

# Smartphone-based Indoor Localization

Zhu Zhicheng

Nanyang Technological University  
G1903395J  
Singapore  
zh0098ng@e.ntu.edu.sg

Tan Mengxuan

Nanyang Technological University  
G1903482L  
Singapore  
mtan099@e.ntu.edu.sg

Ong Jia Hui

Nanyang Technological University  
G1903467L  
Singapore  
jong119@e.ntu.edu.sg

**Abstract**—Smartphones are sensor-rich devices. Smartphone-based indoor localization is a device-based localization technique that uses the readings collected by the various sensors at various time intervals. In this report, the team first analyzed and pre-processed the sample spatio-temporal dataset provided by Microsoft Indoor Competition 2.0 [1]. Subsequently, the waypoint, Geomagnetic and WiFi RSSI data were visualized using Python’s matplotlib library. Lastly, a deep learning-based location fingerprinting model was developed to predict the indoor positions based on different feature inputs.

**Index Terms**—urban computing, indoor localization

## I. INTRODUCTION

### A. Background

There are two main schemes of wireless indoor localization namely; device-based and device-free. In device-based localization, a wireless device like a smartphone is attached to the target and its location is computed based on its interaction with other deployed wireless devices in the building. On the other hand, in device-free localization, the wireless infrastructure of the environment determines the person’s location by analyzing their impact on wireless signals. Regardless of the schemes used, the aim of indoor localization is to accurately pinpoint the location of the target through wireless signals. In this project, the team was tasked to process the smartphone-based indoor spatio-temporal data provided into appropriate visualizations.

### B. Objectives

The main objective of this project is to work with the data collected by various smartphone sensors and understand the challenges of using such spatio-temporal data. The sample dataset from Microsoft Indoor Location Competition 2.0 [1] will be used to study the device-based indoor sensor readings. After the team’s preliminary data analysis, the dataset is found to contain dense magnetic and WiFi data, but has comparably much sparser waypoints as they are manually recorded by the surveyors. Hence, data augmentation was performed during data preprocessing to alleviate this issue to allow more precise visualizations of magnetic and RSSI heatmaps along the surveyors’ footpaths. The deep learning location fingerprint model built by the team will be elaborated and discussed in latter sections.

## II. DATASET

### A. Data Information

The full Microsoft Indoor Location Competition 2.0 sample dataset consists of dense indoor signatures of WiFi, geomagnetic field, iBeacons etc., as well as ground truth collected from 2,718 floors of 300 buildings in 3 Chinese cities [1]. The dataset used in this project is sample set containing indoor traces from two buildings, denoted as site1 and site2 respectively. These trace data are collected by a site-surveyor using an Android smartphone attached to his body while walking from point to point within the site building.

The given dataset contains raw trace files saved and grouped within each floor of site directories. Floor plan metadata such as raster floor image, size information (height, width) and geojson map json files can also be found in the floor directory. The collected scanning results in trace text files are formatted as  $\{Time\}\{Data\ Type\}\{Value\}$ , where  $\{Time\}$  values are in Unix Time (milliseconds) and the  $\{Value\}$  of each row varies depending on the  $\{Data\ Type\}$ .

TABLE I: Values for each Data Type

S/N	Data Type	Values (Space delimited)
1	TYPE_WAYPOINT	Px, Py
2	TYPE_ACCELEROMETER	X, Y, Z, accuracy
3	TYPE_GYROSCOPE	X, Y, Z, accuracy
4	TYPE_MAGNETIC_FIELD	X, Y, Z, accuracy
5	TYPE_ROTATION_VECTOR	X, Y, Z, accuracy
6	TYPE_ACCELEROMETER_UNCALIBRATED	X_b, Y_b, Z_b, X_a, Y_a, Z_a, accuracy
7	TYPE_GYROSCOPE_UNCALIBRATED	X_b, Y_b, Z_b, X_a, Y_a, Z_a, accuracy
8	TYPE_MAGNETIC_FIELD_UNCALIBRATED	X_b, Y_b, Z_b, X_a, Y_a, Z_a, accuracy
9	TYPE_WIFI	ssid, bbsid, RSSI, frequency, lastseen
10	TYPE_BEACON	UUID, MajorID, MinorID, Tx Power, RSSI, Distance, MAC, Unix Time

Table I enumerates the space delimited values for each data types. There are a total of ten  $\{Data\ Type\}$  ranging from IMU (accelerometer, gyroscope), geomagnetic field (magnetometer) readings to WiFi and Bluetooth iBeacon scanning results. The ground truth data, which is the indoor position of surveyor,  $TYPE\_WAYPOINT$  can be found within

the same text file along the sensor readings.

### B. Dataset Analysis

TABLE II: Record Statistics

Site	Floor	Height	Width	Waypoints	A/M/R	WiFi
1	B1	232	320	1034	265122	333724
1	F1	176	240	975	290966	868647
1	F2	177	240	1049	382553	778674
1	F3	179	242	1012	475461	702449
1	F4	179	242	1042	356808	703488
2	B1	393	304	534	132112	201667
2	F1	220	237	1006	272351	795937
2	F2	220	237	362	85189	228229
2	F3	220	237	278	69756	177006
2	F4	220	237	215	54967	122219
2	F5	220	237	298	67935	163887
2	F6	220	237	565	130772	272565
2	F7	220	237	273	67491	178423
2	F8	220	237	265	66601	147718

An exploratory data analysis (EDA) is performed on the sample dataset to better understand the data distribution. Table II shows the computed statistics of each floor in the two sites. The floor plan information like height and width, number of waypoints, number of ACCELEROMETER & MAGNETIC\_FIELD & ROTATION\_VECTOR data (**A/M/R**) and number of WiFi data are tabulated in Table II.

Several observations are made during the tabulation. Firstly, the **A/M/R** sensor readings share the same timestamps, which means that the measurements of the three sensors are recorded at the same time. Another notable finding is that the WiFi readings do not share the same timestamp as the **A/M/R** sensor readings. At each time interval, multiple WiFi data with different RSSI are recorded as separate entries in the text file. It is important to note that the timestamps for each reading do not directly correspond to the timestamps in recorded waypoints.

Table II also revealed that the ratio of waypoint data points to sensor readings is very low. This sparsity of the raw ground truth data will be too coarse to be used as time positions for plotting of geomagnetic and WiFi RSSI heatmaps. Hence, some form of data augmentation is required to estimate the waypoints at finer timestamps to plot more accurate visualizations. Details on this implementation will be elaborated in Section II-C.

### C. Data Preprocessing

Due to the sparsity of the ground truth waypoint positions, there is a need for data augmentation to create more estimated indoor positions. These estimated positions are termed as step positions and are used to represent the predicted positional location of the surveyor when the sensor readings are recorded at the particular timestamp.

Using the *TYPE\_ACCELEROMETER* data, the changes in accelerometer magnitudes can be used to detect the surveyor's movement [2]. When the magnitude of the accelerometer data reached the maximum above certain threshold, it is considered as the peak of the step. On the other hand, when a decreasing acceleration reached the minimum below certain threshold, the value is treated as a valley. Every peak is counted as a step. After step detection, the length of stride (distance) for each step will be computed by taking the difference between two valley points of the accelerometer magnitudes. The *TYPE\_ROTATION\_VECTOR* data collected through the gyroscope will be used to compute the body orientation of the surveyor. Using the stride length, orientation and the actual timestamps, the relative directional positions of the surveyor's movement can be found. The final step positions at each **A/M/R** timestamps are estimated based on the absolute positions of the ground truth positions.

Although the WiFi RSSI data are collected at different time intervals from the **A/M/R** data, the augmented step positions of **A/M/R** timestamps can be used to estimate the WiFi readings timestep due to the high frequency of **A/M/R** data. Hence in the WiFi RSSI visualizations, the nearest timestep positions from augmented or actual step positions as waypoints.

## III. ESSENTIAL TASKS

In this section we present the visualizations of waypoints, geomagnetic heatmaps as well as the RSS heatmaps. For both the geomagnetic heatmaps and RSS heatmaps, we compare the visualizations between those generated solely from the ground truth waypoints and those generated from the augmented step positions.

