DeepNavi: A Deep Signal-Fusion Framework for Accurate and Applicable Indoor Navigation

QUN NIU, Sun Yat-sen University and Guangdong Key Laboratory of Information Security Technology
NING LIU*, Sun Yat-sen University and Guangdong Key Laboratory of Information Security Technology
JIANJUN HUANG, Sun Yat-sen University
YANGZE LUO, Sun Yat-sen University
SUINING HE, The Hong Kong University of Science and Technology
TAO HE, Sun Yat-sen University and Guangdong Key Laboratory of Information Security Technology
S.-H. GARY CHAN, The Hong Kong University of Science and Technology
XIAONAN LUO, Guilin University of Electronic Technology

Indoor navigation plays a crucial role in indoor location-based services. Single signal-based navigation systems, however, are prone to sensor noises, signal ambiguities and are specific to trial sites. To address these, existing work fuses different signals with user trajectories. Despite their accuracy, many of them are specific to input signals and navigation modes (e.g., spot-based or sequence-based) and are computationally expensive in large sites. Additionally, they do not give predictive uncertainty estimations, leading to a lack of trust in navigation instructions.

In this paper, we propose a unified framework for accurate indoor navigation in various modes with different inputs, termed *DeepNavi*. We exploit either convolutional or recurrent neural networks for initial feature extraction. Afterwards, we insert fully connected layers to generalize extracted signal-dependent features to a shared domain before fusion. Then, we leverage state-of-the-art ensemble learning to learn multiple predictive models. By combining them together, we further reduce the impact of signal noises and achieve high accuracy. Finally, we insert mixture density networks to model more generalized data distributions and provide uncertainty estimations. We have implemented DeepNavi and conducted extensive experiments in two different trial sites with different signal combinations. Experimental results show that DeepNavi reduces location errors by more than 20% with comparable orientation accuracy.

CCS Concepts: • **Networks → Location based services**; • **Human-centered computing → Ubiquitous and mobile computing systems and tools**.

Additional Key Words and Phrases: Indoor navigation, signal fusion, deep learning, ensemble learning

*Corresponding author.

Authors' addresses: Qun Niu, niuqun@mail2.sysu.edu.cn, Sun Yat-sen University and Guangdong Key Laboratory of Information Security Technology; Ning Liu, liuning2@mail.sysu.edu.cn, Sun Yat-sen University and Guangdong Key Laboratory of Information Security Technology; Jianjun Huang, Sun Yat-sen University, huangjj29@mail2.sysu.edu.cn; Yangze Luo, Sun Yat-sen University, luoyz5@mail2.sysu.edu.cn; Suining He, The Hong Kong University of Science and Technology, sheaa@cse.ust.hk; Tao He, Sun Yat-sen University and Guangdong Key Laboratory of Information Security Technology, hetao23@mail2.sysu.edu.cn; S.-H. Gary Chan, The Hong Kong University of Science and Technology, gchan@cs.ust.hk; Xiaonan Luo, Guilin University of Electronic Technology, luoxn@guet.edu.cn.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2019 Association for Computing Machinery.
2474-9567/2019/9-ART99 $15.00
https://doi.org/10.1145/3351257

RIGHTSLINK

ACM Reference Format:
Qun Niu, Ning Liu, Jianjun Huang, Yangze Luo, Suining He, Tao He, S.-H. Gary Chan, and Xiaonan Luo. 2019. DeepNavi: A Deep Signal-Fusion Framework for Accurate and Applicable Indoor Navigation. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 3, 3, Article 99 (September 2019), 24 pages. https://doi.org/10.1145/3351257

1 INTRODUCTION

Indoor navigation, as an important part in indoor location-based services, has received much attention in a wide range of applications, such as guiding customers to a restaurant or a conference room [38], helping visually-impaired users to work independently [22, 36] and indoor augmented reality-based gaming [9], etc. To provide high-quality indoor location-based services, one has to achieve sufficient positioning accuracy. Existing approaches extract location cues from various signals for navigation. Therefore, extracting *stable* and *distinguishing* location features is crucial for accurate indoor navigation (in terms of location and orientation errors).

Based on navigation modes, recent approaches are broadly divided into two categories: *spot-based* and *sequence-based* [27, 28]. Spot-based navigation approaches are those that estimate current location and orientation with *instant* inputs, which are collected at a given timestamp (e.g., an image [7] or a radio fingerprint [10]) at this spot. In contrast to spot-based approaches, sequence-based ones leverage sequential inputs, which are collected in a time window when the user is walking, such as videos [6] and intensity sequences [32], to extract temporal correlations and infer current location and orientation correspondingly.

In complicated indoor sites, prior navigation approaches are prone to large errors due to environment and user factors. For example, image/video-based indoor navigation achieves sufficient accuracy in sites with rich visual textures, such as shopping malls and food plazas [30, 34]. While in other sites with sparse or repetitive visual textures, such as hospitals and offices, it does not work well. Geomagnetism [1] presents distinguishing patterns in corridors due to ferromagnetic disturbances, such as steel-based buildings structures, electrical wires and appliances [16]. Nevertheless, it sometimes suffers from feature ambiguities due to similar building structures, leading to large errors.

Based on above observations, we study feature-level fusion scheme for accurate indoor navigation with wide applicability in different sites, e.g., an office area with repetitive visual features, a food plaza with geomagnetic ambiguities, as well as navigation modes, such as spot-based and sequence-based approaches. This is because features extracted from different inputs possess *complementary* location cues. By fusing them together, we generate more distinguishing location features across different sites, thus achieving high accuracy and wide applicability. However, there are several major challenges in realizing fusion-based indoor navigation systems:

- *Mode-dependent feature extraction*: In spot-based navigation, recent approaches focus on extracting features using convolutional neural networks (CNN) [23, 24, 43] with instant inputs, e.g., images. While in sequence-based navigation, others conduct feature extraction using recurrent neural networks (RNN) with temporal sequences [21], e.g., geomagnetic sequences. These networks are specially tailored to inputs and navigation modes, leading to a lack of generality.
- *Large diversities in features*: As input signals are fundamentally different from each other (in terms of sampling frequency, dimension), extracted features reside in their respective feature spaces with different dimensions. Take Wi-Fi and geomagnetism for example, their sampling frequencies vary from each other significantly (1Hz v.s. 50Hz). Therefore, it is challenging to fuse these diverse features *directly* by feature concatenation.
- *Significant navigation errors with noisy inputs*: Due to the impact of environmental noises, input signals collected by users may be noisy temporally. For example, images taken by users could be blurry due to shaky hands or a loss of focus [25]. Magnetic readings are prone to statistical noises indoors. Wi-Fi fingerprints

[1] We use "magnetism" and "geomagnetism", "magnetic" and "geomagnetic" interchangeably in this paper.

RIGHTSLINK

are fluctuating with multi-path fading and temporal blocking of nearby pedestrians [35, 44]. These noises could result in large errors.
- *A lack of uncertainty estimation*: In many learning-based navigation tasks, they formulate an optimization problem which minimizes the cost of predictions (location and orientation in our case) with the mean squared error. However, many do not capture the predictive uncertainty of estimations. Recent work estimates uncertainty values with a single Gaussian distribution. This is restrictive of data distributions, which may lead to unreliable uncertainty predictions [26].

Inspired by the recent advancement of neural networks, we propose *DeepNavi*, a unified **deep** network-based indoor **navi**gation framework. DeepNavi has wide applicability in various sites (it fuses complementary features to enhance feature distinctiveness) and different navigation modes (it uses a sequential network architecture so that it can process both instant inputs and sequential inputs). After extracting initial features from each input, DeepNavi fuses these features together. Based on the fused feature, DeepNavi estimates current location and orientation and guides users to their destinations. In summary, DeepNavi has the following novelties:

- *Unified feature extraction with wide applicability*: Spot-based navigation is a special instance of sequence-based navigation (length is 1). Consequently, we propose a unified sequential feature extraction framework based on recurrent networks, which extracts features from either instant or sequential inputs. Specifically, we extract visual features corresponding to each frame with convolutional networks, which is better with images and use recurrent networks to process other sequential inputs.
- *Feature mapping to common space through non-linear mapping*: In order to bridge the gap between preliminary input-dependent features, we further incorporate non-linear fully connected (FC) layers to generalize them. Through non-linear generalization, these features are mapped to common feature space and become less dependent on raw inputs, thus are more generalized for feature fusion.
- *Accurate navigation with ensemble learning*: We enhance the navigation accuracy with the ensemble learning. More specifically, we train several models and give *multiple* independent estimations with given inputs. Then, we *combine* them together to smooth out the impact of noises, thus reducing navigation errors by more than 20% in our trial sites (in terms of location prediction).
- *Use of mixture models for uncertainty estimation*: Beyond predicting locations and orientations, we further incorporate mixture density networks (MDN) [3] to provide predictive uncertainty estimations. By leveraging the MDN, we model our training data with more generalized mixture models, thus reducing the restrictive assumption of a single Gaussian distribution and providing uncertainty estimations.

As an example, we study indoor navigation with two signal combinations: 1) images and magnetic sequences; 2) images and Wi-Fi fingerprints. To validate the performance, we have conducted extensive experiments in two different trial sites: an office area with repetitive visual textures and a large food plaza with geomagnetic ambiguities. Experimental results demonstrate that DeepNavi reduces location errors by more than 20% and achieves comparable orientation accuracy in our trial sites. Please note that DeepNavi is a unified navigation framework, which can also process other input signals, such as videos, radio frequency fingerprints and visible light, among others, with corresponding initial feature extractor. Afterwards, we can feed them into our sequence-based network for navigation. Furthermore, DeepNavi can be integrated into other techniques, such as particle filters, to calibrate user locations and orientations. Based on the practical needs, our model can be either spot-based with instant inputs or sequence-based with sequential inputs with its general architecture. It can also be deployed in various mobile platforms, such as smartphones, Google Glasses or robots, for accurate indoor navigation.

The remainder of this paper is structured as follows. After reviewing the related work in Section 2, we overview the workflow of DeepNavi in Section 3. Then, we elaborate the network design in Section 4, followed by experimental evaluations in Section 5. We discuss the limitations of DeepNavi in Section 6 and conclude in Section 7.

RIGHTSLINK

2 RELATED WORK

As Global Positioning System (GPS) signal does not penetrate well indoors, researchers resort to other signals for accurate indoor localization and navigation [14, 15]. Based on navigation modes, recent approaches are broadly divided into two categories: spot-based and sequence-based. We compare these approaches as follows.

Spot-based navigation approaches are those that use instant inputs to estimate current location and orientation. Li et al. [29] build a context model of the environment and estimate user locations with dynamic Bayesian network using an input image. PoseNet [24] trains a neural network to regress current location and orientation. CloudNavi [41] builds a 3D point cloud using 3D cameras such as Google Tango [2]. Then, it registers collected images to the cloud and fuses particle filters with inertial readings to navigate users. In addition to images, radio frequency (RF) signals, e.g., Wi-Fi and Bluetooth, are employed during indoor localization and navigation as well. SiFi [11] collects channel information of targets [3] from a special access point with several spatially distributed antennas. Then it calculates target locations with the time of arrival measured by these antennas. Murata et al. [33] navigate users by comparing user-collected Bluetooth fingerprints with a database. Pulsar [51] estimates the angle of arrival from several light sources and triangulates targets at a spot. Despite their high accuracy in specific trial sites, these approaches are prone to signal noises. For example, image-based navigation approaches rely on rich colors and textures to find location clues. These images can sometimes be blurry due to shaky hands of users or a loss of focus, leading to unstable features and large errors [19, 20]. RF signals, on the other hand, are less stable in complicated indoor environment due to multi-path fading and temporal occlusions with wall partitions and nearby pedestrians [37, 53]. Different from these approaches that leverage single signal (e.g., visible light, Wi-Fi, Bluetooth) for indoor navigation, we study a general signal-fusion framework to extract more distinguishing location features and achieve higher accuracy.

