Abstract—Indoor pathway models have recently emerged as promising indoor positioning techniques for indoor location-based Internet of Things (IoT) applications. However, the indoor pathway models reported in the literature are commonly extracted from indoor maps that are both coverage-limited and expensive. Furthermore, the extraction process is also time-consuming and error-prone. In this work, a novel method is developed to automatically construct semantic-rich indoor pathway models from the crowdsourced trajectory data collected by smartphone sensors without requiring maps and additional devices. More specifically, the pedestrian trajectories are first obtained using the built-in inertial sensors of the smartphone. After that, indoor pathway models are constructed by exploiting the pedestrian activity information derived from human activity recognition (HAR) and structural nodes (e.g. doors and elevators) extracted from the trajectory. Furthermore, a short-range trajectory clustering method is proposed to improve the accuracy of the indoor pathway model. In addition to indoor positioning, the resulting model can provide structural information of an indoor environment as well as semantic information about pedestrians and the environment, which is particularly useful for advanced IoT applications and services. Extensive field measurements demonstrate that the resulting indoor pathway models are of about one-meter accuracy in most experiments.

I. INTRODUCTION

As a new generation of information technology, the Internet of Things (IoT) is expected to transform every corner of our lives. In particular, indoor location-based services (LBSs) play a key role in expanding and enriching IoT applications in indoor environments. The pathway model has a significant advantage in supporting LBSs because it can accurately detect the indoor location and network relationships. Conventionally, the availability of indoor maps is required for developing any pathway models, which presents a few challenges in practice. First, indoor maps, in general, have very limited coverage due to the lack or inaccessibility of digital maps. Second, the construction or update of an indoor map often requires extensive labor and time [1]. Finally, the pathway model extraction from an indoor map is a time-consuming and error-prone process. Thus, efficient and low-cost construction of an indoor pathway model for indoor localization requires further research.

Trajectory data with rich pedestrian behavior information can be efficiently collected and analyzed to infer the surrounding environment with high accuracy. Therefore, using trajectory data to automatically construct pathway models is well regarded as a promising approach. In the outdoor environment, the Global Positioning System (GPS) trajectory data has been widely used for the construction and updating of road network models [2]. Since GPS is unavailable in indoor environments, alternative technology needs to be explored. Currently, smartphones are equipped with a rich set of sensors including wireless radio receiving sensors (e.g., Wi-Fi and Bluetooth), inertial sensors (e.g., accelerometer and gyroscope) and environmental perception sensors (e.g., barometer and light) [3]. These sensors are able to provide a variety of mobile big data, which opens a new paradigm for the collection of indoor trajectory data.

Among the smartphone-assisted methods, Wi-Fi and Bluetooth are commonly used to collect trajectory data for indoor localization. Although these methods can achieve satisfactory accuracy for indoor environments, they typically require the prior collection of fingerprints, which is both labor-intensive and time-consuming. To collect low-cost indoor trajectory data collection, the built-in inertial sensors of smartphones have become a recent focus for researchers. The smartphone-based pedestrian dead reckoning (PDR) method is a typical example of using inertial sensors for indoor navigation. Despite its many advantages, this PDR method is prone to errors, particularly the error accumulated over time and distance. As a result, the trajectory data obtained by the smartphone typically cannot be directly used for the construction of indoor pathway models. This calls for effective methods to reduce trajectory errors.

While using indoor maps is a viable solution to alleviate the cumulative error of PDR, this approach is not considered in this work due to the aforementioned constraints with indoor maps. In the literature, alternative solutions have been developed to reduce the cumulative error by leveraging multiple localization methods. For example, Wi-Fi can be used to correct the error of PDR; In return, PDR is applied to smooth the trajectory data obtained by Wi-Fi [4]. However, the actual implementation of such multi-method solutions requires additional algorithms such as the particle filter, Kalman filter and Bayesian filter, which usually incurs large hidden computational cost. This drawback defeats our goal of obtaining low-cost and convenient trajectory information.

