-3}, 10^{-4})$ (c) Best $\lambda_S, \lambda_T = (0, 10^{-5})$

(d) Best $\lambda_S, \lambda_T = (0, 10^{-4})$ (e) Best $\lambda_S, \lambda_T = (0, 10^{-5})$ (f) Best $\lambda_S, \lambda_T = (10^{-3}, 10^{-4})$

Fig. 3. High label availability (50%, 70% labeled): reconstruction error for various degrees of spatial smoothness (λ_S) and temporal smoothness (λ_T).

can be used as a good benchmark to evaluate algorithms for fingerprint-based trajectory reconstruction. In our future work, we will conduction more in-depth experiments in larger time and space scales.

ACKNOWLEDGEMENTS

This work was supported by the NSF award CNS-1116430. Any opinions, findings and conclusions or recommendations expressed in this material are ours and do not necessarily reflect those of the NSF.

REFERENCES

[1] P. Bahl and V. N. Padmanabhan, "Radar: An in-building rf-based user location and tracking system," in *INFOCOM*, 2000, pp. 775–784.

[2] Y. Chen, D. Lymberopoulos, J. Liu, and B. Priyantha, "Fm-based indoor localization," in *Proceedings of the 10th international conference on Mobile systems, applications, and services*, ser. MobiSys '12. New York, NY, USA: ACM, 2012, pp. 169–182. [Online]. Available: http://doi.acm.org/10.1145/2307636.2307653

[3] A. Varshavsky, E. de Lara, J. Hightower, A. LaMarca, and V. Otsason, "Gsm indoor localization," *Pervasive Mob. Comput.*, vol. 3, no. 6, pp. 698–720, Dec. 2007. [Online]. Available: http://dx.doi.org/10.1016/j.pmcj.2007.07.004

[4] P. Hu, L. Li, C. Peng, G. Shen, and F. Zhao, "Pharos: Enable physical analytics through visible light based indoor localization," in *Proceedings of the Twelfth ACM Workshop on Hot Topics in Networks*, ser. HotNets-XII. New York, NY, USA: ACM, 2013, pp. 5:1–5:7. [Online]. Available: http://doi.acm.org/10.1145/2535771.2535790

[5] J. Chung, M. Donahoe, C. Schmandt, I.-J. Kim, P. Razavai, and M. Wiseman, "Indoor location sensing using geo-magnetism," in *Proceedings of the 9th international conference on Mobile systems, applications, and services*, ser. MobiSys '11. New York, NY, USA: ACM, 2011, pp. 141–154. [Online]. Available: http://doi.acm.org/10.1145/1999995.2000010

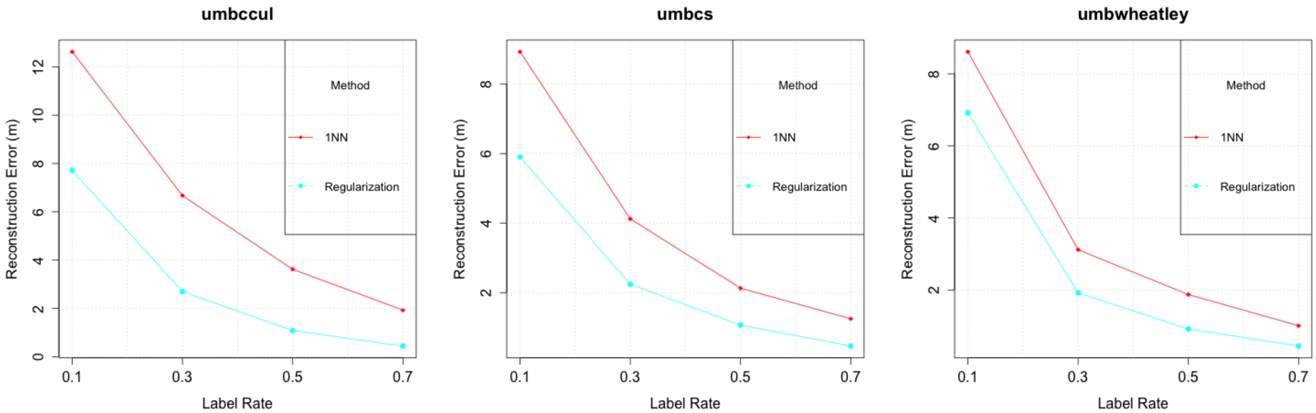
[6] M. Brunato and R. Battiti, "Statistical learning theory for location fingerprinting in wireless lans," *Comput. Netw.*, vol. 47, no. 6, pp. 825–845, Apr. 2005. [Online]. Available: http://dx.doi.org/10.1016/j.comnet.2004.09.004

[7] C. Figuera, J. L. Rojo-Álvarez, M. Wilby, I. Mora-Jiménez, and A. J. Caamaño, "Advanced support vector machines for 802.11 indoor location," *Signal Process.*, vol. 92, no. 9, pp. 2126–2136, Sep. 2012. [Online]. Available: http://dx.doi.org/10.1016/j.sigpro.2012.01.026