In order to enhance the navigation accuracy, some researchers propose to fuse complementary inputs. For example, Argus [46] leverages distance constraints inferred from several images to reduce Wi-Fi localization errors. However, it asks users to select and point to a *landmark*, which is tedious and error-prone for novice users. HyRise [8] proposes a hybrid strategy that leverages Wi-Fi readings and barometers to estimate current floor. ViNav [7] fuses Wi-Fi with images to navigate users with higher accuracy and efficiency. First, it narrows down the location space of a user with a Wi-Fi fingerprint. Then it estimates current position and orientation within this area based on structure from motion (SfM) [12], which is computationally expensive during the 3D construction and navigation stage in a large trial site. DeepNavi also fuses inputs for more accurate indoor navigation. However, it advances these spot-based approaches in several ways. First, DeepNavi is a general framework based on sequential architectures. It works with both spot-based and sequence-based navigation modes, thus achieving higher applicability. Second, DeepNavi realizes direct feature-level fusion schemes. By non-linear mapping of features from different inputs to a common feature space, it fuses them directly at the feature level so as to reduce the propagation error of sequential fusion approaches (use one input to infer a candidate area and localize with another input). Furthermore, DeepNavi incorporates ensemble learning to further reduce input noises and achieves higher accuracy. By training prediction models in the offline stage, it realizes efficient and real-time navigation services.

In contrast to spot-based navigation systems, sequence-based ones extract temporal correlations from sequential inputs collected in a time window for indoor navigation. ppNav [49] proposes fingeram, which consists of diagrammed sequential Wi-Fi fingerprints, and compares it with those collected by a surveyor to give navigation instructions. To reduce the impact of fluctuating Wi-Fi signals on the localization accuracy, Tian et al. [42] extract temporal correlations based on a new propagation model to locate a target accurately. Lightitude [18] measures changes in light intensity values and aligns a query sequence with a database to localize. Other sequence-based

[2] https://en.wikipedia.org/wiki/Tango_(platform)

[3] We use "target" to refer either a human or a robot to be localized.

RIGHTSLINK

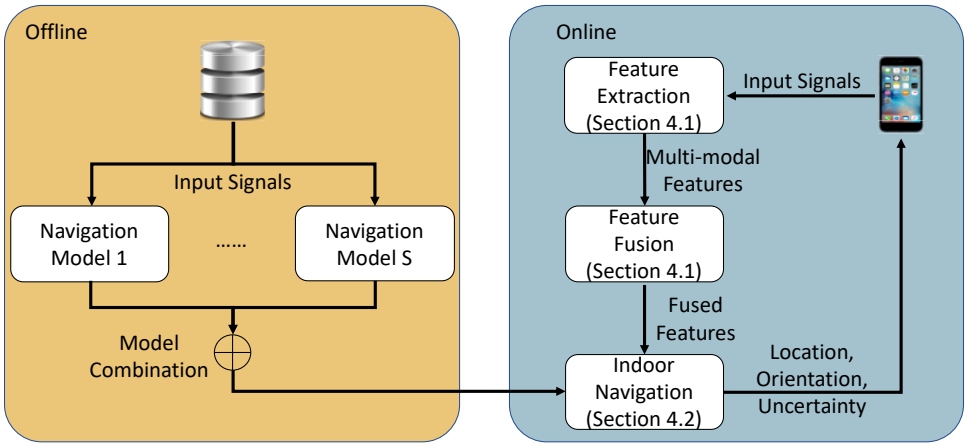


Fig. 1. The overall workflow of DeepNavi. We train our prediction models in the server and use them to predict the location and orientation of a client.

approaches further enhance the navigation accuracy by fusing several inputs. Magicol [39] proposes bi-directional particle filters and uses opportunistic Wi-Fi sensing to reduce localization errors. Travi-Navi [52] calculates the locations inferred with instant Wi-Fi and geomagnetism, respectively. Then it fuses these location estimations with equal weight values to estimate target positions. DeepNavi further advances these approaches in that it is general enough to fuse several inputs in different navigation modes. This is because spot-based navigation is a special case of sequential navigation with instant inputs. Furthermore, it introduces a direct feature-level fusion scheme to realize more effective feature fusion and uses ensemble learning to reduce the impact of environmental noises, thus achieving higher accuracy.

As discussed earlier, particle filters are widely used in many localization and navigation tasks [4, 31], because they help to reduce errors with user motion and floorplan constraints. Different from particle filter-based approaches, DeepNavi focuses on effective feature extraction and fusion for indoor navigation. Therefore, it is *orthogonal* to particle filter-based approaches. However, it is possible to integrate particle filters into our framework to further reduce errors with floorplan constraints. These particle filter-based approaches can also use our framework to opportunistically calibrate current location and orientation.

3 OVERVIEW OF THE SYSTEM WORKFLOW

We show the overall workflow of the proposed navigation system in Figure 1. To demonstrate, we navigate targets with images and geomagnetic sequences. By fusing these two inputs, we provide accurate navigation in popular areas with sparse (e.g., offices, hospitals) and abundant (e.g., shopping malls, food plaza) visual textures.

Our navigation system consists of two stages: an *offline stage* and an *online stage*. In the offline stage, a surveyor walks and uses a client program to record input signals automatically. During the survey, this surveyor initiates the program, walks in the site, thus speeding up the site survey. Then, we label the ground truth locations and orientations and store them in a database. With the collected data and labels, we leverage the ensemble learning technique to learn several models with our training data and combine them together for higher accuracy.

In the online navigation stage, a target initiates a client program to record images and geomagnetic sequences. Then the client program uploads collected data to a remote server automatically. Upon receiving the query data, DeepNavi estimates current location as well as orientation, and sends navigation instructions to this

RIGHTSLINK

Table 1. Major symbols in DeepNavi.

Notations	Definitions
$\mathbf{x}_t$	Estimated 2-D location of a target at time t
$\mathbf{o}_t$	Estimated 4-D quaternion of a target at time t
$\mathbf{Q}_t$	Output state of RNN at time t
$\mathbf{h}_t$	Hidden state of RNN at time t
$\mathbf{M}_t$	Magnetic sequence with fixed length at time t
$\mathbf{I}_t$	Image frame collected at time t
$\mathbf{k}_t$	Concatenation of location and orientation estimations at time t
S	Number of ensembles
G	Number of Gaussian distributions in mixture models

target. Specifically, we feed inputs into our network for initial feature extraction. Then we generalize them with additional FC layers and fuse features from different signals (Section 4.1). Afterwards, we discuss ensemble learning-based navigation using MDN in Section 4.2. Finally, DeepNavi gives location $\mathbf{x}_t$ and orientation $\mathbf{o}_t$ at time t, respectively.

We show major symbols used in this paper in Table 1.

4 CORE FRAMEWORK OF DEEPNAVI

Indoor navigation with a single input source can be error-prone due to environmental ambiguities in complicated indoor sites, e.g., similar geomagnetic patterns with symmetric building structures, visual ambiguities with repetitive window and door structures in office areas. These ambiguous inputs can lead to large navigation errors (in terms of location and orientation). Furthermore, due to the impact of noises, e.g., blurry images, fluctuations, extracted features from inputs can be noisy as well, leading to large errors in location and orientation predictions. Many existing approaches focus on giving numerical estimations, but do not provide predictive uncertainties, which may lead to a lack of trust in navigation instructions.

Based on these observations, we propose a unified, signal-fusion framework for indoor navigation. We illustrate the framework in Figure 2, which consists of three major components: feature extraction and fusion (Section 4.1) and target navigation (Section 4.2). Then, we discuss the benefits of the ensemble learning on the navigation performances in Section 4.3.

4.1 Unified Feature Extraction and Fusion

As discussed above, recent navigation approaches are broadly divided into two categories: spot-based and sequence-based. Spot-based navigation approaches leverage instant inputs to infer current location and orientation. In contrast to them, sequence-based navigation schemes estimate current location and orientation with sequential inputs in a time window. They advance spot-based approaches by exploiting temporal correlations of inputs to improve navigation accuracy.

In this section, we discuss the design of a unified feature extraction framework for both spot-based and sequence-based indoor navigation. Our intuition is that a spot-based navigation scheme is a special case of sequence-based navigation, where the number of temporal inputs equals 1. Based on this, we design a unified navigation framework based on recurrent network structures, e.g., RNN, which are capable of processing sequential inputs with its recurrent architecture (left part in Figure 2).

Recent approaches leverage various inputs for indoor navigation, such as Wi-Fi, radio frequency identification (RFID), images, visible light and geomagnetism, among others. These inputs vary from each other in various

RIGHTSLINK

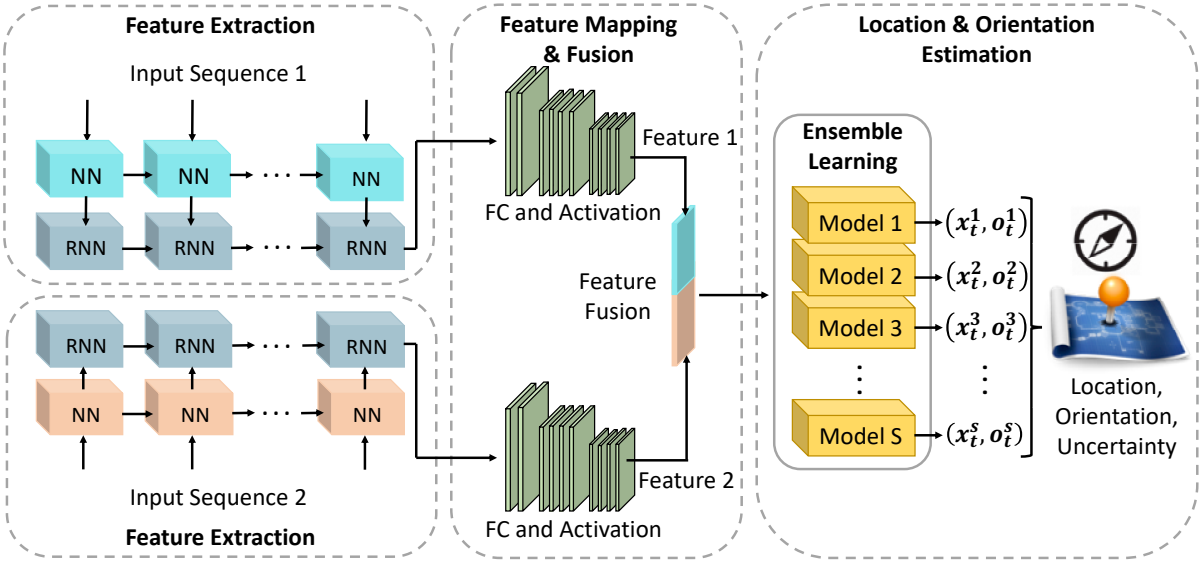


Fig. 2. Framework of DeepNavi, which consists of three parts, feature extraction, feature mapping and fusion and location & orientation estimation.