### A. Visualization of Waypoints

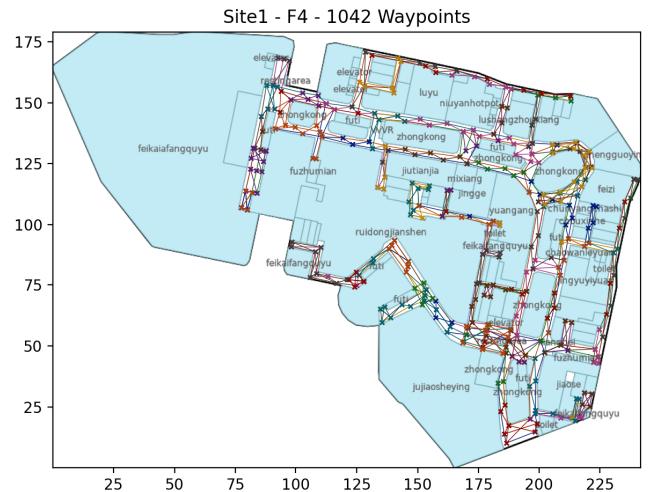


Fig. 1: Plot showing the most (1042) waypoints collected on the 4th Floor of Site 1

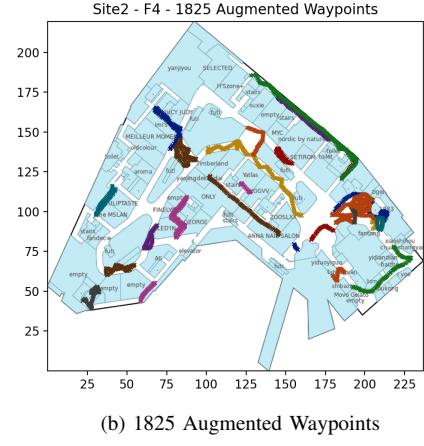
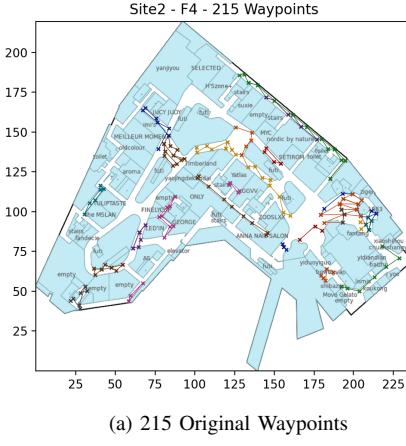


Fig. 2: Plots showing the original waypoints and augmented step positions on the 4th Floor of Site 2

We visualized the waypoints using Matplotlib plotting function and Seaborn for dynamic colouring. The original floor image was first loaded into the plotting function. Each floor of each site contains multiple text files. These text files are processed one by one, with datapoints of *TYPE\_WAYPOINT* retrieved and grouped together. Subsequently, each group of waypoints (x,y coordinates) are plotted on top of the floor plan with the same coloured marker, “x”. The lines drawn across each group also have the same colour as the group’s markers and they are representative of the estimated routes along the recorded points.

Figure 1 and 2a represent the plots for the most and least number of waypoints collected in the sample dataset respectively. All other original waypoint plots for the rest of the sites can be found in Appendix A-A. Based on the tabulated waypoint counts shown in Table II, it is evident that the waypoints recorded by the surveyors in Site 2 are much lesser than that of Site 1. The result of this sparsity can also be seen in the large gaps between the markers in Figure 2a. Moreover, the different coloured intersecting lines (estimated path) in Figure 1 also revealed that the surveyor for Site 1 Floor 4 either walked along the same pathway multiple times or there may be more than one surveyor collecting the sensor data. We hypothesize that the sparsity of the collected waypoints in Site 2 will adversely impact the outputs of the geomagnetic and RSSI heatmaps. Hence as discussed in Section II-C, there is a need to perform data augmentation using the accelerometer and rotation data to map with the denser geomagnetic and RSSI data.

Furthermore, from the original waypoint plots, it is difficult to gauge precise positioning along wider footpaths. The effects of augmenting only the recorded waypoints can be seen in Figure 2b. It shows the augmented step positions visualization in Site 2 Floor 4. The waypoint markers are much denser, filling in most positions along the surveyor’s trails as compared to estimated paths denoted by the lines in the original plot in

TABLE III: Original versus Augmented waypoint counts

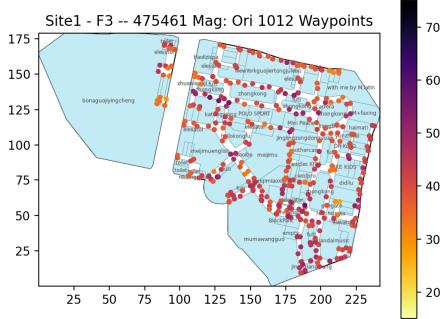
Site	Floor	Waypoints	
		Original	Augmented
1	B1	1034	8468
1	F1	975	8483
1	F2	1049	9569
1	F3	1012	9138
1	F4	1042	8982
2	B1	534	4363
2	F1	1006	8759
2	F2	362	2766
2	F3	278	2142
2	F4	215	1825
2	F5	298	2270
2	F6	565	4511
2	F7	273	2326
2	F8	265	2211

Figure 2a. Hence, geomagnetic and RSSI data can now be mapped more precisely along the pathway of the surveyor. This is also shown in Table III where the augmented waypoint counts were greatly increased over the ground truth positions. The rest of the waypoint plots can be found in Appendix A-A and A-B.

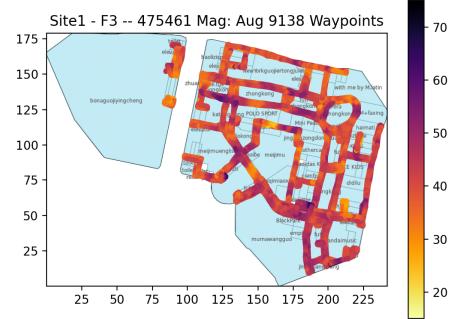
#### B. Visualization of Geomagnetic Heatmaps

As shown in Table II, we know that the data points of the geomagnetic sensor readings, denoted as **A/M/R**, are collected much more frequently than the manually recorded waypoints. This is likely because the readings are automatically collected by the smartphone while the surveyor(s) walked from one waypoint to another. If the original waypoints and their timestamps are used to plot the geomagnetic heatmaps, it will result in the sparse plot as shown in Figure 3b.

In order to plot more accurate geomagnetic heatmaps along the footpaths taken, there is a need to perform data augmentation as described in Section II-C. The resulting step positions allow us to map the geomagnetic sensor readings to the closest timestamps shown in Figure 3a. The steps of aligning each geomagnetic reading ( $MG_i = (x_i, y_i, z_i, t_i)$ )



(a) 1012 Original Waypoints



(b) 9138 Augmented Waypoints

Fig. 3: Geomagnetic heatmaps for 3rd Floor of Site 1 with the most (475461) readings mapped to different waypoint counts

to the step positions starts with attaching each geomagnetic data to the closest augmented waypoints data (Equation 1). After this, since there are much more geomagnetic readings (Table II) than augmented waypoints (Table III), there will be multiple  $MG_i$  readings linked with the same time step  $t_k^{WP}$ . Hence, these magnetic values are aggregated using Equation 2, whereby the mean of the sum squared of  $x, y, z$  coordinates of the  $N$  number of  $MG$  readings at the same time step  $k$  is computed.

$$MGlinks_k = \{(x_i, y_i, z_i, t_i) | k = \arg \min_j |t_j^{WP} - t_i|\} \quad (1)$$

$$WPmg_k = \frac{1}{N} \left( \sum_{MGlinks_k} \sqrt{x_i^2 + y_i^2 + z_i^2} \right) \quad (2)$$

where  $t_j^{WP}$  is the timestamp of  $j$ th waypoints data

$N$  is the length of  $MGlinks_k$

These aggregated values can be viewed as the magnetic intensity strength, with higher values representing a stronger intensity (darker hue) over the lower values. Figure 3a shows the magnetic intensity plot after mapping the data points to the augmented step positions. Note that the “cmap” of the colorbar is intentionally inverse to allow stronger intensity to have a dark hue and vice versa. All heatmaps are plotted on the colour range between intensity value 15 to 75, with outliers being coloured as either the upper or lower bound. The rest of the magnetic heatmaps can be found in Appendix A-C.

### C. Visualization of RSS Heatmaps

We visualize the RSS of 3 Wi-Fi APs at 3 different locations according to both the ground truth waypoints and the augmented waypoints, shown in Figure 4. Each  $TYPE\_WIFI$  instance is denoted as  $WIFI_i = (r_i, t_i, bssid_i)$  where  $r_i$  represents the Wifi signal strength of this instance and  $bssid_i$

is the BSSID of the AP of this instance. To localize a  $WIFI_i$  instance, we compare the time  $t_i$  this instance is generated with all timestamps  $t_j^{WP}$  in the ground truth waypoints or the augmented waypoints, depending on which method we are using. We then assign the waypoint position ( $P_x, P_y$ ) whose timestamp is closest to the time the  $TYPE\_WIFI$  instance is generated, this is same as the procedure of aligning geomagnetic readings to step positions in Section III-B. Similar to Equation 1, we can use Equation 3 to find all the Wifi instances with the same BSSID  $b$  for every waypoint.

$$WIFI_k^b = \{(r_i, t_i, bssid_i) | k = \arg \min_j |t_j^{WP} - t_i|\} \quad (3)$$

After we have determined the positions of all  $TYPE\_WIFI$  instances, for an unique AP with BSSID  $b$  at one particular position, there may be multiple  $TYPE\_WIFI$  RSSI  $r_i$  records. Hence, we simply compute the average RSSI to represent the Wifi strength of an AP at a particular waypoint. Similar to Equation 2, we can use the following formula to compute the RSSI of an AP with an unique BSSID  $b$ :

$$WP_{rss_k}^b = \frac{1}{N} \sum_{WIFI_k^b} r_i \quad (4)$$

where  $WP_{rss_k}^b$  represents the RSSI strength of an AP with unique BSSID  $b$  at a particular waypoint position.  $k$  is the time at which this waypoint is generated. Similar to Equation 2,  $N$  here is the total number of  $WIFI_i$  instances records at timestep  $k$  for BSSID  $b$ .

