WIO-EKF: Extended Kalman Filtering-based Wi-Fi and Inertial Odometry Fusion Method for Indoor Localization

Pan Zhou#, Hao Wang#, Raffaele Gravina, *Senior Member, IEEE,* and Fangmin Sun*, *Member, IEEE,*

Abstract—Indoor location and navigation technologies are crucial for healthcare, security and other location-based services. Wi-Fi and inertial sensors have become mainstream indoor localization technologies for wearable device platforms due to simple deployment and low cost. This study proposes an extended Kalman filtering (EKF)-based multimodal sensor fusion algorithm for indoor localization, combining Wi-Fi fingerprint and inertial measurement unit (IMU) data to provide accurate and continuous pedestrian localization. The main contributions of this work are threefold. Firstly, a Wi-Fi fingerprint data augmentation method based on Access Point (AP) location sorting is proposed and a regression network model with a convolutional denoising autoencoder for WiFi-based indoor localization (CDAELoc) is designed to improve the robustness. Secondly, a dual-branch deep inertial odometry (DbDIO) network model for IMU-based indoor localization is introduced, consisting of two branches with various convolutional kernel sizes to extract features at different scales. Finally, an EKF-based Wi-Fi and Inertial Odometry (WIO-EKF) fusion localization system is presented, utilizing the predicted results from the proposed CDAELoc and DbDIO models as the system observations and mitigating the initial heading error of DbDIO. The proposed models are applied to the UJIIndoorLoc, RoNIN public datasets and self-collected dataset. Experimental results prove that the proposed CDAELoc model outperforms other Wi-Fi localization models, reducing the average positioning error by 12.5%. The proposed DbDIO model achieves higher accuracy and requires fewer model parameters than any other deep inertial odometry model. Finally, the average positioning error of WIO-EKF is lower than those of CDAELoc and DbDIO by 34% and 42%.

Index Terms—Wi-Fi fingerprinting, inertial measurement unit, indoor localization, extended Kalman filter, deep learning.

I. INTRODUCTION

INDOOR localization and navigation technologies are vital for many tasks, including emergency evacuation and rescue, augmented reality, and patient healthcare [1]. However, GPS signals are significantly attenuated in indoor environments, posing challenges to precise positioning. Therefore, various methods have been proposed to overcome this limitation including Wi-Fi, IMU, infrared (IF) and ultrasound (U/S) signals, Bluetooth, radio frequency identification (RFID), and cameras [2]. Each technology has its advantages and drawbacks in different application scenarios, and their development depends on considerations regarding deployment costs, positioning accuracy, stability, and security aspects.

Among them, U/S and IF technologies require specific signal reception devices not integrated into smartphones, hindering their widespread implementation. While Bluetooth and Near Field Communication (NFC) are available on most smartphones, they have limited ranges and require the deployment of additional hardware devices. Finally, smartphone cameras have low privacy security and high power consumption. Notably, modern smartphones are equipped with sensors such as a gyroscope, accelerometer, magnetometer, and Wi-Fi signal receivers. Considering the prevalence of smartphones and the extensive deployment of routers in public buildings, Wi-Fi and IMU have undeniable inherent advantages.

Wi-Fi indoor localization methods are mainly classified into geometric and fingerprinting methods. The former use triangulation techniques based on the angle of arrival, time of flight, or time of arrival to achieve positioning [3]. The latter methods are more accurate than geometric ones. The common fingerprint features include received signal strength (RSS) and channel state information (CSI) [4]. Compared to RSS, CSI is more stable and accurate, but it requires specific signal acquisition software for retrieval, which limits its application.

IMU-based classic positioning algorithms include physical methods (e.g., double integration [5]) and heuristic approaches [6], such as zero-velocity updates [7], [8] and pedestrian dead reckoning (PDR). The IMU currently equipped in smartphones is miniaturized and low-cost, but it has larger measurement errors, which can quickly amplify errors in physical algorithms. Besides, in practice, most users carry their smartphones in

Pan Zhou, Hao Wang, and Fangmin Sun are with the Joint Engineering Research Center for Health Big Data Intelligent Analysis Technology, Shenzhen Institute of Advanced Technology, Chinese Academy of Sciences, Shenzhen, China.