[8] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: A geometric framework for learning from labeled and unlabeled examples," *J. Mach. Learn. Res.*, vol. 7, pp. 2399–2434, Dec. 2006. [Online]. Available: http://dl.acm.org/citation.cfm?id=1248547.1248632

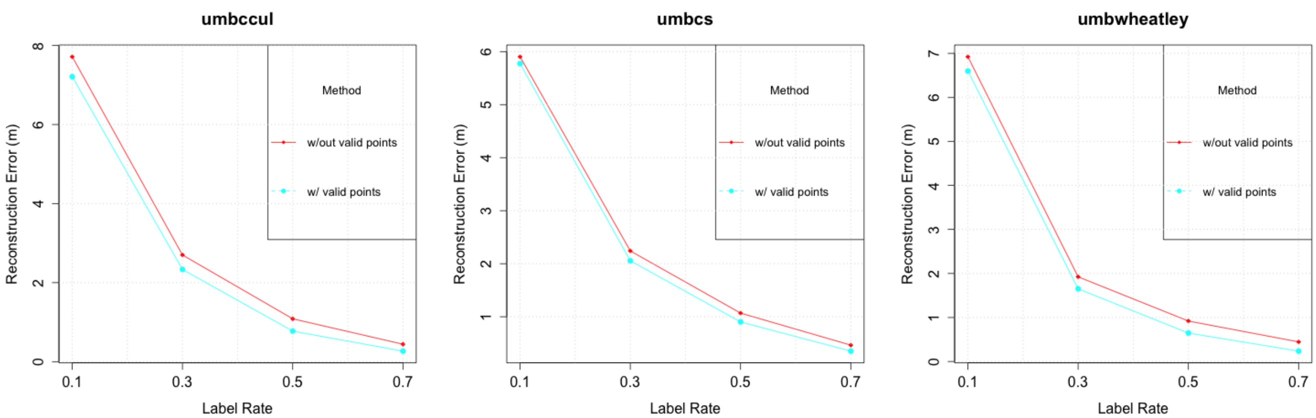
[9] J. J. Pan, Q. Yang, H. Chang, and D. Y. Yeung, "A manifold regularization approach to calibration reduction for sensor-network based tracking," in *Proceedings of the Twenty-First National Conference on Artificial Intelligence*, Boston, United States, 2006, pp. 988–993.

[10] J. J. Pan and Q. Yang, "Co-localization from labeled and unlabeled data



(a) (b) (c)

Fig. 4. Comparison between the proposed regularization approach and the kNN approach.



(a) (b) (c)

Fig. 5. The trajectory reconstruction algorithm with or without the availability of the information about the valid locations.

using graph laplacian," in *Proceedings of the Twentieth International Joint Conference on Artificial Intelligence*, Hyderabad, India, 2007, pp. 2166–2171.

[11] D. A. Tran and T. Zhang, "Fingerprint-based location tracking with hodrick-prescott filtering," in *Proceedings of the 7th IFIP Wireless and Mobile Networking Conference (WMNC 2014)*, Algarve, Portugal, May 2014.

[12] K. Whitehouse, C. Karlof, and D. Culler, "A practical evaluation of radio signal strength for ranging-based localization," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 11, no. 1, pp. 41–52, Jan. 2007. [Online]. Available: http://doi.acm.org/10.1145/1234822.1234829

[13] T. Roos, P. Myllymäki, H. Tirri, P. Misikangas, and J. Sievänen, "A probabilistic approach to WLAN user location estimation," *International Journal of Wireless Information Networks*, vol. 9, no. 3, pp. 155–164, July 2002. [Online]. Available: http://dx.doi.org/10.1023/A:1016003126882

[14] C. Laoudias, D. G. Eliades, P. Kemppi, C. G. Panayiotou, and M. M. Polycarpou, "Indoor localization using neural networks with location fingerprints," in *Proceedings of the 19th International Conference on Artificial Neural Networks: Part II*, ser. ICANN '09. Berlin, Heidelberg: Springer-Verlag, 2009, pp. 954–963.

[15] D. A. Tran and P. Truong, "Total variation regularization for training of indoor location fingerprints," in *Proceedings of ACM MOBICOM Workshop on Mission-Oriented Wireless Sensor Networking (ACM MiseNet 2013)*, Miami, Sep 2013.