From what we have expected, using the augmented waypoints, we are able to obtain a more fine-grained visualization of an AP RSSI (Figure 4), allowing us to better understand the change in signal strength at different positions of a floor. Here, a lighter colour represents a stronger RSSI strength and from the colour of the heatmap, we may infer the exact location of an AP. For example, for the chosen AP (06:74:9c:2b:28:af) at F1 of Site 1, we can see a cluster of light green dots

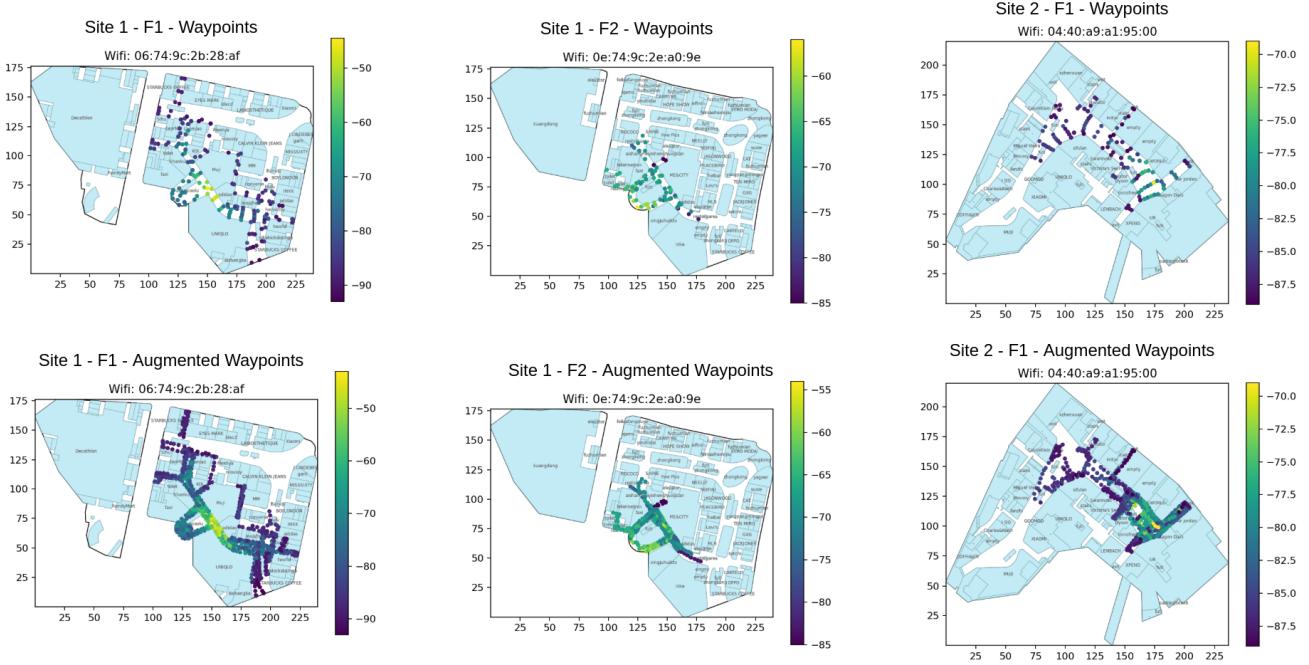


Fig. 4: RSS heat maps of 3 Wi-Fi APs at 3 different locations. Comparing those generated from waypoints (top row) with those generated from augmented waypoints (bottom row).

surrounding the *Huangxiaolu* store, suggesting that the AP may be in that store. In Appendix A-D, we show the RSS heatmaps of a random AP for all floors in Site 1 and Site 2.

#### IV. BONUS TASKS

##### A. Deep learning Location Fingerprinting Model

The goal of the deep learning fingerprinting model is to carry out accurate indoor localization given certain features of an user's smartphone data. In this section, we describe how such a model can be developed. First we show how we preprocess and feature engineer the given data so that they are suitable as input to the deep learning model. Next, we illustrate the design of the main architecture of the model and lastly show some localization results of the deep learning model. In the results section, we also show how the performance of our model change given different combinations of sensors' features.

*1) Preprocessing and feature engineering:* The groundtruth waypoints ( $P_x, P_y$ ) recorded by the volunteers will be used as data labels for our deep learning model. The raw data from the Geomagnetic, Wifi and iBeacon sensors are also aligned to the waypoints according to procedure described in Section III-B and Section III-C. We have attempted to use the augmented time step positions as the ground truth labels, but based on our experiments, we found that the prediction accuracy of using the recorded waypoint values is much higher. Hence, we will only document the results of using actual timestamp in the following section.

We introduced two additional features  $\text{wifi}_{\text{det}}$  and  $\text{ibeacon}_{\text{det}}$  to indicate whether or not WiFi detection or Beacon detection have been carried out when the data is being recorded. The reason for adding these features is that at many instances, the WiFi sensors are unable to pick up WiFi signals and hence their RSS are almost negligible. However, the model may be mistaken that there is totally no WiFi at this location. Hence we want to differentiate between the case where there is actually no WiFi at a location and the case where there is WiFi at a location but the sensors are unable to pick up the signal. This is also the reason for introducing the  $\text{ibeacon}_{\text{det}}$  feature.

We split our dataset such that 80% will be used for training our model and 20% for testing. The final inputs to the deep learning model consists of features from all 3 types of sensors: Geomagnetic, WiFi and iBeacon. The labels used to train the deep learning model in a supervised manner are the groundtruth waypoints. Hence:

$$\text{Input}_{\text{Geomagnetic}} = [M_x, M_y, M_z, MI]^T$$

$$\text{Input}_{\text{Wifi}} = [\text{wifis}, \text{wifi}_{\text{det}}]^T$$

$$\text{Input}_{\text{iBeacon}} = [\text{ibeacons}, \text{ibeacon}_{\text{det}}]^T$$

$$\text{Output Label} = (P_x, P_y)$$

where  $M_x, M_y, M_z$  and  $MI$  are the geomagnetic data. **wifis** and **ibeacons** indicate the Wifi strengths and Beacon strengths of different Wifi APs and Beacon points respectively. The dimensions of **wifis** and **ibeacons** depend on how many BSSIDs and UUIDs on a floor's train data respectively. For example, if Site2 F4 has 2000 unique BSSIDs in the train data and 500 BSSIDs in test data, then the dimension of **wifis** will be 2000. For unique BSSIDs feature that are present in the test data but not in train data, they will be ignored during evaluation.

**2) Model Architecture:** We design a model to take in features from the 3 types of sensors to predict the groundtruth waypoints for localization. Figure 5 illustrates the architecture of our deep learning model. The model consists of 3 encoders networks that are tasked to learn meaning representations from the raw data of the 3 sensors separately. The model then merges the learned or encoded representations from the 3 encoders and passes these information to a final set of hidden layers to predict the location. It is also worth noting that we have also added several ‘skip’ connections from the raw input data to the merging layer. The reason for doing so is to allow the network to preserve the importance of the raw sensors’ information while learning meaningful representation from the raw data at the same time. For instance, the geomagnetic sensor’s data  $[M_x, M_y, M_z, MI]^T$  is used as inputs to both the Geomagnetic Encoder as well as the merging layer. As we have observed that the geomagnetic sensors’ data is more commonly found in our samples as compared to the data from the other sensors, we want to avoid the loss of important information when the data goes through the encoder network. This type of ‘skip’ connections are typically found in architectures like the ResNet [3] and U-Net [4] where the networks aim to preserve important information when training very deep networks.

**3) Loss Function:** Since we are training a supervised learning model to predict the locations given sensors data, we define a mean-squared error (MSE) function to train the deep learning model as follows:

$$MSE = \frac{1}{M} \sum_{i=1}^M (P_{xi} - \hat{P}_{xi})^2 + (P_{yi} - \hat{P}_{yi})^2 \quad (5)$$

where  $M$  represents the total number of samples in our training set.