Compared to the methods aforementioned, the crowdsourcing technology provides a convenient way to obtain trajectory data and subsequently, a new approach for constructing the
indoor model. The advantage of this technology is that it can accomplish the sensing and computing tasks through the unintended participation of regular pedestrians. In the field of indoor location services, crowdsourcing technology using smartphones can be applied to construct and update indoor wireless fingerprint maps and floor plans at low cost. However, due to the complexity of the indoor environment and the diversity of pedestrians, it remains very challenging to obtain accurate indoor environment information through crowdsourced trajectory data.

In this paper, we propose a method that automatically constructs an indoor pathway model solely based on the crowdsourced trajectory data without requiring indoor maps. As a result, the proposed method is considered more practical as compared to the conventional map-dependent methods. More specifically, we propose to use the built-in smartphone inertial sensors to obtain pedestrian trajectory data followed by the pedestrian location estimation. In addition, we propose to first utilize human activity recognition (HAR) to derive special nodes (such as elevators, stairs, and turns) involved in the trajectories before exploiting these special nodes for trajectory error correction. Our experimental results show that the constructed pathway model can accurately express structural information of an indoor environment while providing both geometric and semantic information to support accurate and efficient indoor localization.

II. Trajectory Information Acquisition

When a pedestrian walks with a mobile phone, the built-in inertial sensors record a series of data. By analyzing the data, we can derive various information about the pedestrians including the walking distance and direction. More specifically, the distance information is estimated by an accelerometer while the directional information is obtained by a gyroscope and a magnetometer. Furthermore, the semantic information can be further derived from the data. In the following, the details of each information acquisition are discussed.

A. Distance information

The distance information can be estimated through steps and step-lengths that can be obtained by analyzing the three-axis acceleration values during a pedestrian’s walking. In order to reduce the noise in the raw acceleration data, low-pass filtering is first applied to remove spurious peaks. Since the acceleration value exhibits periodic variation during pedestrian walking, the peak step detection algorithm can be utilized to detect steps [5]. Similarly, the magnitude of the acceleration change can be used to estimate the step length. The synthetic acceleration value \( a(t) \) is calculated as follows:

\[
a(t) = \sqrt{a^2_x(t) + a^2_y(t) + a^2_z(t)} - g,
\]

where \( a_x(t) \), \( a_y(t) \), and \( a_z(t) \) are the readings of an accelerometer in the \( x \), \( y \) and \( z \) axes at time \( t \), respectively. Furthermore, \( g \) represents the Earth’s gravity.

Denoted by \( \mu \) the pedestrian’s stride length parameter, the step length in Step \( k \) can be computed as

\[
l_k = \mu \sqrt{a_{\text{max}}(k) - a_{\text{min}}(k)},
\]

for \( 0 < k \leq N_{\text{step}} \) with \( N_{\text{step}} \) being the total number of steps. Furthermore, \( a_{\text{max}}(k) \) and \( a_{\text{min}}(k) \) stand for the maximum and minimum synthetic acceleration values in this step, respectively. Clearly, \( \mu \) is pedestrian-dependent. It is initially set to 0.5 and updated during the trajectory correction process.

B. Directional information

The directional information can be derived from the gyroscope and magnetometer in a smartphone. The gyroscope measures the relative angular changes while the magnetometer determines the direction through the magnetic field. Same as the usual directional information expression, “North”, “East”, “South” and “West” are extracted from the angular value to indicate trajectory direction. Despite that a magnetometer can directly obtain directional information, it is susceptible to the environment. In contrast, while a gyroscope is not affected by magnetic fields, it suffers from cumulative errors over long-term estimation. To overcome these drawbacks of each sensor, a gyroscope is utilized to obtain angular changes during walking while the magnetometer reading is exploited to estimate the initial direction and correct the gyroscope directional information. Thus, the angular value \( \theta \) of each step obtained by the gyroscope is given as follows:

\[
\theta = \sum_{i=1}^{N_{\text{sample}}} \rho_i \Delta t,
\]

where \( \rho_i \) and \( \Delta t \) stand for the angular velocity value of each sample and the sampling interval, respectively. Furthermore, the sampling frequency is set to 50 Hz based on experience, i.e. \( \Delta t \) is set to 0.02 s. Finally, \( N_{\text{sample}} \) represents the number of samples in a step. The position of k-th step \((x_k, y_k)\) can be calculated from the position of the previous point \((x_{k-1}, y_{k-1})\) according:

\[
\begin{align*}
x_k &= x_{k-1} + l_k \cos \theta_k \\
y_k &= y_{k-1} + l_k \sin \theta_k,
\end{align*}
\]

where \( l_k \) and \( \theta_k \) indicate the step length and angle of step \( k \), respectively.