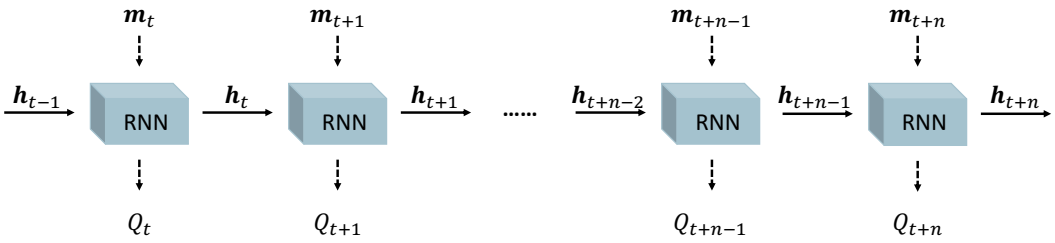


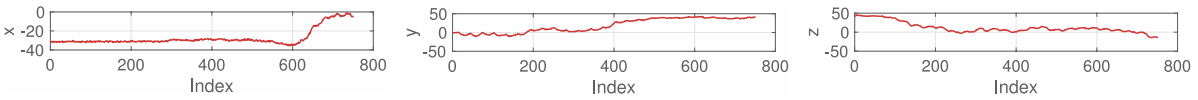
Fig. 3. An illustration of RNN that processes sequential inputs.

different aspects, such as dimensions (images with tens of thousands of pixels v.s. geomagnetic vector with 3 values). To cope with these differences, we use different neural networks (NN) to extract initial features from corresponding inputs. Take images for example, we use CNN to extract features. This is because CNN looks at each patch of an image, processes it with small convolutions and grasps a local regional understanding. Therefore, CNN is better with images. As for other inputs, such as Wi-Fi fingerprints or geomagnetic sequences, we can use RNN to extract features. This is because RNN uses a series of nodes to form a directed graph, which allows it to capture the temporal correlations of input sequences. In addition, we can also process radio fingerprints with RNN. This is because a fingerprint can be viewed as a sequence of ordered received strength values. By using the above networks to extract initial features, we use the RNN architecture to further extract temporal correlations.

As an example, we illustrate the feature extraction through RNN in Figure 3. Let $\mathbf{m}_t$ be the signal readings at time t. Please note that $\mathbf{m}_t$ can be any signal sampled at this time, such as a radio or a geomagnetic fingerprint. Without loss of generality, we use $\mathbf{m}_t$ to denote geomagnetic fingerprint collected at time t.

RIGHTSLINK

(a) Consecutive frames taken while a user is walking (from left to right).



(b) Fluctuating geomagnetism at either ends and center of a long corridor.

Fig. 4. Noisy signals collected in a trial site. Images taken by the surveyor or user could be noisy due to the motion blur. Geomagnetic readings can also be fluctuating with cheap mobile sensors.

The internal state of RNN at time t is denoted by $\mathbf{h}_t$, which is calculated based on previous RNN state $\mathbf{h}_{t-1}$ and current input $\mathbf{m}_t$ as follows:

$$\mathbf{h}_t = \tanh(\mathbf{W}_1\mathbf{m}_t + \mathbf{b}_1 + \mathbf{W}_2\mathbf{h}_{t-1} + \mathbf{b}_2), \tag{1}$$

where $\mathbf{W}_1$ and $\mathbf{W}_2$ are the weights for current input and previous hidden state, respectively. Different from other neural networks, RNN shares weights across layers, which reduces the number of parameters in the network and facilitates fast training. $\mathbf{Q}_t$ is the output corresponding to $\mathbf{m}_t$, which indicates the estimated location and orientation.

After extracting initial features from inputs separately, we then map these features through non-linear mapping layers to a common feature space, which reduces the discrepancy of initial features. Specifically, we insert additional FC layers to generalize these extracted features. We illustrate this in the middle part in Figure 2.

4.2 Joint Pedestrian Navigation and Uncertainty Estimation

Due to the environment and user factors, inputs collected by ordinary users are usually noisy. Figure 4(a) shows several consecutive images taken during the navigation. It shows that images taken by users could be blurry (first two images), which renders feature extraction unreliable. In addition, we have collected geomagnetism at three different positions, i.e., two ends and middle of a corridor (Figure 4(b)). Although we keep the device static at these positions, geomagnetic readings are fluctuating due to sensor noises, leading to large errors. Inspired by Lakshminarayanan et al. [26], we propose to leverage deep ensemble learning in our navigation system for accurate predictions. Specifically, we learn multiple models with our training data and combine them together. By model combination, we are able to smooth out temporal navigation errors and achieve sufficient accuracy. In addition, we leverage MDN and model data distribution with mixture models so as to provide uncertainty estimations simultaneously (right part in Figure 2).

To train a network that determines the location and orientation simultaneously, we define our loss function as follows. Suppose we have N test cases in a mini-batch. The ground truth location and orientation corresponding to n-th test case are $\mathbf{x}_n$ and $\mathbf{o}_n$, respectively. The loss function is defined as follows:

$$\mathcal{L} = \sum_{n=1}^{N}(||\mathbf{x}_n - \hat{\mathbf{x}}_n|| + \alpha||\mathbf{o}_n - \hat{\mathbf{o}}_n||), \tag{2}$$

RIGHTSLINK

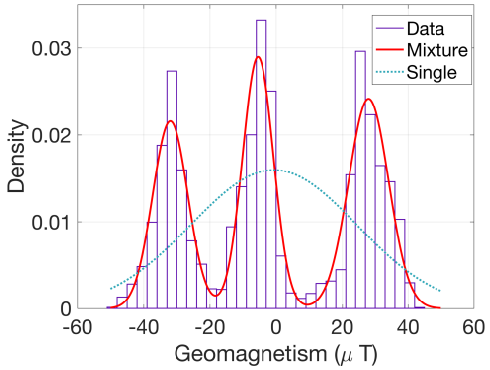


Fig. 5. Geomagnetic data fitting with Gaussian distribution(s). It shows that the fitting is better with mixture distributions.

where $\hat{\mathbf{x}}_n$ and $\hat{\mathbf{o}}_n$ denote corresponding estimated location and orientation, respectively. α is a weight value that balances location and orientation losses. We use $||\cdot||$ to denote L_2 norm.

According to the above loss function, our network determines conditional mean location and orientation based on our training data. Then the probability distribution of our estimation is:

$$p_\theta(\mathbf{k}_n|\mathbf{I}_n, \mathbf{m}_n) = \mathcal{N}(\mu_\theta(\mathbf{I}_n, \mathbf{m}_n), \sigma_\theta(\mathbf{I}_n, \mathbf{m}_n)), \tag{3}$$

where $\mathbf{k}_n = [\hat{\mathbf{x}}_n, \hat{\mathbf{o}}_n]$, which is the concatenation of location and orientation estimations. $\mathbf{I}_n$ and $\mathbf{m}_n$ denote the image and magnetic sequence of test case n. θ denotes the parameters of our network. Denote the predicted mean and variance by $\mu_\theta(\mathbf{I}_n, \mathbf{m}_n)$ and $\sigma_\theta(\mathbf{I}_n, \mathbf{m}_n)$, respectively. For notation simplicity, we omit θ and subscript n in the following discussions. Based on the Gaussian distribution, our objective is to minimize the negative log-likelihood:

$$-\log p(\mathbf{k}|\mathbf{I}, \mathbf{m}) = \frac{\sigma^2(\mathbf{I}, \mathbf{m})}{2} + \frac{(\mathbf{k} - \mu(\mathbf{I}, \mathbf{m}))^2}{2\sigma^2(\mathbf{I}, \mathbf{m})} + c, \tag{4}$$

where c is a constant value. This satisfies the Gaussian distribution with mean μ and variance σ.

However, the assumption of a single Gaussian distribution is restrictive for training and trial data, which reduces its applicability in practical scenarios. Figure 5 shows the distribution of received geomagnetism from the x axis of the magnetometer collected in the office area (Figure 6). The total number of readings is 45, 489. We use MATLAB Gaussian fitting [4] to find a distribution that fits these readings. The fitting with mixture models (consists of three Gaussian components), however, is better with collected data based on differences between predictions and real values. This indicates that using mixture models for data prediction could lead to better accuracy in practical scenarios. Therefore, we replace previous regression layers with MDN for estimation. By leveraging mixture models, the distribution of location and orientation estimations becomes:

$$p(\mathbf{k}|\mathbf{I}, \mathbf{m}) = \sum_{g=1}^{G} \omega_g \mathcal{N}(\mu_g(\mathbf{I}, \mathbf{m}), \sigma_g(\mathbf{I}, \mathbf{m})), \tag{5}$$

[4]https://ww2.mathworks.cn/help/stats/gmdistribution.html

RIGHTSLINK

where G is the number of Gaussian distributions in our mixture model and ω_g is the weight corresponding to g-th ($1 \leq g \leq G$) model. These sum of these weight values of each Gaussian component is 1:

$$\sum_{g=1}^{G} \omega_g = 1. \tag{6}$$

Based on the MDN, we elaborate the estimation of location and orientation with ensembles as follows. Given S models, the predicted location and orientation of a target is:

$$\mu_\star(\mathbf{I}, \mathbf{m}, \mathbf{k}) = S^{-1} \sum_{s}^{S} \sum_{g}^{G} \omega_{s,g}(\mathbf{I}, \mathbf{m}, \mathbf{k}) \mu_{s,g}(\mathbf{I}, \mathbf{m}, \mathbf{k}), \tag{7}$$

while the variance σ of current estimation is:

$$\sigma^2_\star(\mathbf{I}, \mathbf{m}, \mathbf{k}) = S^{-1} \sum_{s}^{S} \sum_{g}^{G} (\omega_{s,g}(\mathbf{I}, \mathbf{m}, \mathbf{k})(\sigma^2_{s,g}(\mathbf{I}, \mathbf{m}, \mathbf{k}) + \mu^2_{s,g}(\mathbf{I}, \mathbf{m}, \mathbf{k}))) - \mu^2_\star(\mathbf{I}, \mathbf{m}, \mathbf{k}). \tag{8}$$

Consequently, we generate more accurate location and orientation estimations. The reasons are as follows. First, we model our data with mixture distribution models rather than with a single Gaussian model. Therefore, our framework is more practical. Second, we generate more diverse models with the ensemble learning technique. Then, we perform model combination by combining multiple models together to make more accurate and powerful one for navigation.

4.3 Discussions on the Benefit of Ensemble Learning

In this section, we theoretically show that ensemble learning boosts the navigation accuracy. We consider a neural network $\hat{f}(\cdot)$ used for regression with bias $\mathbf{B}$ from the ground truth, modeled as:

$$\hat{\mathbf{y}} = \hat{f}(\mathbf{x}) = f(\mathbf{x}) + \mathbf{B}, \tag{9}$$

where the ground truth value is:

$$\mathbf{y} = f(\mathbf{x}). \tag{10}$$

We use $\mathbf{B}$ to denote the overall bias of the prediction model. The bias is classified into two categories: bias with zero mean value and bias with non-zero mean value. We discuss the improvement with ensemble learning with zero mean value and non-zero value separately.

Suppose $\mathbf{B}$ consists of only zero mean value distributions, we have:

$$\hat{\mathbf{y}} = \hat{f}(\mathbf{x}) = f(\mathbf{x}) + \mathbf{b}, \tag{11}$$

where $\mathbf{b}$ is a random variable over randomized models trained on a given $\mathbf{x}$ and $\mathbf{b}$ is independent of $\mathbf{x}$.