**4) Evaluation Metric:** The performance of the model is being evaluated on the test set where we measure the Euclidean distance between the predicted location and the groundtruth location:

$$Metric = \sqrt{(P_x - \hat{P}_x)^2 + (P_y - \hat{P}_y)^2} \quad (6)$$

A smaller Euclidean distance would indicate a better performance by the model.

**5) Training:** Figure 6 shows our loss curve and error plots of our model training over 120 epochs. The model was initially set to train for 300 epochs, but early stopping was triggered at around 120 epochs when there is no improvement in the training and validation errors.

**6) Results:** In this section, we show the results of our model for every floor in the 2 sites. We also analyse how combinations of features from the different sensors may affect the performance model. For every combination of sensors features, we train a model for every floor in the 2 sites. The results are displayed in Table IV.

TABLE IV: Performances of our deep learning models on all floors of the 2 sites. The average Euclidean distance (in meters) between the groundtruth data in the test sets and predictions are shown.

Site/Floor	Geomagnetic	Geomagnetic + Wifi	Geomagnetic + iBeacon	All Sensors
1/B1	52.88	9.76	30.06	9.64
1/F1	46.78	7.66	48.75	7.71
1/F2	45.88	10.00	43.15	11.51
1/F3	47.19	10.01	38.52	11.44
1/F4	47.51	9.65	42.13	11.44
2/B1	42.79	12.30	27.13	14.89
2/F1	44.17	12.12	32.70	10.83
2/F2	39.56	11.48	37.82	15.76
2/F3	37.42	10.77	41.36	10.54
2/F4	49.42	14.02	47.91	12.27
2/F5	41.87	6.35	39.14	14.25
2/F6	57.19	12.02	44.23	14.05
2/F7	50.49	10.99	41.43	11.63
2/F8	36.92	13.49	37.34	12.79
<b>Average</b>	<b>45.72</b>	<b>10.76</b>	<b>39.41</b>	<b>12.05</b>

As shown in Table IV, all feature combinations contain geomagnetic data. This is to ensure that every sample contains some feature values for our network to learn and evaluate. As Wifi and iBeacon data are very sparse, using them without the geomagnetic data may result in unsatisfactory performances. Do also note that the model architecture shown in Fig. 5 will also change according to the sensors combinations. For instance, the model will only have the Geomagnetic Encoder in the ‘**Geomagnetic**’ setting and only the Geomagnetic Encoder and Wifi Encoder will be present in the ‘**Geomagnetic + WiFi**’ setting.

From the final Euclidean distances comparison between the different feature combinations shown in Table IV, we can conclude that the fingerprinting model will fare poorly if only “Geomagnetic” or “Geomagnetic + iBeacon” features are used. The best average Euclidean distance error of 10.76m is found to only use ‘**Geomagnetic + WiFi**’ data. Our team further explored the reason for the predictions using “All Sensors”, that is with inclusion of “iBeacon” data, fared worse than the predictions of “Geomagnetic + iBeacon”. Based on our analysis, we hypothesize it could be due to the missing iBeacon data at some of the time steps at some sites that lead to the poor performance.

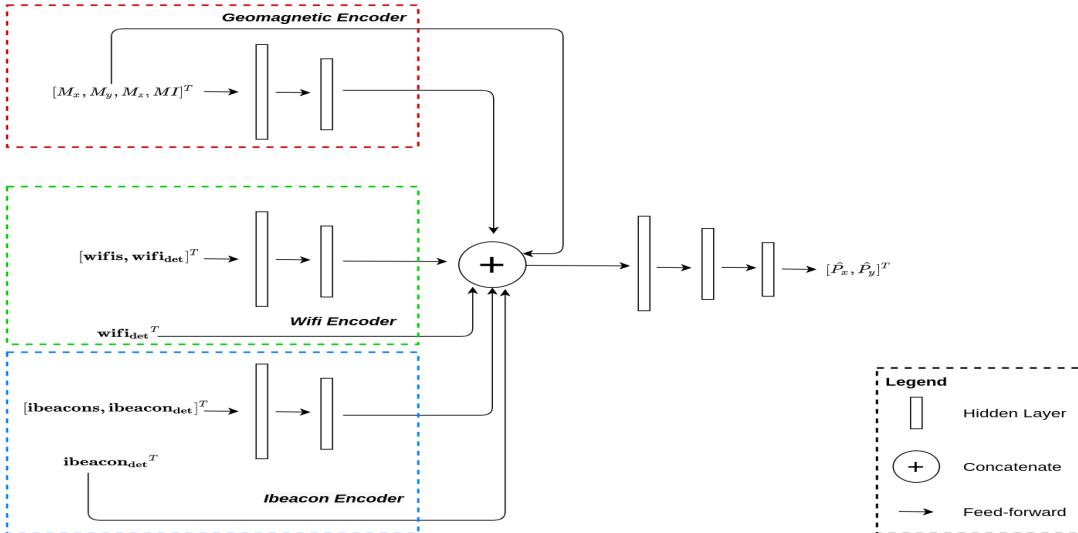


Fig. 5: Model architecture of our deep learning-based location fingerprinting model.

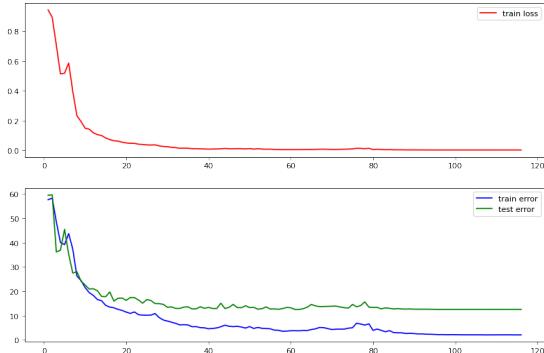


Fig. 6: Plots showing training loss (top) and training + validation error (bottom) for Site 2 F2.

## V. CONCLUSION

After working with the Microsoft Indoor Localization sample dataset, the team now collectively has a better understanding on the various sensors available in the Android smartphone and the spatio-temporal data they can collect. These spatio-temporal data include accelerometer, gyroscope, magnetic field, rotation vector, RSSI and iBeacon data. However, one of the major challenges faced by the team was the sparsity of the manually recorded waypoints. As a result, we had to preprocess the data to generate more estimated time step positions of the surveyors in order to produce better magnetic and RSSI heatmap visualizations.

For the first essential task, we plotted and compared the differences between the original ground truth waypoints and the augmented time step positions. From the plots, we can infer that the waypoints recorded in site 1 are much denser than those of site 2 in the given sample dataset. As the time step positions of collected magnetic and WiFi data are still much greater than the augmented time steps, some form

of mean aggregation needs to be performed when multiple sensor readings are mapped to the same time step. Using this aggregation technique, we are able to plot both the geomagnetic and WiFi RSSI heatmaps for the second and third essential tasks.

Next, our team moved on to develop a deep learning location fingerprinting model. Three sensor data types, namely Geomagnetic, WiFi and iBeacon data are experimented as inputs for the model. The model is designed to be modular by first encoding the meaning representation of each of sensor data using separate layers. Taking reference of the ‘skip’ connections popularized by modern deep learning approaches, we concatenated the feature representations together with their original sensor readings into one merged feature representation. These merged features will then be used as the inputs of another Decoder network to generate predictions. The loss function for the model is MSE and the evaluation metric is the Euclidean distance between the groundtruth location and the predicted positions. Based on our experiments of the different sensor readings from the given sample dataset, we found that using both Geomagnetic and WiFi data performed the best with the least average error.

Finally, the team can draw conclusions to the difficulties of working with spatio-temporal data after performing the various tasks. Firstly, as the sensor readings are recorded irregularly, the sensor readings aggregation to the estimated time step positions will inevitably lead to some form of inaccuracies, especially in cases with extreme values. Furthermore, as previously discussed, the sparsity of the waypoint data makes it difficult to predict for accurate time step positions even after augmentation.

## VI. FUTURE WORK

The team is keen in exploring more ways to improve on the location fingerprinting model in the future. Some of the ideas include training a Convolutional Neural Network using waypoint plots on the floor plan image to improve the prediction accuracy of the indoor positioning and also make use of an ensemble of various machine learning methods.

## GROUP MEMBER CONTRIBUTIONS

The data analysis and pre-processing steps are done collaboratively as a team. Each of the essential tasks are distributed equally among the group members. Ong Jia Hui completed the first essential task, Zhu Zhi Cheng performed the second essential task and Tan Mengxuan did the third essential task. All three members worked on the Deep-learning Location Fingerprinting Model together.

## REFERENCES

- [1] M. Research, “Indoor location competition 2.0 (sample data and code),” 2020. [Online]. Available: <https://github.com/location-competition/indoor-location-competition-20>
- [2] A. Abadleh, E. Al-Hawari, E. Alkafaween, and H. Al-Sawalqah, “Step detection algorithm for accurate distance estimation using dynamic step length,” in *2017 18th IEEE International Conference on Mobile Data Management (MDM)*. IEEE, 2017, pp. 324–327.
- [3] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [4] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

## APPENDIX

### A. All Waypoint Visualizations

Figures 7 and 8 show the original waypoint plots of all floors in Site 1 and Site 2. Note that the plots are best viewed after digital zoom.