C. Activity information

Different activities have different action types and behavior patterns, which is reflected in the recordings of inertial sensors. In this study, the readings from the inertial sensors are utilized as basic data to recognize some common indoor activities including standing, walking, taking the elevator, going up (or down) stairs, and opening a door. The primary processes for recognizing these activities include preprocessing, feature extraction, and classification. As shown in Figure 1, the proposed activity recognition approach includes two phases: The first phase uses some sample data to generate an activity codebook. After window segmentation and feature extraction processing, multiple feature pattern sequences, including acceleration mode\((P^{a}_1, P^{a}_2, \ldots, P^{a}_i)\), gyro-reflected mode \((P^{g}_1, P^{g}_2, \ldots, P^{g}_j)\), and magnetometer mode \((P^{m}_1, P^{m}_2, \ldots, P^{m}_k)\), can be obtained from the raw data. The
second phase recognizes the activity of new data through pattern matching.

The primary processes for recognizing these activities include preprocessing, feature extraction, and classification. As shown in Figure 1, the proposed activity recognition approach includes two phases: Phase I uses some sample data to generate an activity codebook while Phase II recognizes the activity of new data through pattern matching.

1) Phase I: In Phase I, window segmentation is first performed before the feature extraction processing. After that, multiple feature pattern sequences, including acceleration mode \((P^0_1, P^0_2, \cdots, P^0_m)\), gyro-reflected mode \((P^g_1, P^g_2, \cdots, P^g_m)\), and magnetometer mode \((P^m_1, P^m_2, \cdots, P^m_m)\), can be obtained from the raw data.

Preprocessing: The preprocessing process includes data de-noising and data segmentation. A low-pass filter is first used to remove noise and smooth data before a windowing technique is applied to segment sensor data. However, this segmentation method is not capable of distinguishing activity transition. In comparison, using some special events to segment data is more suitable for data with multiple activities. In this study, the special events are heel strike and toe-off in one step. As a result, the window size is determined by the number of samples produced by each step. The sliding window method is used when no steps are detected over a period of time. Figure 2 illustrates the proposed window segmentation method. In Figure 2, the red dot stands for the detected starting point of each step while the window between two consecutive red dots is considered to be an event window. Furthermore, Figure 2 also shows some intervals with no step events being detected within a sliding window.

2) Feature extraction: After segmentation, a feature extraction process is used to obtain the appropriate time or frequency features from each time window. Some simple and efficient features such as mean, standard deviation, and waveform indicators are selected for activity classification. Depending on the data characteristics, even a small number of features are sufficient to produce satisfactory classification. According to the characteristics of the sensors, different features are extracted as shown in Table I.

3) Phase II: Despite that data obtained by the sensors is pedestrian-dependent even for the same activity, each activity can be identified by its unique pattern. In Phase I, an activity codebook is established by analyzing the characteristics of common activities. In the codebook, each mode represents one activity with one or more feature pattern sequences. In Phase II, when new data is acquired, its features are derived and compared with the codebook before the corresponding activity information can be determined. This pattern matching computation can be executed in an optimized order for better computational efficiency. As a motion-sensitive sensor, accelerometers in mobile phones can be used to easily determine motion states. Therefore, the acceleration mode is first used to match the corresponding activity. For example, the standard deviation and mean of the window data can be used efficiently to distinguish between stationary and motion states. The peak pattern can assist in determining the activities of going up (or down) the stairs and taking the elevator up (or down). In addition, the unique variation of the magnetometer during door opening can be used to efficiently recognize door opening activities. In general, common indoor activity information can be obtained through these sensors.