Using ensemble learning, the predicted value $\mathbf{y}_e$ is defined as follows:

$$\mathbf{y}_e = S^{-1} \sum_{s=1}^{S} \hat{\mathbf{y}}_s. \tag{12}$$

Suppose $\mathbf{b}$ is distributed with zero mean value, then we have:

$$\mathbf{y}_e = \mathbf{y} + S^{-1} \sum_{s=1}^{S} \mathbf{b}_s. \tag{13}$$

When the number of ensembles S is sufficiently large, $S^{-1} \sum_{s=1}^{S} \mathbf{b}_s$ becomes 0, its mean value. In this case, the final estimation $\mathbf{y}_e$ is close to its true value.

RIGHTSLINK

(a) Office area.

(b) Food plaza.

Fig. 6. Two trial sites in our experiment. (a) A featureless office area. (b) A spacious food plaza.

However, in many cases where the expected mean of $\mathbf{B}$ is a non-zero value with additive bias. Then the regression model in Equation 9 becomes $\hat{\mathbf{y}} = \hat{f}(\mathbf{x}) = f(\mathbf{x}) + \mathbf{b} + \mathbf{d}$, where $\mathbf{d}$ denotes the non-zero bias. Based on the above, the expected error with ensemble learning is:

$$E(\varepsilon_e) = E(S^{-1} \sum_{s=1}^{S} (f(\mathbf{x}_s) + N^{-1} \sum_{i=1}^{N} (\mathbf{b}_{si} + \mathbf{d}_{si}) - \mathbf{y}_s)^2), \tag{14}$$

where N denotes the number of trial cases. The expectation of error without ensemble learning is:

$$E(\varepsilon_{\bar{e}}) = E(S^{-1} \sum_{s=1}^{K} (f(\mathbf{x}_s) + \mathbf{b}_s + \mathbf{d}_s - \mathbf{y}_s)^2). \tag{15}$$

If S is large enough, we can replace $N^{-1} \sum_{i=1}^{N} (\mathbf{b}_{si} + \mathbf{d}_{si})$ with its mean and apply inequality $E(X^2) \geq E^2(X)$:

$$E(\varepsilon_e) \leq E(\varepsilon_{\bar{e}}). \tag{16}$$

This depicts that the ensemble learning reduces the navigation error given sufficient number of models ($E(\varepsilon_e)$ smaller than $E(\varepsilon_{\bar{e}})$). We further experimentally demonstrate the improvement of accuracy with the ensemble learning in Section 5.2.

5 ILLUSTRATIVE EXPERIMENTAL RESULTS

We have implemented the proposed framework and conducted extensive experiments in two different trial sites to evaluate its performance. In this section, we first discuss our experimental settings, including trial sites, comparison schemes and baseline parameters in Section 5.1. Then we illustrate the navigation performance in terms of location and orientation errors in Section 5.2. Finally, we evaluate the system overhead including time and energy consumption in Section 5.3.

5.1 Experimental Settings and Comparison Schemes

We have conducted extensive experiments in two different trial sites, an office area in our university and a food plaza. Covering around $2,800\ m^2$, the office area consists of long corridors, white walls and sparse visual features (Figure 6(a)). In contrast to the office area, the food plaza (covering around $3,500\ m^2$) has many stores, rich features and large open space (Figure 6(b)) with relatively stable geomagnetic readings.

In the survey stage, we walk along designated paths (denoted by red solid lines) that cover popular areas in our trial sites to collect signals. Although the food plaza is spacious, it has many tables and chairs densely distributed

RIGHTSLINK

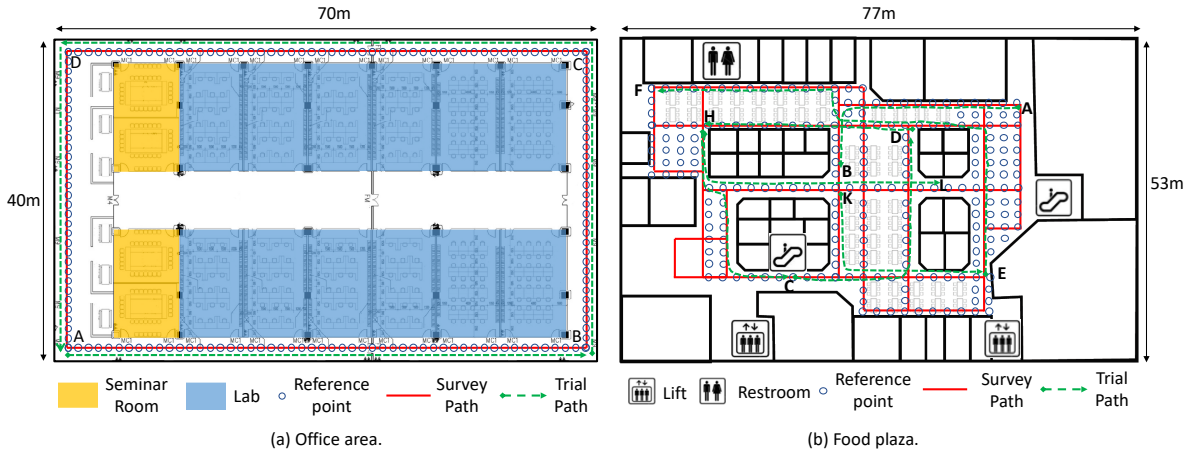


Fig. 7. Floorplans of our trial sites, where red solid lines indicate our survey paths.

across this site. Therefore, we find paths that many customers walk along to collect training data. Figure 7 illustrates floorplans and survey paths in these sites. We collect videos (around 30 frames) and geomagnetic sequences (around 50 readings) per second. Our dataset with images and geomagnetic sequences is available at GitHub [5]. Please note that the data collection with videos and geomagnetic sequences is fast due to their high sampling frequency. Furthermore, we do not have to stop to select landmarks or take independent images. In addition to images and geomagnetic sequences, we have also collected images and Wi-Fi fingerprints in these trials sites to evaluate the navigation error with them. In the survey stage, we collect videos (around 30 frames per second) and Wi-Fi fingerprints along the survey paths. However, as the sampling frequency of Wi-Fi fingerprints is relatively slow (around 1Hz), we design survey grids and stop at these points to collect around 10 fingerprints during survey. To generate our training and evaluation dataset, we align them according to the timestamps when an image, a geomagnetic sequence or a Wi-Fi fingerprint is collected. Then, we pair an image with a geomagnetic sequence or a Wi-Fi fingerprint to generate a training or an evaluation sample.

In our experiment, we take Intel RealSense Camera ZR300 [6] and attach Huawei Mate 9 with it for location and orientation labeling. Intel ZR300 has a stereoscopic depth-sensing camera for distance measurement (from target to the device), a fisheye camera and an accurate inertial measurement unit. We use the robot operating system (ROS) [7] and an optimization-based multi-sensor state-estimator termed VINS-Fusion [8] to process its readings and determine its accurate location and orientation with the inputs from Intel ZR300. Based on location and orientation estimations, we label ground truth locations and orientations of trial devices because they are tied to each other closely and have the same location and orientation values. Being a signal-fusion framework, the main focus of this paper is to fuse different signals together for indoor navigation. Therefore, the estimation of location and orientation is based on input signals. In our experiment, we do not recover user trajectories with motion sensors for attitude detection.

We invite two volunteers to take part in our trials. Both of these volunteers are male students from our school. The heights of these volunteers are around 166cm and 175cm, respectively. During the trial, we ask these

[5]https://github.com/gh835470669/DeepNaviDataset
[6]https://software.intel.com/en-us/realsense/zr300
[7]https://www.ros.org/
[8]https://github.com/HKUST-Aerial-Robotics/VINS-Fusion

RIGHTSLINK

volunteers to hold the phone in upright position so that it has clear view ahead. However, we do not ask users to hold the phone that is exactly perpendicular to the floor. In the office area, where the path is constrained by the corridors (pedestrians usually walk along corridors to find destinations), we ask one volunteer to conduct the navigation trial. The volunteer (175cm) select ends of corridors (denoted by A, B and D) as his starting positions and destinations. This volunteer selects three set of starting positions and destinations (A to B, B to D via C and D to A) and takes these tasks consecutively. Then, our DeepNavi gives instructions and guides the volunteer from each start position to the destination. The trial distance in this site is around 220m with 228 trial cases.

While in the food plaza, which covers more areas, we have two volunteers. Each of them selects several pairs of starting and ending positions (usually they are locations near the popular counters, lifts or escalators). Then, they feed these positions into our application. Based on these information, DeepNavi provides navigation plans and guides users to their destinations from their starting positions. Volunteer 1 chooses three pairs of start positions and destinations: A to B, D to H and E to F. On the other hand, volunteer 2 chooses four pairs: C to D, K to E, H to L, H to C. To sum up, the overall trial distance is around 309m with 520 trial cases in the food plaza. In our trial, we also use the ZR300 to record images, depth readings, inertial readings and calculate current position and orientation with the ROS framework. We use them as ground truth values. In the navigation stage, volunteers collect 1 images, record 50 geomagnetic samples or 1 Wi-Fi fingerprint per second for evaluation. In the food plaza, volunteers take turns in doing the navigation tasks.

Without loss of generality, we use the ResNet [13] to extract visual features from input images due to its strong learning ability and efficiency in training with its residual technique. However, it is also possible to integrate other convolutional networks according to the requirements of specific applications. In the training stage, we first resize images to 256x256 and then center crop the input image of 224x224 pixels as ResNet does in their experiment. While in the test stage, we conduct random crop of images. We remove the last few classification layers and get high-dimensional features from ResNet. Then we use the gated recurrent units (GRU) [5] to extract correlations of sequential inputs, where we feed each individual input to a node and use the output vector as the magnetic location feature. This is because it incorporates the forget gate so as to retain short-term memory of recent inputs and works well with geomagnetic sequences with low dimensions. The number of layers in the GRU is set to 4 in our experiment, with which we are able to extract distinguishing features. Furthermore, it is also possible to integrate other recurrent networks, such as LSTM [17], into our framework and extract features. We train our neural network and conduct evaluations on an Ubuntu Server (14.04) with an NVIDIA GTX 1080 GPU card, an Intel i7-6700 CPU card and 48 GB memory. We implement our network based on PyTorch 1.0.1 [9].

We compare the navigation accuracy with the following state-of-the-art approaches:

- *PoseNet* [24]: PoseNet is an image-based localization system. Using training images and corresponding locations and orientations, they fine-tune GoogLeNet [40] to regress current position and orientation. For fair comparison, we implement PoseNet with ResNet and evaluate accordingly.
- *MaLoc* [45]: MaLoc adopts augmented particle filter by considering the geomagnetic distribution at each position. Based on the distribution, it sets corresponding weights and employs weighted average of particles to track users.
- *Travi-Navi* [52]: Travi-Navi is a fusion navigation system, which collects both geomagnetism and Wi-Fi and fuses them with particle filters to navigate users continuously. In our experiment, we use geomagnetic sequences for localization.

To evaluate the generality of the proposed DeepNavi with other signal combinations, we conduct another trial with images and Wi-Fi fingerprints. We introduce the baseline comparison schemes as follows:

[9]https://pytorch.org/

RIGHTSLINK

- DeepNavi-Wi: DeepNavi-Wi is a baseline scheme that navigates users with images and Wi-Fi fingerprints. It uses the same network framework as DeepNavi, where Wi-Fi fingerprints are viewed as a sequence of received values and processed by RNN.
- Wi-Fi: To evaluate the improvement over pure Wi-Fi, we remove the CNN module of DeepNavi-Wi so it estimates current location with a single Wi-Fi access point. Each Wi-Fi fingerprint is viewed as a signal sequence and is processed by recurrent networks.