### B. All Augmented Timesteps Visualizations

Figures 9 and 10 show the plots for the augmented time step positions of all floors in Site 1 and Site 2. Note that the plots are best viewed after digital zoom.

### C. All Magnetic Heatmaps

Figures 11 and 12 show the magnetic intensity plots of all floors in Site 1 and Site 2 after time step alignment. Note that the plots are best viewed after digital zoom.

### D. Wifi Heatmaps from all floors in both Sites

Figures 13 and 14 show the Wifi RSSI heatmap of a random AP of all floors in Site 1 and Site 2 using the augmented waypoints. Note that the plots are best viewed after digital zoom.

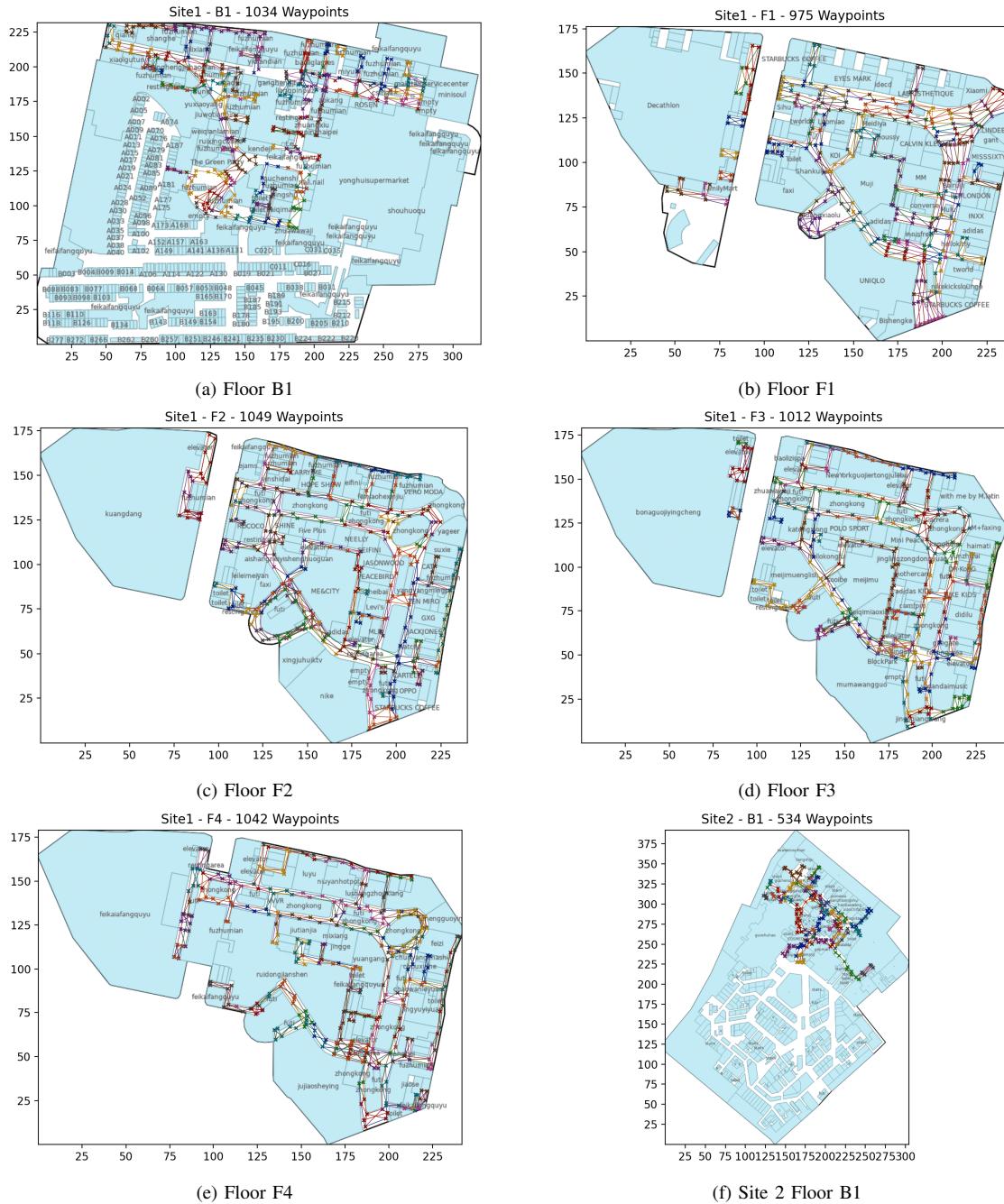
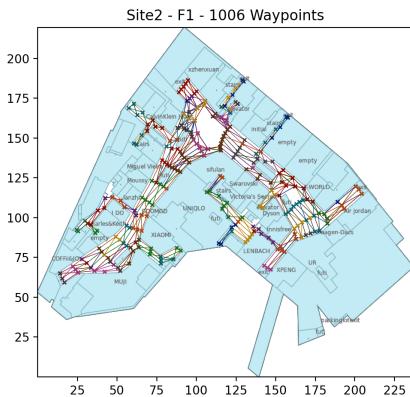
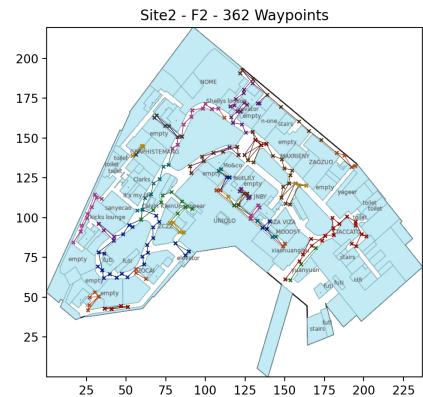


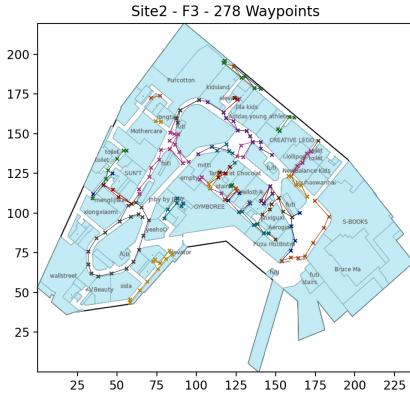
Fig. 7: Waypoint Visualizations of all floors in Site 1 and B1 of Site 2