Table I: Feature extraction

<table>
<thead>
<tr>
<th>Sensors</th>
<th>Features</th>
<th>Feature pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accelerometer</td>
<td>mean (P^1_1)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>standard deviation (P^2_2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>peaks (P^g_3)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>zero crossings (P^4_3)</td>
<td></td>
</tr>
<tr>
<td>Gyroscope</td>
<td>max (P^g_1)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>min (P^g_2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>sum (P^g_3)</td>
<td></td>
</tr>
<tr>
<td>Magnetometer</td>
<td>peak (P^m_1)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>valley (P^m_2)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>zero crossing (P^m_3)</td>
<td></td>
</tr>
</tbody>
</table>

With the acquisition of distance, direction and semantics information, a trajectory can be well expressed. However, it is worth noting that the expression of a trajectory is in a relative coordinate system as the initial position is unknown.
III. CONSTRUCTING A SEMANTIC-RICH INDOOR PATHWAY MODEL

Some similar trajectories can be selected from crowdsourced trajectories to construct an indoor pathway model. However, in practice, these trajectories are prone to large errors for long-distance estimation. To overcome this problem, a short-range trajectory clustering approach is developed to generate nodes.

To illustrate the concept of the proposed short-range trajectory clustering, we use an example shown in Figure 3 in which a total of 200 trajectories were selected in a building to construct a multi-floor pathway model. These trajectories contain the semantic information of “Taking the elevator to the fourth (or fifth) floor”. In Figure 3, the red star marker represents the elevator while the blue dots and green squares indicate the detection points of the turns and the doors in the trajectories, respectively. Some of the trajectories are unsatisfactory and require further corrections. Assuming that the short-range trajectory error estimated by the inertial sensors is small, our strategy is to obtain stable reference nodes through trajectory segments. The process of obtaining stable nodes is shown in Figure 4: First, we determine the node \( t_1 \) by calculating the average of the first turning node in all trajectories; Subsequently, the next special node is estimated using the trajectory segments starting at \( t_1 \), and the rest of the nodes can be determined in the same manner.

To further improve the accuracy of the estimated nodes, boxplots are used to detect outliers. The distances between the estimated nodes and the central node are input as the boxplot data. Figure 5 shows that outliers are detected at nodes \( t_2 \) and \( t_3 \); thus, the coordinates of reference nodes are recalculated without the outliers.

Some additional improvements can be achieved in the resulting pathway model. For instance, the turning angle is typically a right angle while the adjacent door points are usually aligned in a straight line. Combining these features, the complete pathway model constructed using the crowdsourced trajectories is shown in Figure 6 where the red nodes are the elevators, and the green and dark red nodes indicate the door and turning points, respectively. These improvements can significantly reduce the indoor trajectory error. We will further analyze the error of the constructed pathway model in the next section.

IV. EXPERIMENTAL RESULTS

A. Localization Error

1) Constructed model error: Figure 7(a) compares the constructed pathway model and the ground truth of one floor. The trajectories utilized for constructing this model start from the special elevator node \( e_1 \) as their reference point. It can be seen from the figure that the resulting model error grows with the distance from the reference point, which is particularly evidenced over the two regions marked by the red and blue dashed lines in the figure. In order to reduce the model error of these two regions, two special nodes, namely the stair node \( e_2 \) and elevator node \( e_3 \), are also utilized as reference starting points to generate part of the pathway model. Since the number of stair and elevator points on each floor is small, the relative position with reference to these reference points can be easily narrowed down and quickly determined. Subsequently, these models are combined by common connection points. According to the method described in Section III, multiple trajectories are employed to construct the models of these two regions as shown in Figure 7(b) and Figure 7(c). The
two models are associated with the model in Figure 7(a) by connecting points $t_1$ and $t_2$, respectively.

Furthermore, the error values can be calculated based on the distance between the estimated door nodes and their actual locations on the map. The cumulative distribution function (CDF) of the resulting cumulative error of the constructed model is shown in Figure 8, where the curve with cross markers indicates the model constructed by the trajectories with $e_1$ as the only reference point. In contrast, the curve with triangle markers stands for the results generated with reference points $e_1$ and $e_2$. Finally, the curve with circle markers represents the results generated with all three reference points, i.e. $e_1$, $e_2$ and $e_3$. Their average errors are 1.3 m, 1.06 m, 0.95 m, respectively. It should be borne in mind that only a small number of trajectories were used to generate Figure 8 without requiring additional external resources (e.g. indoor maps). Thus, these results are indeed very encouraging.