We have conducted extensive experiments to evaluate the performance in our trial sites. In the office area, we select 10 stable access points that have large coverage and build a W-Fi fingerprint database. While in the shopping mall, we have 50 access points deployed by shop owners. The reason we use more access points in the food plaza is that it is comparatively larger. In the survey stage, we collect fingerprints at 164 reference points (RPs) as RADAR [1] does, which are vectors of received signal strength indicators collected at known locations. While in the food plaza, we collect fingerprints at 221 RPs. At each RP, we collect 10 fingerprints and use their average received signal strength values to reduce the impact of signal fluctuations. The distance between two RPs in the office area is around 1.2m while that in the food plaza is around 1m. As neural networks usually require a large amount of training data, we augment the Wi-Fi dataset by linear interpolation between two adjacent RPs. This is because they are close to each other. Specifically, we insert two fingerprints between two RPs. After interpolation, we associate an image with a Wi-Fi fingerprint based on locations where they are collected and construct our training dataset.

The performance metrics are defined as follows:

- *Location error*: In the n-th trial, the ground truth location of a user is $\mathbf{x}_n$ while estimated one is $\hat{\mathbf{x}}_n$. We define the mean positioning error with Euclidean distance between ground truth locations and estimated ones as follows:

$$\mathcal{E}_l = N^{-1} \sum_{n=1}^{N} ||\mathbf{x}_n - \hat{\mathbf{x}}_n||, \tag{17}$$

  where $||\cdot||$ denotes L_2 norm.
- *Orientation error*: In the n-th trial, the ground truth orientation of a user is $\mathbf{o}_n$ while estimated one is $\hat{\mathbf{o}}_n$. We evaluate the orientation error with angular distances between ground truth and estimated ones as follows:

$$\mathcal{E}_o = N^{-1} \sum_{n=1}^{N} \arccos(\mathbf{o}_n, \hat{\mathbf{o}}_n). \tag{18}$$

- *Time and power consumption*: To evaluate the power consumption, we develop an Android application that collects and sends images and magnetic readings (or images and Wi-Fi fingerprints) to a remote server through Wi-Fi continuously as in [2]. Then we determine the power drop based on percentage of power displayed in the status bar before and after the evaluation. Then, we kill all other applications and measure the power drop within a same amount of time. By subtracting these two values, we get the overall power consumption of the application. Finally, we determine the average energy consumption of each trial by dividing the overall power consumption by the total number of trial cases. As for time consumption, we measure the overall time of all trials and divide it by the trial number to get the average time consumption for each trial.

We compare the orientation error with PoseNet, DeepNavi and DeepNavi-Wi. As MaLoc and Travi-Navi do not give orientation estimations in their paper, we do not compare the orientation with them.

We present our baseline parameters in this paper in Table 2. The training sample is defined as a pair of input signals for training. Similarly, an evaluation sample is a pair of input signals for evaluation. Take images and geomagnetic sequences for example, we first align them based on the timestamps of collection. Then, we pair an

RIGHTSLINK

Table 2. Baseline parameters in our experiments.

Parameter	Values	
	Food Plaza	Office
Training samples (image+magnetism)	24,668	13,824
Evaluation samples (image+magnetism)	520	228
Training samples (image+Wi-Fi)	470	490
Evaluation samples (image+Wi-Fi)	149	163
Number of images	1	1
Geomagnetism length	16	16
Epoch	300	300
Ensemble	3	3

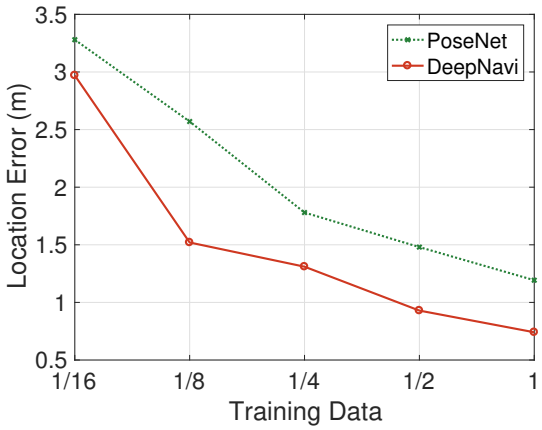


Fig. 8. Location error v.s. training samples (office area).

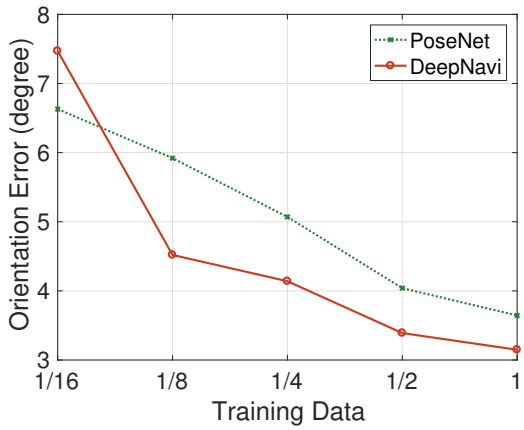


Fig. 9. Orientation error v.s. training samples (office area).

image and a geomagnetic sequence with 16 readings, where the image and the last reading of geomagnetism are collected at the same time. After experimental evaluations with α, we set its value to 1 to achieve better estimation accuracy.

5.2 Illustrative Navigation Results

In this section, we present illustrative navigation results in our trials.

Figure 8 illustrates the location error with different portions of training data. It shows that the mean location error decreases with more training data. This is because the network learns more robust features with more training data, which is helpful for increasing accuracy. The decrease in navigation error slows down with more than 1/8 of training data. This is because we have sufficient location information from the training data. With more than 1/2 of training data, we are able to achieve sufficient location accuracy. However, the time consumption for network training increases with more training data. Therefore, it is possible to train navigation network with 1/2 of data to achieve trade-off between location errors and training time.

Figure 9 presents the orientation error with different portions of training data. It shows similar trend as that of the location error. With more training data, the orientation error decreases. This is because our network model learns more distinguishing features from more training data, thus reducing the orientation error significantly.

RIGHTSLINK

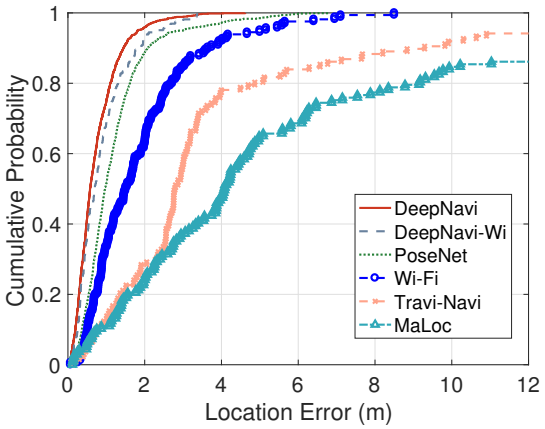


Fig. 10. CDF of location error (office area).

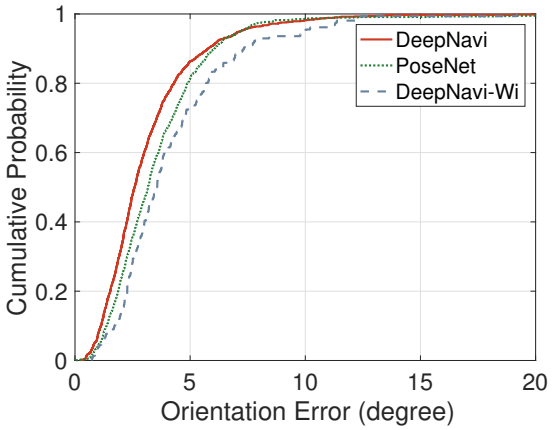


Fig. 11. CDF of orientation error (office area).

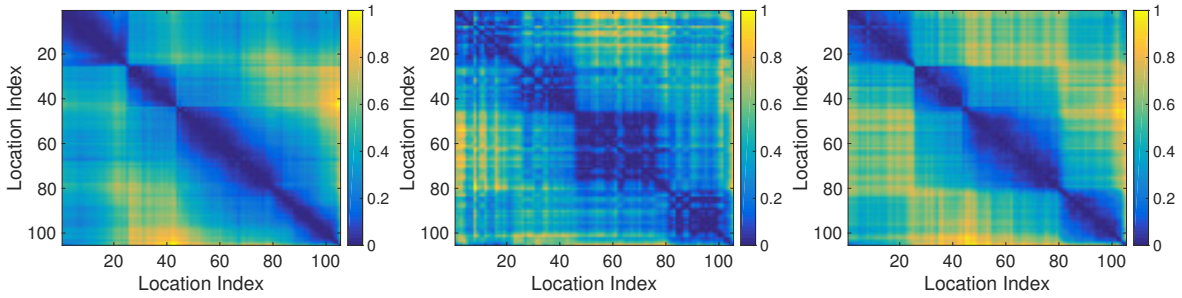
As the portion of training data is larger than 1/8, the decrease in orientation error slows down. This is because our model has learned sufficient information from the training data. With even more data, the orientation error remains relatively stable. Therefore, it is practical to train the navigation model with around 1/2 of training and achieve sufficient accuracy.

Figure 10 shows the cumulative localization error in the office area. Although this site contains repetitive structures (e.g., doors, windows), which are usually challenging for traditional visual feature point-based approaches (e.g., scale-invariant feature transform, speeded-up robust features), experimental results show that the proposed DeepNavi and DeepNavi-Wi are able to achieve sufficient accuracy. The reasons are as follows. First, by fusing visual images with magnetic sequences or Wi-Fi fingerprints, we are able to generate more distinguishing features in sites with sparse visual textures for accurate navigation. Second, we leverage deep neural networks to find signal correlations between signals and extract joint features for navigation. Third, we fuse predictions from multiple models for navigation, which reduces location errors significantly.

Figure 11 presents the orientation error in the office area. It demonstrates that our fusion network achieves comparable accuracy in featureless office area with images and geomagnetism. In addition to images and geomagnetic sequences, we have also evaluated the orientation error with images and Wi-Fi. Experimental results show that the mean orientation errors are similar: DeepNavi (3.1 degrees), DeepNavi-Wi (4.2 degrees) and PoseNet (3.6 degrees). The reason that the errors are similar is that we estimate current orientation based on images. Other signals, such as Wi-Fi fingerprints and geomagnetic sequences, do not provide such information themselves. Therefore, adding them does not help reduce the orientation error. Consequently, the mean orientation errors of them are similar. In many indoor sites, corridors are usually orthogonal to each other (relative angle is around 90 degrees). Our mean orientation errors are significantly smaller. Therefore, the errors in orientation do not deteriorate the quality of navigation services.

To demonstrate the effectiveness of feature fusion, we collect images and geomagnetic readings at 105 positions uniformly distributed in the office area. We present difference matrices of visual, geomagnetic and fused features in Figure 12(a), Figure 12(b) and Figure 12(c), respectively (Blue indicates small differences and yellow indicates large differences). It shows that features extracted at distant locations (upper right and lower left areas) can be similar, which leads to large navigation errors. With feature fusion, we are able to increase the distinctiveness of features and enlarge differences between distant positions (many blue areas are turned into yellow or green

RIGHTSLINK



(a) Image. (b) Geomagnetism. (c) Fusion.