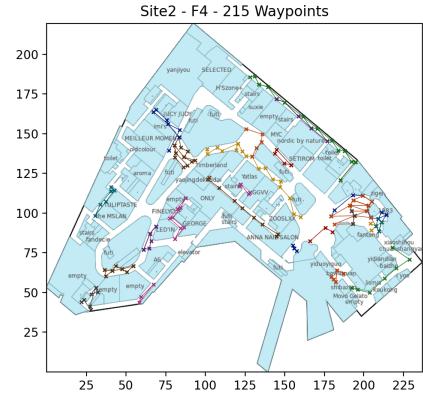
(a) Floor F1



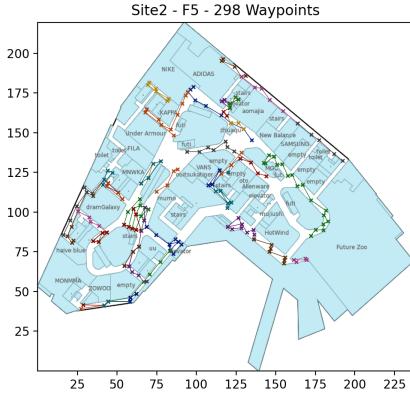
(b) Floor F2



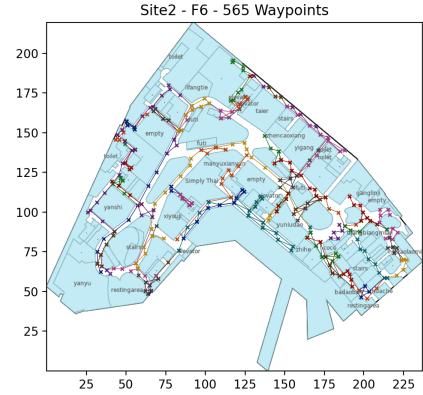
(c) Floor F3



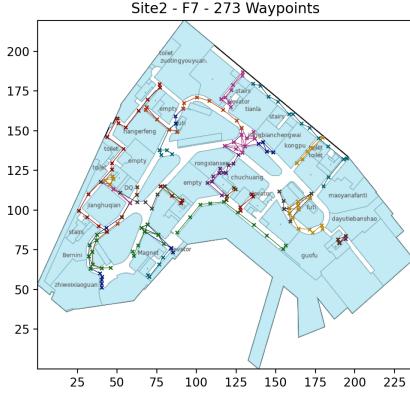
(d) Floor F4



(e) Floor F5



(f) Floor F6



(g) Floor F7

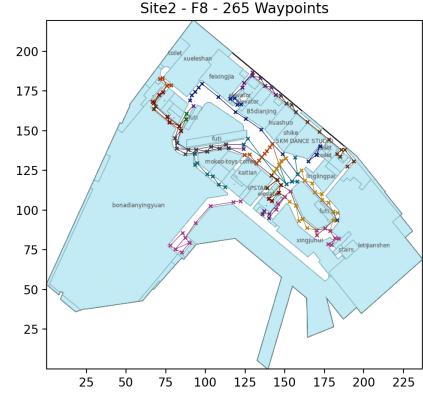


Fig. 8: Waypoint Visualizations of all floors in Site 2 except B1

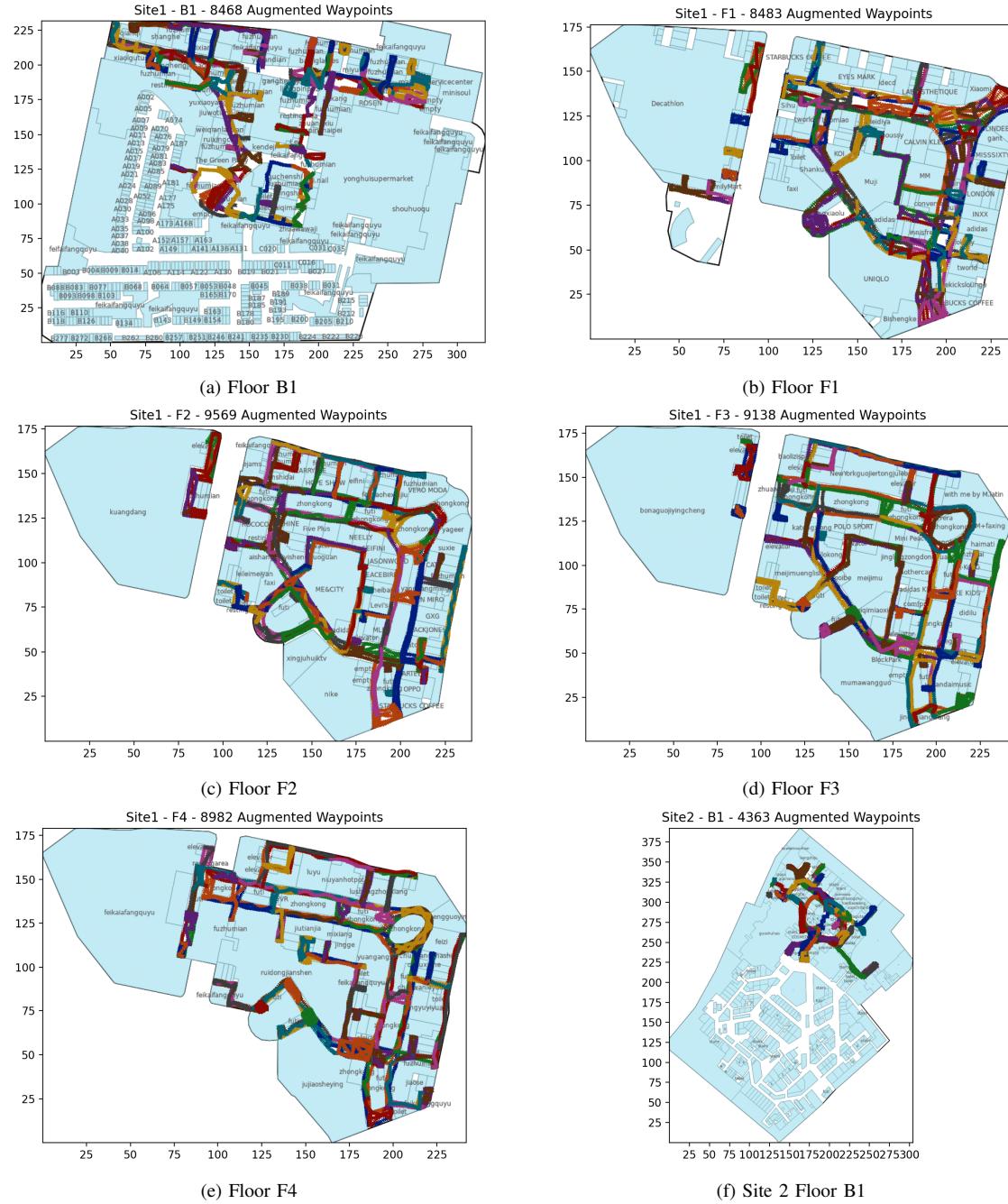
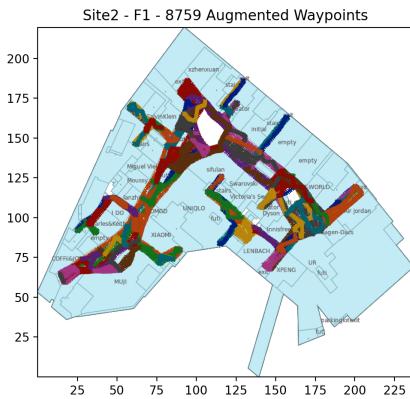
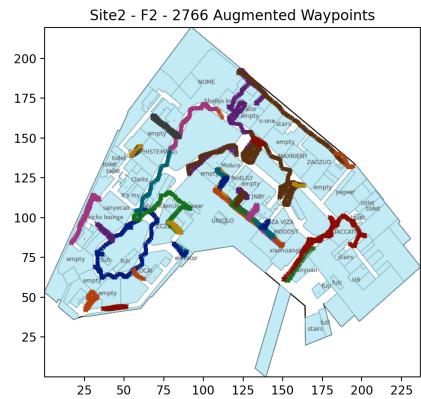


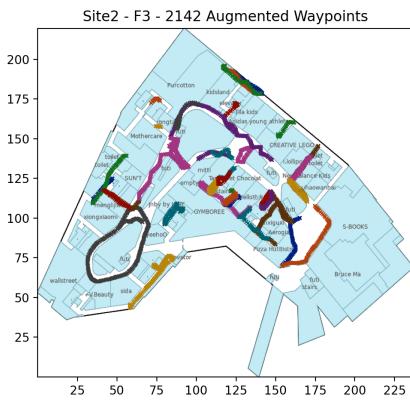
Fig. 9: Augmented Timesteps Visualizations of all floors in Site 1 and B1 of Site 2