Raffaele Gravina is with the Department of Computer, Modeling, Electronics and Systems Engineering, University of Calabria, Italy.

Pan Zhou is also with the University of Chinese Academy of Sciences, Beijing, China.

#Pan Zhou and Hao Wang contributed equally to this work.

Corresponding author: Fangmin Sun, fm.sun@siat.ac.cn

unrestricted mode, which limits the applicability of heuristic algorithms. With the development of deep learning, data-driven deep inertial odometry (IO) has gained increasing attention [9], [10]. Deep IO utilizes the powerful feature extraction and data fitting capabilities of deep learning to derive results from IMU signals directly, offering significant advantages compared to classical algorithms.

In addition, the advantages and drawbacks of single indoor localization technologies are apparent, while the fusion of multimodal sensor data can largely compensate for their respective limitations. Wi-Fi and IMU fusion localization can provide continuous and more accurate positioning performance over a longer period. Currently, most studies focus on PDR/Wi-Fi fusion localization [11], but there is a lack of IO/Wi-Fi fusion methods of EKF.

This work adopts a deep learning algorithm to combine Wi-Fi fingerprinting localization with deep inertial odometry (DIO) by fusing via the extended Kalman filtering (EKF) to improve positioning performance and stability. The main contributions of this paper can be summarized as follows:

1) We propose a Wi-Fi fingerprint data augmentation strategy based on Access Point (AP) location sorting, effectively improving Wi-Fi positioning accuracy in multi-building scenarios and reducing data preprocessing complexity. The proposed CDAELoc model has higher noise robustness than traditional fingerprint-matching methods.

2) The proposed DbDIO model achieves the highest accuracy with the fewest parameters, making it more suitable for deployment on memory-constrained embedded devices.

3) We present a fusion localization system based on EKF that is robust to initial heading (yaw) errors of inertial odometry. This system provides accurate positioning results in our local dataset.

The rest of this paper is structured as follows: Section II describes and discusses related work on indoor localization. Section III introduces the proposed Wi-Fi fingerprinting network model, DIO, and the design of the EKF fusion system. Section IV describes the experiments and performance evaluation on public and local datasets. Finally, Section V draws concluding remarks and outlines future directions.

II. RELATE WORKS

Currently, indoor localization studies are classified into three types: Wi-Fi fingerprint-based methods, data-driven inertial odometry methods, and multi-sensor fusion methods. This section provides a comprehensive review of relevant indoor localization methods.

Wi-Fi fingerprint-based Indoor localization: Currently, most Wi-Fi-based fingerprinting methods utilize RSS as the fingerprint feature. The fingerprinting process typically involves two stages: offline and online. In the offline stage, a mapping between Wi-Fi fingerprints and corresponding location coordinates is established to construct a fingerprinting database. In the online stage, a fingerprint-matching algorithm determines the user's location. Researchers have proposed various localization algorithms to estimate the user's position. The K-Nearest Neighbor (KNN) algorithm, known for its low

complexity and does not require explicit training, has been widely used for fingerprint matching [4]. For example, Zeng [12] employed the Spearman distance as the weight for the weighted KNN algorithm. Compared to traditional localization algorithms, they achieved a reduction of approximately 7% in average positioning error (APE). However, the KNN algorithm requires traversing the entire fingerprinting database, which can lead to a longer execution time when dealing with large datasets. Previous studies [13]–[15] addressed this issue by utilizing k-means to partition the localization area, reducing the computational load for the weighted KNN algorithm.