Fig. 12. Feature differences between locations uniformly selected in an office area. (a) Feature differences with images. (b) Feature differences with geomagnetism. (c) Feature differences with fusion.

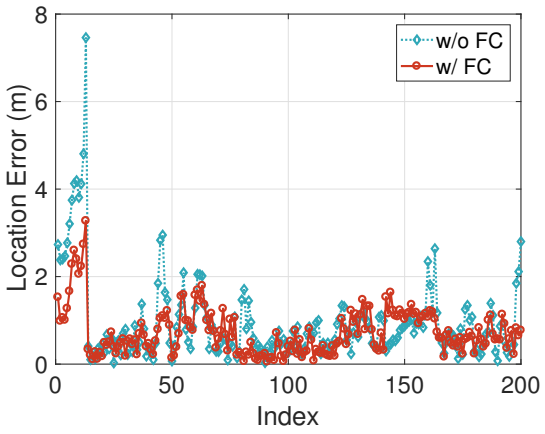


Fig. 13. Location error with (w/) and without (w/o) FC layers (office area).

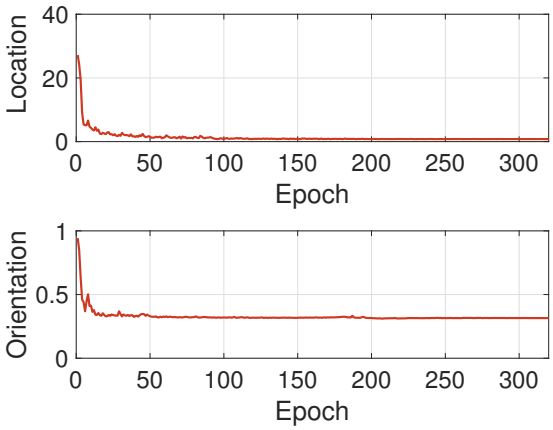


Fig. 14. Navigation loss v.s. Epoch (office area).

in the upper right and lower left areas in Figure 12(c)). This proves that our framework is able to enhance the distinctiveness of features.

Figure 13 shows consecutive location errors. It shows that by inserting FC layers to process diverse features after initial feature extraction, we are able to reduce location errors. This is because raw features extracted from different signals are significantly different from each other. By inserting additional FC layers, we are able to map these raw features to a common feature space through non-linear mapping. Combined with the learning capacity of recent neural networks, our network fuses these signals effectively, thus increasing the accuracy.

Figure 14 presents the navigation loss with different numbers of training epochs. It shows that the navigation error (in terms of location and orientation error) decreases with more epochs. This is because our model can fit our data better with more iterations. As the number of epochs grows larger, navigation error decreases more slowly. This is because our model has learned sufficient information from training data. Finally, the loss converges after 300 epochs. It takes more time to train a network with more epochs. To achieve trade-off between navigation accuracy and training time, we train the model for 300 epochs.

RIGHTSLINK

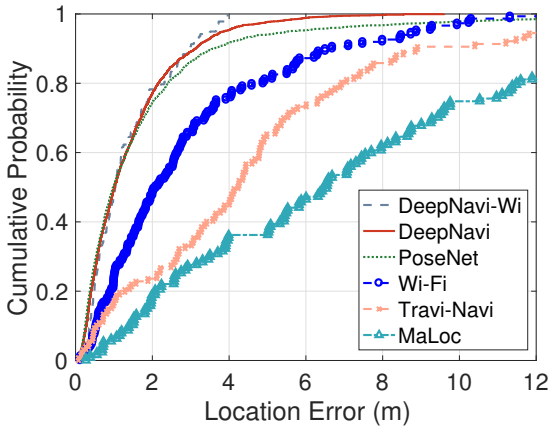


Fig. 15. CDF of location error (food plaza).

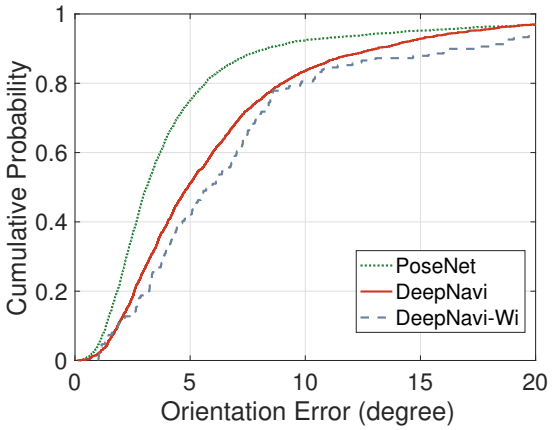


Fig. 16. CDF of orientation error (food plaza).

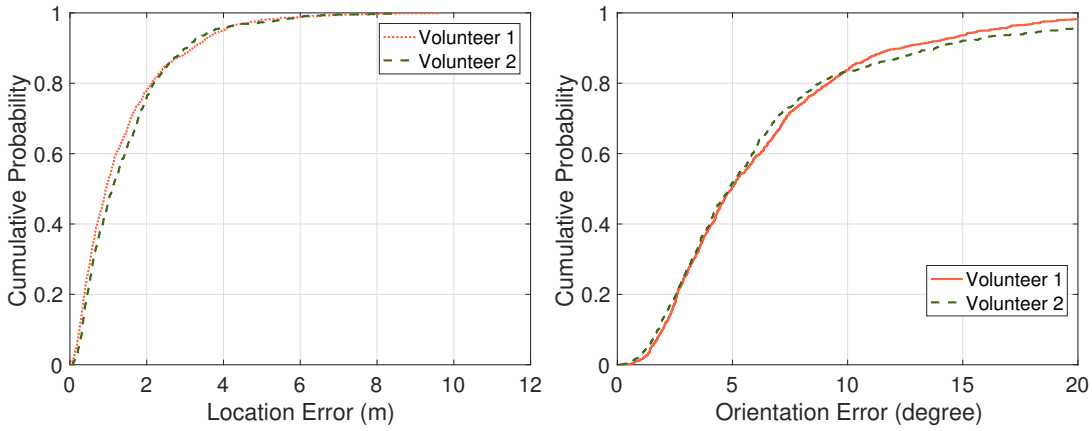
We compare the location error in Figure 15. It shows that the proposed DeepNavi and DeepNavi-Wi are able to achieve sufficient accuracy in the food plaza with large open areas. The trial site is challenging for geomagnetic localization approaches, mainly due to magnetic ambiguities in open areas. On the other hand, vision-based navigation approaches (PoseNet) can achieve sufficient accuracy in the food plaza with rich visual textures, e.g., store logos. However, due to environment and user factors, such as similar textures of distant store logos, strong illumination and motion blur, the location error can be large with noisy features. By incorporating complementary signals, such as images and geomagnetism, DeepNavi extracts more distinguishing location features, thus increasing the location accuracy. In addition to geomagnetism, we have also evaluated the location error with images and Wi-Fi fingerprints. It shows that we are able to achieve sufficient accuracy with images and Wi-Fi (DeepNavi-Wi) as well. This proves that our approach is general enough to other signal combinations.

Figure 16 presents the orientation error in the food plaza. It shows that the mean orientation errors of DeepNavi (7.5 degrees) and DeepNavi-Wi (7.4 degrees) are comparable to that of PoseNet (5.6 degrees). This is because in spacious food plaza, visual features are able to provide sufficient orientation information, thus the error remains stable in the food plaza. The relative angles between two corridors indoors are usually 90 degrees (orthogonal to each other). Compared with the relative angle, the orientation errors are small. Therefore, the difference in angular error does not degenerate the quality of navigation services.

We show the CDF of location and orientation errors of two volunteers with DeepNavi in Figure 17(a) and Figure 17(b), respectively. It shows that the location and orientation errors with two volunteers in our trial sites are similar. This is because we use images and geomagnetic signals for navigation, which does not rely on accurate measurement of stride lengths of different users. Furthermore, the images taken by two volunteers are similar, leading to comparable accuracy. Therefore, our approach achieves stable navigation performance with different volunteers.

Figure 18 shows the uncertainty estimation with different location errors. It shows that as the location error increases, uncertainty value increases accordingly. This demonstrates that by inserting MDN into our framework, the proposed network learns uncertainty values from training data. Therefore, our fusion network not only gives navigation estimations, but also uncertainty values as well. Based on these values, our framework can be easily extended to leader-follower navigation systems, where a follower is required to follow the designated path of a leader. Based on the changes in uncertainty values (larger than 0.05), the follower could be 2m away from the

RIGHTSLINK



(a) CDF of location error.

(b) CDF of orientation error.

Fig. 17. Location and orientation errors using images and geomagnetic readings with different volunteers (food plaza).

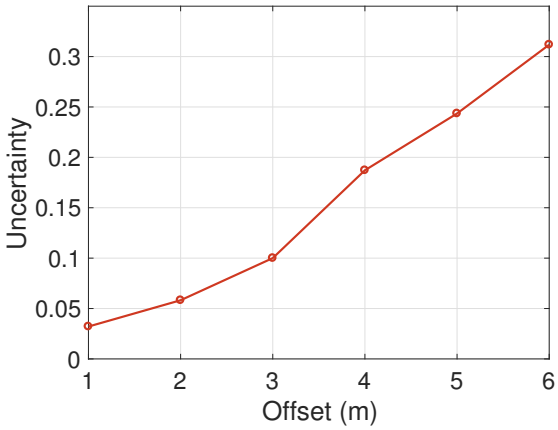


Fig. 18. Uncertainty v.s. offset (food plaza).

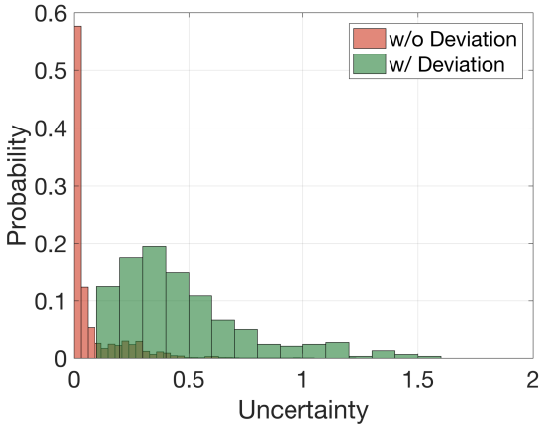


Fig. 19. Distributions of uncertainty with (w/) and without (w/o) unseen data (food plaza).

designated path. In case of deviations, it is possible to deploy some techniques, such as vibrations or toasts, to alert users promptly.

Figure 19 illustrates distributions of uncertainty values with mixture density networks and ensembles in leader and follower mode. It shows that in the case with small deviations (say, fewer than 10m), the uncertainty values are usually small and are distributed unevenly. In these scenarios, DeepNavi is confident about navigation results. However, if a follower is significantly away from designated path, uncertainty value grows larger and more scattered. Based on this, we are confident that a follower is following a path if uncertainty value is smaller than 0.05. Otherwise, the follower is very likely to deviate from path if uncertainty value is significantly larger than 0.1.