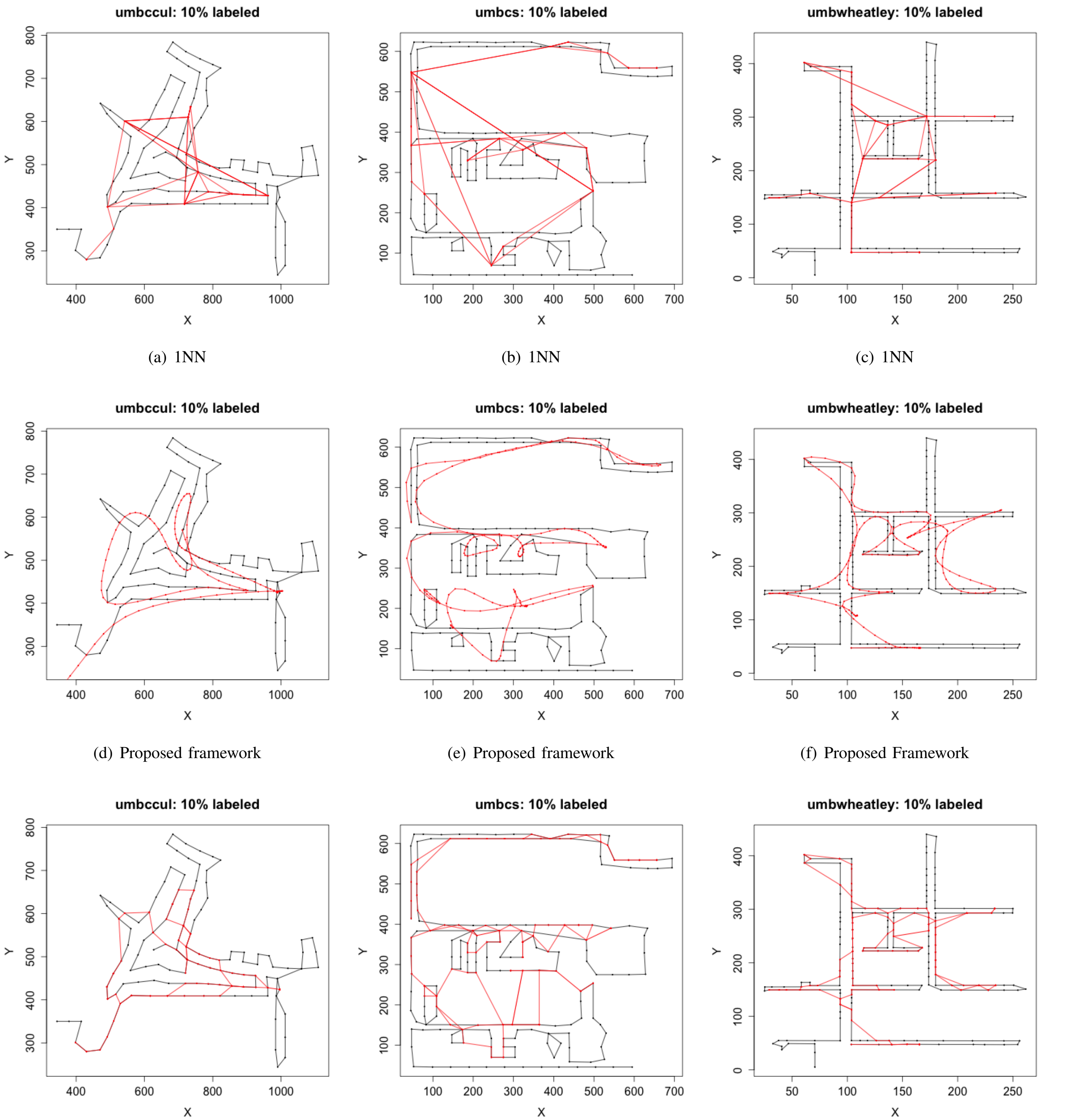
[16] C. De Boor, *A practical guide to splines; rev. ed.*, ser. Applied mathematical sciences. Berlin: Springer, 2001. [Online]. Available: https://cds.cern.ch/record/1428148

[17] C. Wu, Z. Yang, Y. Liu, and W. Xi, "Will: Wireless indoor localization without site survey," *IEEE Transactions on Parallel and Distributed Systems*, vol. 24, no. 4, pp. 839–848, 2013.

[18] Z. Yang, C. Wu, and Y. Liu, "Locating in fingerprint space: wireless indoor localization with little human intervention," in *Proceedings of the 18th annual international conference on Mobile computing and networking*, ser. Mobicom '12. New York, NY, USA: ACM, 2012, pp. 269–280. [Online]. Available: http://doi.acm.org/10.1145/2348543.2348578

[19] A. Boukerche, H. A. B. F. Oliveira, E. F. Nakamura, and A. A. F. Loureiro, "Vehicular ad hoc networks: A new challenge for localization-based systems," *Comput. Commun.*, vol. 31, no. 12, pp. 2838–2849, Jul. 2008. [Online]. Available: http://dx.doi.org/10.1016/j.comcom.2007.12.004

[20] S. Rallapalli, L. Qiu, Y. Zhang, and Y.-C. Chen, "Exploiting temporal stability and low-rank structure for localization in mobile networks," in *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, ser. MobiCom '10. New York, NY, USA: ACM, 2010, pp. 161–172. [Online]. Available: http://doi.acm.org/10.1145/1859995.1860015



(a) 1NN (b) 1NN (c) 1NN

(d) Proposed framework (e) Proposed framework (f) Proposed Framework

(g) With known set of valid locations (h) With known set of valid locations (i) With known set of valid locations

Fig. 6. 1NN vs. Proposed Algorithm vs. Proposed Algorithm with Valid Locations: Visualization of the reconstructed trajectory for the case of 10% label rate.