Furthermore, other machine learning methods, such as Random Forest [16] and Gaussian process regression [17], have also been applied to calculate positioning results. However, due to the variability of Wi-Fi fingerprints, traditional machine learning-based Wi-Fi fingerprinting localization algorithms tend to experience a significant performance deterioration [18]. With the advancement of deep learning, many researchers have applied deep learning techniques to Wi-Fi fingerprinting localization. Some studies have used recurrent neural networks (RNNs) to explore the potential temporal features of Wi-Fi data to improve positioning accuracy. Sahar and Han [19] adopted long short-term memory (LSTM) networks to construct deep regression models, achieving higher positioning accuracy than traditional KNN algorithms. Wang et al. [20] proposed a spatial-temporal localization algorithm combining residual networks with LSTM. Song et al. [21] also used convolutional neural networks (CNN) for indoor localization framework with Wi-Fi fingerprinting. However, data sparsity and temporal fluctuations in Wi-Fi fingerprints deteriorate the positioning performance of CNN. Other studies [3], [22] have addressed this issue by computing the Pearson correlation coefficient between each AP and reference point (RP) and transforming Wi-Fi fingerprints into a two-dimensional image format to mitigate the impact of data sparsity. Then, an appropriate CNN network structure was built to extract useful features from the two-dimensional data using convolutional layers. However, preprocessing Wi-Fi fingerprint data into two-dimensional images was found to be computationally intensive and required further improvement. Additionally, some studies have used autoencoders (AEs) [22]–[24] and denoising autoencoders (DAEs) [25] to extract robust features and improve positioning accuracy. Alitaleshi et al. [26] proposed a novel model by combining an extreme learning machine autoencoder (ELM-AE) with a two-dimensional CNN (EA-CNN), which is based on Wi-Fi fingerprinting and deep learning. The performance of system is evaluated on the public datasets, demonstrating that EA-CNN outperforms conventional CNN methods by significantly reducing average positioning error. Zhao et al. [27] proposed a fingerprint-based localization method for multi-floor indoor environments within multi-building settings. which combines a gradient-boosting neural network (GrowNet) with a LSTM network to effectively capture the nonlinear relationships between RSS fingerprints and spatial locations.

Data-driven inertial odometry (IO): In the field of inertial positioning, more researchers are utilizing deep learning to extract the mapping relationship between inertial signals

and user positions, achieving more accurate and real-time localization. Recent studies have demonstrated that data-driven IO can achieve high-precision 2D path reconstruction using only smartphone IMU data. Chen et al. [28] employed a LSTM network to regress distance and angular changes within predefined time windows, achieving superior prediction performance than traditional inertial navigation systems and pedestrian dead reckoning (PDR). Herath et al. [29] provided the largest publicly available dataset called RoNIN, exceeding 42.7 hours of data from 100 subjects. They applied the average velocity 2D vector as the regression target and compared the localization performance of LSTM, residual network (ResNet), and temporal convolutional network (TCN). Among these networks, ResNet achieved the best localization accuracy. Liu et al. [30] presented a deep learning method for real-time IMU calibration, which significantly improves inertial odometry accuracy by learning sensor errors from high-precision data. Deng et al. [31] proposed a data-driven inertial navigation system that integrates a CNN-EKF for orientation and a CNN-Transformer for translation, achieving superior accuracy compared to existing methods. Wang et al. [32], [33] used the research findings [18] to propose a random orientation initialization method that reduces reliance on unreliable smartphone orientation estimation. They also optimized the model prediction targets to include the average velocity scalar and trigonometric function values. Their approach yielded the best localization results on the RoNIN public dataset. Current research focuses more on improving localization accuracy while neglecting the size of the parameters in data-driven IO network models.

Indoor localization via multimodal sensor fusion: Multimodal sensor fusion is widely used to compensate for the limitations of individual techniques and improve localization accuracy [34]. Commonly used fusion methods include Kalman filtering (KF) and particle filtering (PF). Alwin et al. [35] proposed a smartphone camera and IMU fusion localization system based on a linear KF, achieving an APE of 0.069 m. Ning et al. [36] used a KF to fuse Bluetooth and multiple IMU data, resulting in an APE of 0.8 m. Feng et al. [37] presented an adaptive IMU/UWB fusion method for indoor positioning that uses SVM to detect non-line-of-sight (NLOS) and improves accuracy over traditional methods. However, using cameras introduces such issues as high power consumption and privacy concerns.