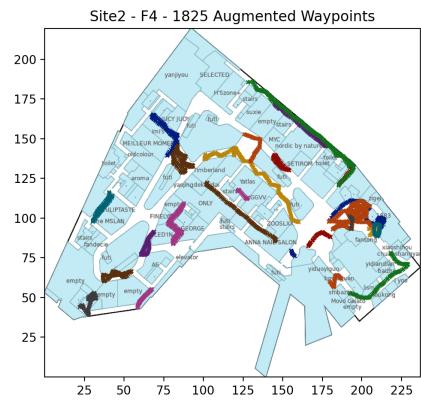
(a) Floor F1



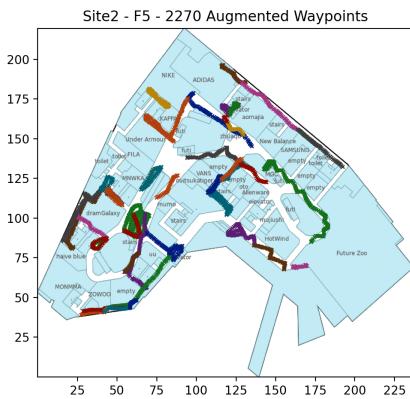
(b) Floor F2



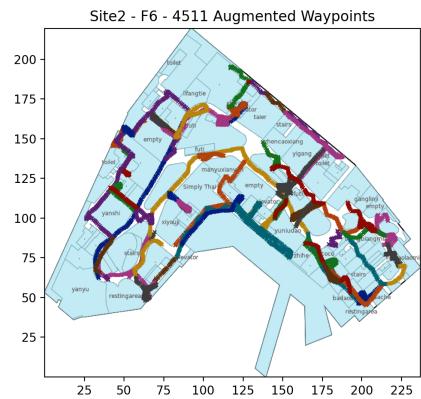
(c) Floor F3



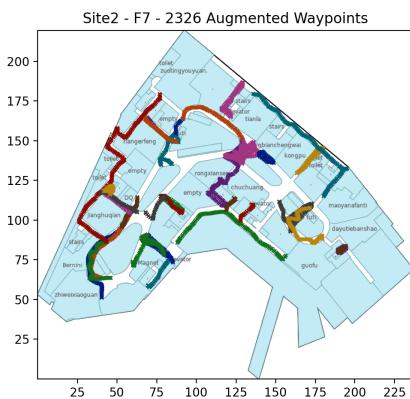
(d) Floor F4



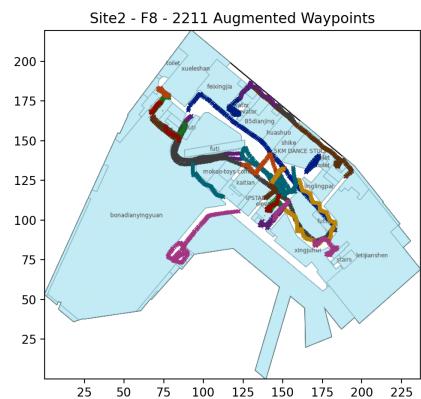
(e) Floor F5



(f) Floor F6

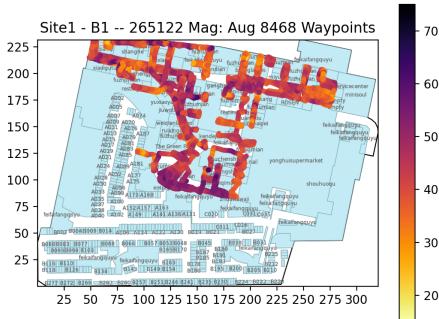


(g) Floor F7

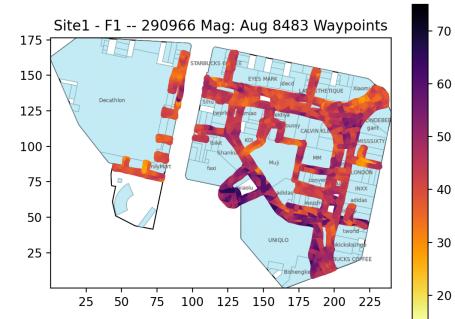


(h) Floor F7

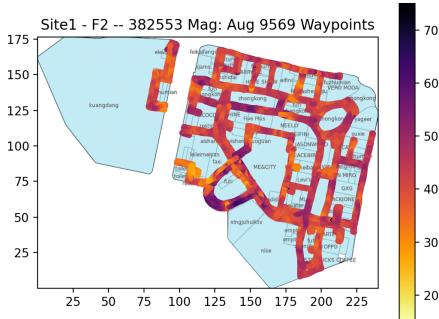
Fig. 10: Augmented Timesteps Visualizations of all floors in Site 2 except B1



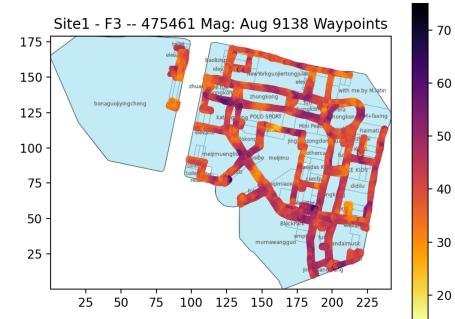
(a) Floor B1



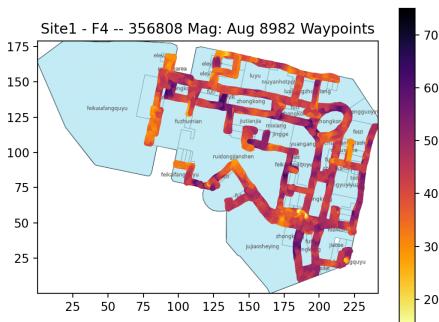
(b) Floor F1



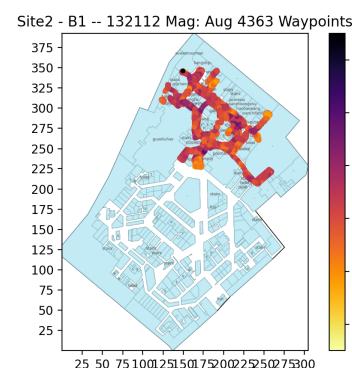
(c) Floor F2



(d) Floor F3

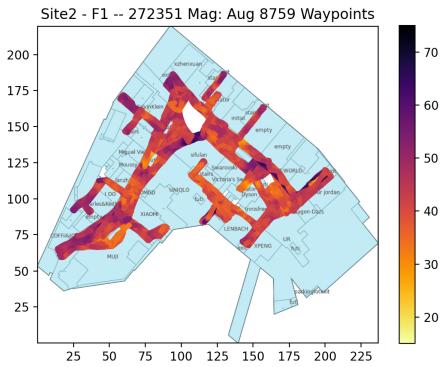


(e) Floor F4

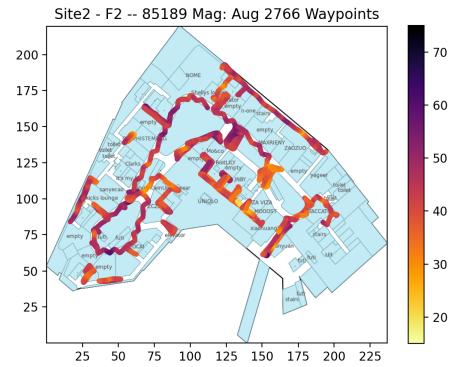


(f) Site 2 Floor B1

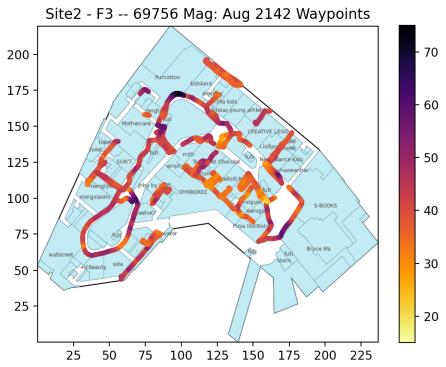
Fig. 11: Magnetic Heatmaps of all floors in Site 1 and B1 of Site 2