2) Assisted localization error: In addition to expressing the indoor structural information, the constructed pathway model can assist the correction of the new trajectory data. Figure 9(a) shows the raw trajectories from the elevator $e_1$ to 10 rooms while the results corrected by the constructed model are shown in Figure 9(b). The errors between the end points of the trajectories and the ground truth are shown in Figure 10. The blue solid and red dashed lines indicate the raw trajectory errors.

Fig. 7: The difference between the pathway model and the ground truth.

Fig. 8: Pathway model error.

Fig. 9: Raw trajectories and corrected trajectories.

Fig. 10: Trajectory errors.
and the corrected trajectory errors, respectively. The error is measured by the Euclidean distance between the estimated door nodes and the real door nodes. The results corrected by the constructed model have shown a distinct advantage over those derived from raw trajectories.

B. Discussions

In an indoor environment, many elements are not unique, such as stairs, elevators and rooms. Therefore, it is challenging to distinguish these elements in crowdsourced trajectory data. In the following, we discuss methods to reduce ambiguities of determining two key elements in the pathway model, namely floor and elevator determination.

![Image](53x360 to 166x444)
(a) The acceleration changes when the elevator goes from the 1st floor to the 2nd floor
(b) The acceleration changes when the elevator goes from the 1st floor to the 3rd floor
(c) The acceleration changes when the elevator goes from the 1st floor to the 4th floor
(d) The acceleration changes when the elevator goes from the 1st floor to the 5th floor

Fig. 11: Acceleration changes during the elevator up.

1) Floor determination: Deriving the floor information from a single trajectory involving elevators is very challenging. However, this problem can be resolved by analyzing a large number of elevator activities from the crowdsourced trajectories. Figure 11 shows the acceleration changes during the elevator moves up: in the beginning, the elevator accelerates from a standstill and the corresponding acceleration mode is “rise, steady, drop”; when the elevator is approaching its targeted floor, its acceleration mode becomes “drop, steady, rise” before it comes to a complete stop. According to the interval between these two modes, the floor can be determined. For example, Figure 11 shows four different scenarios when the elevator moves up from the first floor to different upper floors. In Figure 11(a) where the elevator went from the first floor to the second floor, the two acceleration modes were not well separated in time. In contrast, as shown by the red circle in the Figure 11(b)-(d), the time separation between the two acceleration modes prolonged as the destination floor gets higher.

2) Elevator determination: It is rather common to have multiple elevators in the same building. Figure 12 shows some trajectories that contain elevator activities with the star markers indicating elevator nodes. The directional and distance information reflected by the trajectory can be exploited to determine whether the pedestrians occupy the same elevator or not. For example, if the pedestrian is facing the west when getting off the elevator, it is highly likely that the pedestrian was on the elevator $e_3$, as shown in Figure 12(c). When the trajectories have the same directional information, the distance information of the rectangular area in Figure 12(a) and Figure 12(b) can be used to further determine which elevator the pedestrian has used.

Since elevators are used as special nodes in the trajectory, elevators in different locations are helpful for constructing accurate indoor models. In our experiments, elevators usually serve as either starting or ending points of trajectories to construct an indoor model. Moreover, the trajectory errors increase with distance. Therefore, using different elevator points to estimate the surrounding node locations can result in a more accurate indoor model.

V. CONCLUSION

The indoor pathway model with accurate indoor spatial structure information plays an important role in enabling indoor location-based IoT applications. In this work, a crowdsourced semantic indoor pathway model has been proposed. The model can be automatically constructed from data collected from the smartphone built-in inertial sensors without requiring indoor maps and additional devices. In particular, since special nodes such as elevators and stairs are critical components in the model, HAR has been proposed to extract these special nodes from the pedestrian trajectory. Our extensive experimental results have shown that the constructed pathway model can express the indoor space with significantly improved accuracy.

REFERENCES