Figure 20 presents the cumulative location error with (w/) and without (w/o) ensemble training. It shows that the accuracy without ensembles has longer tails than that with ensembles, indicating larger errors. The reasons

RIGHTSLINK

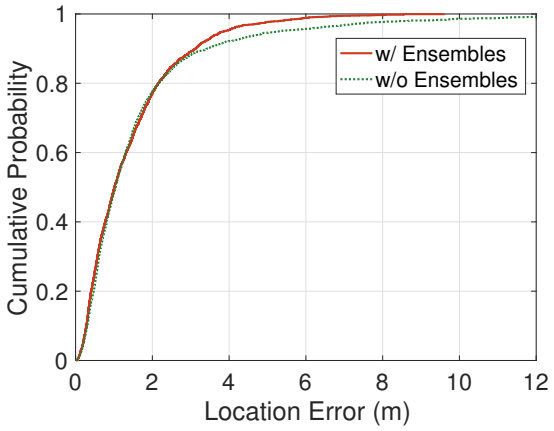


Fig. 20. Comparison of location error during navigation with (w/) and without (w/o) ensembles (food plaza).

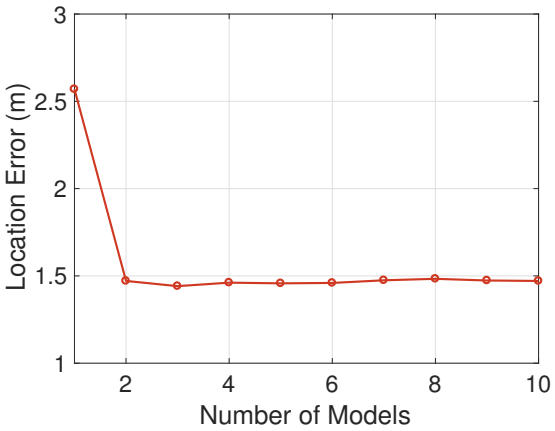


Fig. 21. Location Error v.s. Number of Models (food plaza).

are as follows. In the online navigation stage, the location error could be large due to noisy input signals, e.g., blurry images, noisy geomagnetism readings. In our DeepNavi, we introduce the ensemble learning, which learns multiple models from training data. Then in the online stage, it provides more diverse predictions with multiple models. By combining them together, we are able to smooth erroneous predictions, thus giving more accurate location estimations with these noisy inputs.

Figure 21 presents the mean location error with different numbers of models during navigation. It demonstrates that by incorporating more models, we are able to reduce the location error by a large margin. However, as the number of models increases, the decrease in translation error slows down. This is mainly because we extract sufficient location clues. The introduction of more than three models does not add more additional location clues in our trial site. Therefore, translation errors do not change much. However, it takes more time to train more models. To achieve trade-off between training time and accuracy, we train three models in our experiment.

5.3 System Deployment Overhead

In this section, we evaluate the time, network and energy consumption for navigation using Huawei Mate 9. It takes fewer than 30 minutes to survey each of these sites with videos and geomagnetic sequences. The time consumption of data collection is low compared with image taking of all landmarks in this site (around 2 to 3 hours). This is because we just need to initiate a data collection program, which records videos and geomagnetism automatically without any user participation. In the offline stage, it takes around 70 minutes to train a neural network in our server with an Nvidia 1080 TI graphics card. To sum up, it takes around 3.5 hours to train 3 models with ensemble learning. Please note that the training is conducted offline. Therefore it does not incur additional time of online navigation.

To evaluate the time consumption of a navigation query, we send an image (640x480, around 450KB) and 16 geomagnetic samples (less than 2KB in double precision format) to a server continuously. Theoretically, the overhead of network transmission is around 0.036s with a 100Mbps Wi-Fi router. In the navigation stage, it takes around 0.015s to process one image and a geomagnetism sequence in our server. After estimating the location and orientation, our server sends them back to the client. To sum up, the overall time consumption of a navigation query should be around than 0.051s (=0.036+0.015). Therefore, it is possible to process more than 19 (=1 ÷ 0.051) navigation queries in a second. To evaluate the power consumption in reality, we implement an

RIGHTSLINK

Android application that records images, geomagnetic readings or Wi-Fi fingerprints continuously and sends them to a server through Wi-Fi. Before our trial, we kill all other background applications. Then, we measure signal readings (e.g., an image, a geomagnetic sequence or a Wi-Fi fingerprint), record them and send them to the server through Wi-Fi network. Without specific optimization, the sampling frequencies of images, geomagnetism and Wi-Fi are 1.2Hz, 50Hz and 1Hz, respectively. Limited by the sampling rate, our framework gives around 1 navigation instructions per second. However, with higher sampling frequency, larger bandwidth and streaming technique, it is possible to give more navigation instructions.

We evaluate the power consumption w/ and w/o our application running in the foreground separately. To evaluate the power consumption of image taking and uploading, our application takes and sends 3150 images to a server continuously. During the evaluation, the power drop is around 364mAh (calculated based on the capacity of the battery and the drop in percentage). Then, we measure the power drop with the same amount of time without running any applications (around 3mAh). The total power consumption of image taking and sending is around 361mAh =(364mAh - 3mAh). The average power consumption of taking an image and sending it to a remote server through off-the-shelf Wi-Fi router is around 0.11mAh (=363 ÷ 3150) in our smartphone.

As for magnetometer, we collect and send 81,833 samples to server (sampling frequency is around 50HZ). The total power consumption of collecting geomagnetic readings and sending them to a server through Wi-Fi network is around 286mAh. Then, we evaluate the power drop without any applications (around 1mAh). Then, the total power consumption with geomagnetic data collection and sending is around 285mAh (=286 - 1). Therefore, the average power consumption of collecting and sending a magnetic fingerprint (three float values corresponding to the readings of x, y and z axis from the magnetometer) is 0.003mAh (=285 ÷ 81833). For a single navigation query (sending 1 frame and 16 magnetic readings to the server), the power consumption is around 0.158mAh (=0.11+0.003 × 16) in our smartphone, which is negligible compared with the capacity of state-of-the-art smartphones [10].

In addition to geomagnetism, we have also evaluated the power consumption with images and Wi-Fi fingerprints. The trial application scans Wi-Fi once per second and sends them to a server. Overall, we have 2425 scans and the total power consumption is around 388mAh. Therefore, the power consumption of scanning nearby Wi-Fi networks and sending a fingerprint to a server is around 0.16mAh (=388 ÷ 2425). Adding the power consumption of collecting an image and a Wi-Fi fingerprint, the total power consumption of a navigation query is around 0.27mAh = (0.16+0.11), which is higher than navigation with images and geomagnetism.

6 DISCUSSIONS

In this paper, we mainly focus on signal fusion and accurate indoor navigation with ensemble learning. Extensive experimental results demonstrate the effectiveness of our proposed approach. Although we have addressed the essential problem of signal fusion and have achieved accuracy, a few more problems remain to be addressed (which are not the focus of this paper).

Scalability to multi-floor environment. In this stage, DeepNavi achieves high accuracy in a single floor. With the emergence of grand shopping malls and tall buildings, it is also necessary to provide multi-story indoor navigation. One of the key challenges in the deployment in multi-floor environment is the detection of floor or elevation changes. Currently, DeepNavi mainly focuses on signal fusion for indoor navigation.

Adaptability to environmental changes. In the complicated indoor environment, it is likely that the configuration may change constantly due to renovation. In this case, visual appearances and geomagnetism may change drastically, rendering the previously-trained model inapplicable. In the future, we will study a self-adaptable navigation system that updates itself according to environmental changes. However, it is challenging to design such systems, mainly due to the difficulty of alteration detection and network adaptation.

[10] https://consumer.huawei.com/en/phones/mate20/specs/

RIGHTSLINK

Location privacy of users. Currently, we send user-collected images and geomagnetism to a remote server for navigation. Although this offloads computational burden to a remote server, it may lead to privacy concerns from users.

7 CONCLUSION

As an essential part of indoor location-based services (LBS), the accuracy of navigation has a direct impact on the quality of LBS. However, indoor sites are often characterized as complicated and highly dynamic, which could lead to degraded accuracy. Prior arts leverage signal fusion for navigation, but they are either specific to signals, subject to navigation modes or constrained by the environment.

To address the above, we propose a unified fusion framework for indoor navigation. Based on the recurrent architecture, our model is general to both spot-based and sequence-based navigation. Depending on the inputs, we use convoutional or recurrent networks to extract initial features. As these inputs vary from each other significantly in resolution and sampling frequency, we insert additional fully connected layers to map extracted features to a common feature space for effective feature fusion. To reduce the impact of noisy signals on location and orientation estimations, we leverage ensemble learning to learn multiple models using our training data. By combining them together, we get fine-grained estimations. As original ensemble learning is restrictive on distributions, we further incorporate mixture density networks to generalize our models with mixture of distributions. DeepNavi is a general framework that processes various inputs, including Wi-Fi, Bluetooth, visible light, for both spot-based and sequence-based indoor navigation. As an example, we study indoor navigation with images and geomagnetic sequence as well as images and Wi-Fi fingerprints. Extensive experiments in two trial sites demonstrate that the proposed framework achieves more than 20% improvement on localization accuracy and provides real-time navigation services.

Besides more evaluation, there are three directions that worth investigation for a more systematic exploration of the design space. In order to extend the deployability to multi-floor environment, one can study the motion patterns of users with floor transitions, such as taking the escalators and climbing the stairs. In terms of adaptability to environmental changes, it is possible to study the relationship between environmental changes and uncertainty values. Afterwards, we can also study network adaptation with collected data by semi-supervised learning. By studying model optimization [47, 48, 50], it is also possible to deploy neural networks in mobile devices to reduce privacy concerns.

ACKNOWLEDGMENTS

This work is supported, in part, by Guangzhou Science Technology and Innovation Commission (GZSTI16EG14/201704030079), Guangxi Special Fund Project for Innovation-Driven Development (Grant No. AA18118039), and Opening Project of Guangdong Province Key Laboratory of Information Security Technology (Grant No. 2017B030314131).

REFERENCES

[1] Paramvir Bahl and Venkata N. Padmanabhan. 2000. RADAR: An in-building RF-based user location and tracking system. In *Proc. IEEE INFOCOM*, Vol. 2. IEEE, 775–784.

[2] Rajesh Krishna Balan, Youngki Lee, Tan Kiat Wee, and Archan Misra. 2014. The challenge of continuous mobile context sensing. In *Proc. IEEE COMSNETS*. IEEE, 1–8.

[3] Christopher. M Bishop. 1994. Mixture Density Network.

[4] Chao Cai, Menglan Hu, Doudou Cao, Xiaoqiang Ma, Qingxia Li, and Jiangchuan Liu. 2019. Self-deployable indoor localization with acoustic-enabled IoT devices exploiting participatory sensing. *IEEE Internet of Things Journal* 6, 3 (June 2019), 5297–5311.

[5] Junyoung Chung, Caglar Gulcehre, Kyunghyun Cho, and Yoshua Bengio. 2015. Gated feedback recurrent neural networks. In *Proc. ICML*. ICML, 2067–2075.