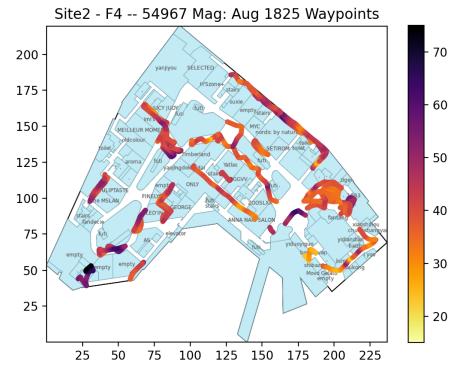
(a) Floor F1



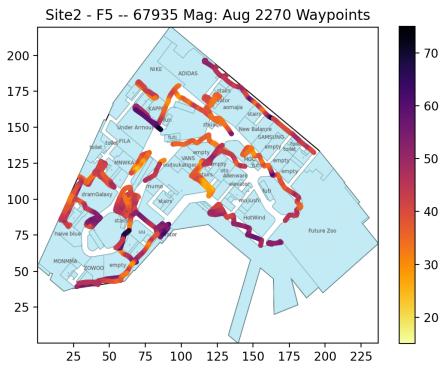
(b) Floor F2



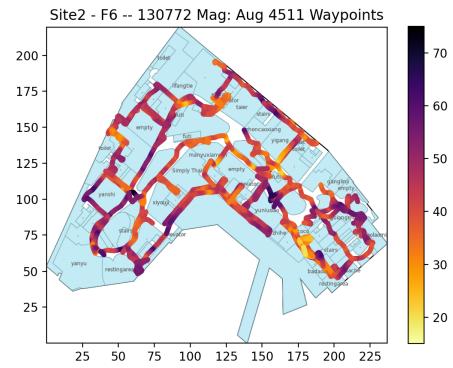
(c) Floor F3



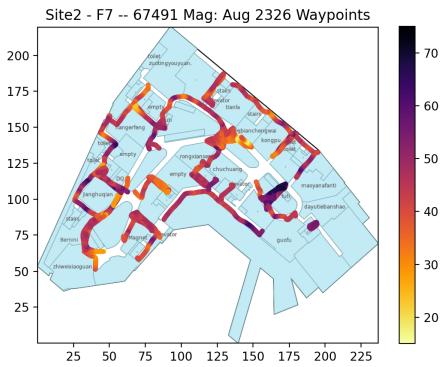
(d) Floor F4



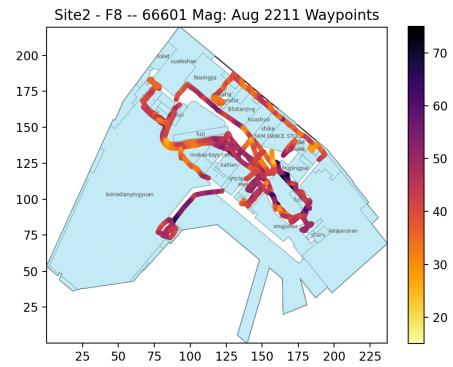
(e) Floor F5



(f) Floor F6

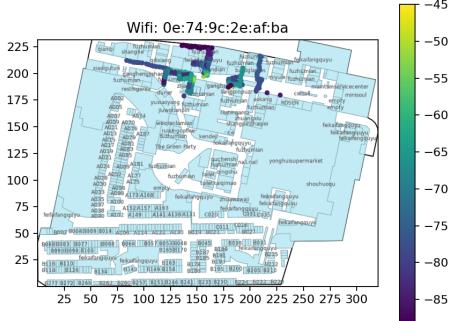


(g) Floor F7

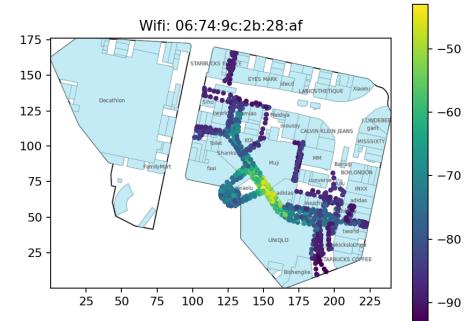


(h) Floor F7

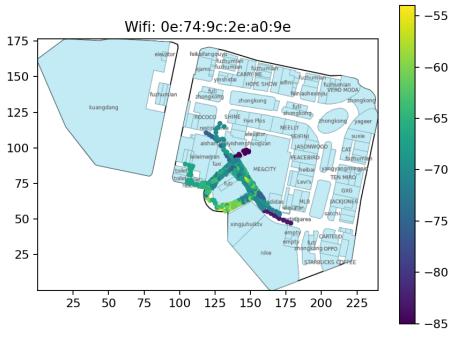
Fig. 12: Magnetic Heatmaps of all floors in Site 2 except B1



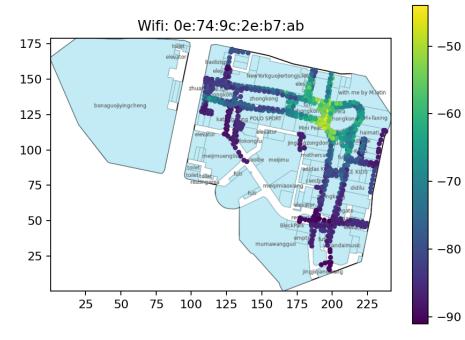
(a) Site 1 - B1



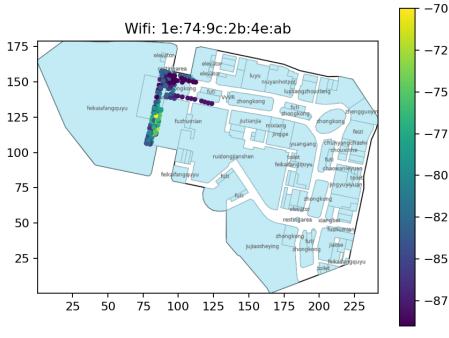
(b) Site 1 - F1



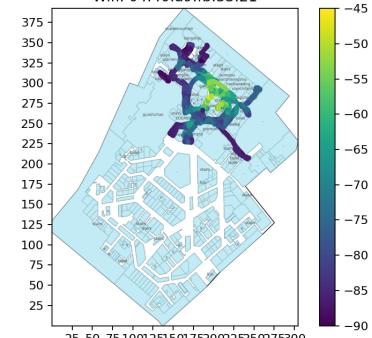
(c) Site 1 - F2



(d) Site 1 - F3

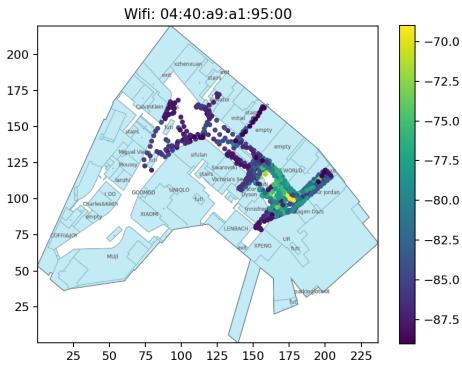


(e) Site 1 - F4

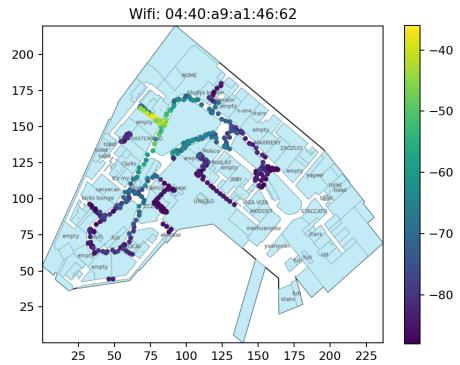


(f) Site 2 - B1

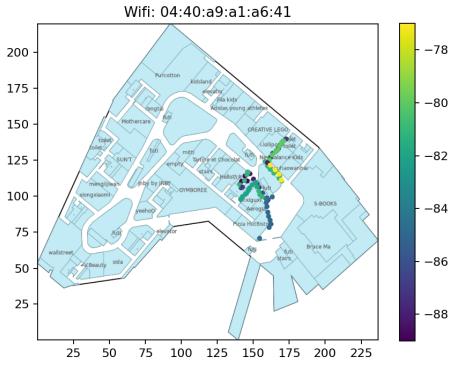
Fig. 13: Wifi RSSI Heatmaps of a random AP from augmented waypoints on all floors in Site 1 and B1 of Site 2



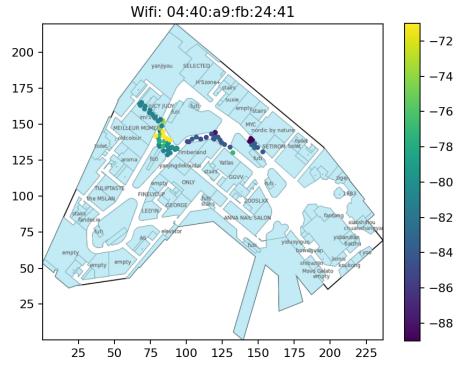
(a) Site 2 - F1



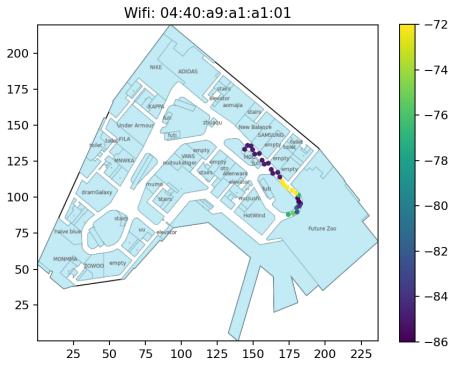
(b) Site 2 - F2



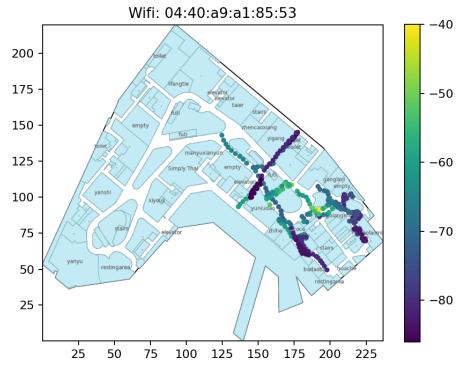
(c) Site 2 - F3



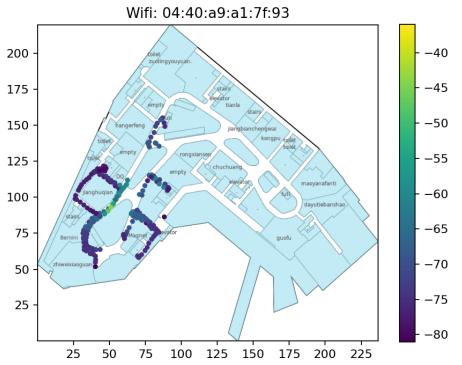
(d) Site 2 - F4



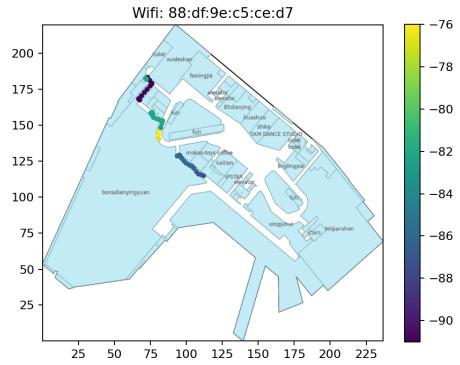
(e) Site 2 - F5



(f) Site 2 - F6



(g) Site 2 - F7



(h) Site 2 - F8

Fig. 14: Wifi RSSI Heatmaps of a random AP from augmented waypoints on all floors in of Site 2 except B1