RIGHTSLINK

[6] Ronald Clark, Sen Wang, Andrew Markham, Niki Trigoni, and Hongkai Wen. 2017. VidLoc: A deep spatio-temporal model for 6-DoF video-clip relocalization. In *Proc. IEEE CVPR*. IEEE, 6856–6864.
[7] Jiang Dong, Marius Noreikis, Yu Xiao, and Antti Ylä-Jääski. 2019. ViNav: A vision-based indoor navigation system for smartphones. *IEEE Transactions on Mobile Computing* 18, 6 (June 2019), 1461–1475.
[8] Rizanne Elbakly, Moustafa Elhamshary, and Moustafa Youssef. 2018. HyRise: A robust and ubiquitous multi-sensor fusion-based floor localization system. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 3 (Sept. 2018), 104:1–104:23.
[9] Rita Francese, Michele Risi, Riccardo Siani, and Genoveffa Tortora. 2018. Augmented treasure hunting generator for edutainment. In *Proc. IEEE IV*. IEEE, 524–529.
[10] Cole Gleason, Dragan Ahmetovic, Saiph Savage, Carlos Toxtli, Carl Posthuma, Chieko Asakawa, Kris M. Kitani, and Jeffrey P. Bigham. 2018. Crowdsourcing the installation and maintenance of indoor localization infrastructure to support blind navigation. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 1, Article 9 (March 2018), 25 pages.
[11] Wei Gong and Jiangchuan Liu. 2018. SiFi: Pushing the limit of time-based WiFi localization using a single commodity access point. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 1, Article 10 (March 2018), 21 pages.
[12] Richard Hartley and Andrew Zisserman. 2003. *Multiple view Geometry in Computer Vision.* Cambridge University Press, Cambridge.
[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proc. IEEE CVPR*. IEEE, 770–778.
[14] Suining He, S.-H. Gary Chan, Lei Yu, and Ning Liu. 2018. SLAC: Calibration-free pedometer-fingerprint fusion for indoor localization. *IEEE Transactions on Mobile Computing* 17, 5 (May 2018), 1176–1189.
[15] Suining He, S.-H. Gary Chan, Lei Yu, and Ning Liu. 2019. Maxlifd: Joint maximum likelihood localization fusing fingerprints and mutual distances. *IEEE Transactions on Mobile Computing* 18, 3 (March 2019), 602–617.
[16] Suining He and Kang G. Shin. 2018. Geomagnetism for smartphone-based indoor localization: Challenges, advances, and comparisons. *ACM Comput. Surv.* 50, 6, Article 97 (Jan. 2018), 37 pages.
[17] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation* 9 (1997), 1735–1780.
[18] Yiqing Hu, Yan Xiong, Wenchao Huang, Xiang-Yang Li, Panlong Yang, Yanan Zhang, and Xufei Mao. 2018. Lightitude: Indoor positioning using uneven light intensity distribution. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 2 (July 2018), 67:1–67:25.
[19] Zengshi Huang, Naijie Gu, Jianlin Hao, and Jie Shen. 2018. 3DLoc: 3D features for accurate indoor positioning. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 1, 4, Article 141 (Jan. 2018), 26 pages.
[20] Beakcheol Jang and Hyunjung Kim. 2019. Indoor positioning technologies without offline fingerprinting map: A survey. *IEEE Communications Surveys & Tutorials* 21, 1 (Firstquarter 2019), 508–525.
[21] Ho Jun Jang, Jae Min Shin, and Lynn Choi. 2017. Geomagnetic field based indoor localization using recurrent neural networks. In *Proc. IEEE GLOBECOM*. IEEE, 1–6.
[22] Hernisa Kacorri, Eshed Ohn-Bar, Kris M. Kitani, and Chieko Asakawa. 2018. Environmental factors in indoor navigation based on real-world trajectories of blind users. In *Proc. ACM CHI*. ACM, 56:1–56:12.
[23] Alex Kendall and Roberto Cipolla. 2016. Modelling uncertainty in deep learning for camera relocalization. In *Proc. IEEE ICRA*. IEEE, 4762–4769.
[24] Alex Kendall, Matthew Grimes, and Roberto Cipolla. 2015. PoseNet: A convolutional network for real-time 6-DOF camera relocalization. In *Proc. IEEE ICCV*. IEEE, 2938–2946.
[25] Tae Hyun Kim, Seungjun Nah, and Kyoung Mu Lee. 2018. Dynamic video deblurring using a locally adaptive blur model. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 40, 10 (2018), 2374–2387.
[26] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. 2017. Simple and scalable predictive uncertainty estimation using deep ensembles. In *Proc. NIPS*. Curran Associates, Inc., 6402–6413.
[27] Christos Laoudias, Adriano Moreira, Sunwoo Kim, Sangwoo Lee, Lauri Wirola, and Carlo Fischione. 2018. A survey of enabling technologies for network localization, tracking, and navigation. *IEEE Communications Surveys & Tutorials* 20, 4 (Fourthquarter 2018), 3607–3644.
[28] Bing Li, Juan Pablo Mũnoz, Xuejian Rong, Qingtian Chen, Jizhong Xiao, Yingli Tian, Aries Arditi, and Mohammed Yousuf. 2019. Vision-based mobile indoor assistive navigation aid for blind people. *IEEE Transactions on Mobile Computing* 18, 3 (March 2019), 702–714.
[29] Liyuan Li, Qianli Xu, Vijay Chandrasekhar, Joo-Hwee Lim, Cheston Tan, and Michal Akira Mukawa. 2017. A wearable virtual usher for vision-based cognitive indoor navigation. *IEEE Transactions on Cybernetics* 47, 4 (April 2017), 841–854.
[30] Mingkuan Li, Ning Liu, Qun Niu, Chang Liu, S.-H. Gary Chan, and Chengying Gao. 2018. SweepLoc: Automatic video-based indoor localization by camera sweeping. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 3, Article 120 (Sept. 2018), 25 pages.
[31] You Li, Zhe He, Zhouzheng Gao, Yuan Zhuang, Chuang Shi, and Naser El-Sheimy. 2019. Towards robust crowdsourcing-based localization: A fingerprinting accuracy indicator enhanced wireless/magnetic/inertial integration approach. *IEEE Internet of Things Journal* 6, 2 (April 2019), 3585–3600.
[32] Wenping Liu, Hongbo Jiang, Hongbo Jiang, Jiangchuan Liu, Xiaoqiang Ma, Yufu Jia, and Fu Xiao. 2019. Indoor navigation with virtual graph representation: Exploiting peak intensities of unmodulated luminaries. *IEEE/ACM Transactions on Networking* 27, 1 (Feb 2019),

RIGHTSLINK

187–200.
[33] Masayuki Murata, Dragan Ahmetovic, Daisuke Sato, Hironobu Takagi, Kris M. Kitani, and Chieko Asakawa. 2018. Smartphone-based indoor localization for blind navigation across building complexes. In *Proc. IEEE PerCom*. IEEE, 1–10.
[34] Qun Niu, Mingkuan Li, Suining He, Chengying Gao, S. H. Gary Chan, and Xiaonan Luo. 2019. Resource-efficient and automated image-based indoor localization. *ACM Trans. Sen. Netw.* 15, 2, Article 19 (April 2019), 31 pages.
[35] Qun Niu, Ying Nie, Suining He, Ning Liu, and Xiaonan Luo. 2018. RecNet: A convolutional network for efficient radiomap reconstruction. In *Proc. IEEE ICC*. IEEE, 1–7.
[36] Eshed Ohn-Bar, João Guerreiro, Kris Kitani, and Chieko Asakawa. 2018. Variability in reactions to instructional guidance during smartphone-based assisted navigation of blind users. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.* 2, 3 (Sept. 2018), 131:1–131:25.
[37] Kun Qian, Chenshu Wu, Yi Zhang, Guidong Zhang, Zheng Yang, and Yunhao Liu. 2018. Widar2.0: Passive human tracking with a single Wi-Fi link. In *Proc. ACM MobiSys*. ACM, 350–361.
[38] Quentin Roy, Simon T. Perrault, Shengdong Zhao, Richard C. Davis, Anuroop Pattena Vaniyar, Velko Vechev, Youngki Lee, and Archan Misra. 2017. Follow-my-lead: Intuitive indoor path creation and navigation using interactive videos. In *Proc. ACM CHI*. ACM, 5703–5715.
[39] Yuanchao Shu, Cheng Bo, Guobin Shen, Chunshui Zhao, Liqun Li, and Feng Zhao. 2015. Magicol: Indoor localization using pervasive magnetic field and opportunistic WiFi sensing. *IEEE Journal on Selected Areas in Communications* 33, 7 (July 2015), 1443–1457.
[40] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proc. IEEE CVPR*. IEEE, 1–9.
[41] Xiaoqiang Teng, Deke Guo, Yulan Guo, Xiaolei Zhou, and Zhong Liu. 2019. CloudNavi: Toward ubiquitous indoor navigation service with 3D point clouds. *ACM Trans. Sen. Netw.* 15, 1, Article 1 (Jan. 2019), 28 pages.
[42] Xiaohua Tian, Mei Wang, Wenxin Li, Binyao Jiang, Dong Xu, Xinbing Wang, and Jun Xu. 2018. Improve accuracy of fingerprinting localization with temporal correlation of the RSS. *IEEE Transactions on Mobile Computing* 17, 1 (Jan 2018), 113–126.
[43] Florian Walch, Caner Hazirbas, Laura Leal-Taixé, Torsten Sattler, Sebastian Hilsenbeck, and Daniel Cremers. 2017. Image-based localization using LSTMs for structured feature correlation. In *Proc. IEEE ICCV*. IEEE, 627–637.
[44] Chenshu Wu, Zheng Yang, and Chaowei Xiao. 2018. Automatic radio map adaptation for indoor localization using smartphones. *IEEE Transactions on Mobile Computing* 17, 3 (March 2018), 517–528.
[45] Hongwei Xie, Tao Gu, Xianping Tao, Haibo Ye, and Jian Lu. 2016. A reliability-augmented particle filter for magnetic fingerprinting based indoor localization on smartphone. *IEEE Trans. on Mobile Computing* 15, 8 (Aug 2016), 1877–1892.
[46] Han Xu, Zheng Yang, Zimu Zhou, Longfei Shangguan, Ke Yi, and Yunhao Liu. 2015. Enhancing wifi-based localization with visual clues. In *Proc. ACM UbiComp*. ACM, 963–974.
[47] Tien-Ju Yang, Andrew Howard, Bo Chen, Xiao Zhang, Alec Go, Mark Sandler, Vivienne Sze, and Hartwig Adam. 2018. NetAdapt: Platform-aware neural network adaptation for mobile applications. In *Proc. Springer ECCV*. Springer International Publishing, 289–304.
[48] Shuochao Yao, Yiran Zhao, Huajie Shao, ShengZhong Liu, Dongxin Liu, Lu Su, and Tarek Abdelzaher. 2018. FastDeepIoT: Towards understanding and optimizing neural network execution time on mobile and embedded devices. In *Proc. ACM SenSys*. ACM, 278–291.
[49] Zuwei Yin, Chenshu Wu, Zheng Yang, and Yunhao Liu. 2017. Peer-to-peer indoor navigation using smartphones. *IEEE Journal on Selected Areas in Communications* 35, 5 (May 2017), 1141–1153.
[50] Chaoyun Zhang, Paul Patras, and Hamed Haddadi. 2019. Deep learning in mobile and wireless networking: A survey. *IEEE Communications Surveys & Tutorials* (Online 2019).
[51] Chi Zhang and Xinyu Zhang. 2017. Pulsar: Towards ubiquitous visible light localization. In *Proc. ACM MobiCom*. ACM, 208–221.
[52] Yuanqing Zheng, Guobin Shen, Liqun Li, Chunshui Zhao, Mo Li, and Feng Zhao. 2017. Travi-Navi: Self-deployable indoor navigation system. *IEEE/ACM Trans. on Networking* 25, 5 (Oct 2017), 2655–2669.
[53] Jincao Zhu, Youngbin Im, Shivakant Mishra, and Sangtae Ha. 2017. Calibrating time-variant, device-specific phase noise for COTS WiFi devices. In *Proc. ACM SenSys*. ACM, 15:1–15:12.

RIGHTSLINK