In contrast, wireless localization technologies like Bluetooth require deploying numerous external devices in indoor environments, and UWB systems can be costly and complex to deploy. Therefore, the fusion of Wi-Fi and IMU has distinct advantages. Carrera et al. [38] utilized a particle filter to fuse Wi-Fi and IMU signals, achieving an APE of 1.01 m on their collected dataset. Zou et al. [39] also used a PF to fuse Wi-Fi and IMU signals. Still, they additionally incorporated iBeacon data to correct the drift error of PDR in areas with poor Wi-Fi signal coverage. Abdellatif et al. [40] utilized the PF algorithm to integrate IMU data with RSSI, enhancing the accuracy of indoor localization by leveraging PDR and magnetic fingerprinting techniques, and achieving an enhanced accuracy of 96.32% on their dataset. Nurpeiissov et al. [41]

developed an end-to-end sequence learning model based on LSTM to fuse Wi-Fi and IMU signals, and their model achieved an APE of 1.1 m on their self-built dataset. Herath et al. [6] developed Fusion-DHL, employing Wi-Fi-based least squares optimization and floor plan-based CNN networks to align and optimize the motion paths generated by the inertial odometer, respectively. Xu et al. [42] introduced SeqIPS, a deep learning-based indoor positioning system that fuses Wi-Fi and IMU data, uses a co-teaching network to handle noisy labels and a GAN-based domain adaptation module for IMU data labeling. SeqIPS achieved an average localization error of 3.37 m.

In summary, fusing data from multiple sensors exhibits higher localization accuracy compared to single sensor technologies. However, existing studies have mainly focused on fusing PDR and Wi-Fi, and comprehensive solutions for fusing deep inertial odometry and Wi-Fi are lacking. Due to the susceptibility of PDR methods to missed and false detections in the gait detection process, coupled with the unreliability of stride length estimation, we propose the DbDIO model, which significantly reduces the number of model parameters while ensuring performance. Regarding fusion positioning research, particle filtering has strong adaptability to nonlinear systems and minimal error accumulation. However, it faces challenges such as slow computation speed and difficulty in particle selection. Conversely, the EKF algorithm is simple and requires fewer computational resources. Therefore, this paper adopts EKF as the fusion method and a reasonable state equation, using Wi-Fi fingerprint localization results and deep inertial odometry outputs as observation values for the fusion system to estimate more accurate positioning results.

III. METHOD AND MATERIALS

A. Dataset

UJIIndoorLoc dataset [43]: It provides a standardized dataset for various Wi-Fi positioning algorithms. The dataset includes fingerprint data from three buildings, but no test samples are available. This paper subdivides the training samples into a training set and a validation set in an 8:2 ratio. All validation samples are used as the test set. The time interval between data collection for the training and test sets is ten days. According to [44], data from phone 17 and irrelevant APs were excluded, resulting in a total of 465 APs used in this study. This dataset is referred to as Dataset1 in this paper and is used for training and validating the proposed CDAELoc model.

RoNIN dataset: It is the largest inertial navigation dataset to date, with a total collection duration exceeding 42.7 hours. The dataset includes IMU data collected from smartphones and true 3D motion data outputted by Tango, an augmented reality computing platform. This allowed application developers to create user experiences that included indoor navigation, physical space measurement and augmented reality in a virtual world. Similar technologies include ARCore [45], ARKit [46], and AREngine [47]. The experiment participants could carry the smartphones naturally during daily activities. Due to safety and privacy concerns, only approximately 50% of the data

from RoNIN have been released. This dataset is referred to as Dataset2 in this paper and is used for training and validating the proposed DbDIO model.

Local dataset: We collected real-world Wi-Fi and IMU signals on the SIAT campus. The entire data collection process was subdivided into two phases. Phase 1: only Wi-Fi signals were collected and used to train the proposed CDAELoc network model. A laser rangefinder provided the coordinates of RPs. A total of 190 RPs were selected, and the data were collected for approximately 25-30 s at each RP with a sampling frequency of 1 Hz. Phase 2: Wi-Fi and IMU signals were synchronously collected and used to validate the performance of the proposed CDAELoc and DbDIO models in real-world scenarios. The ground truth data were obtained from AREngine. This dataset was referred to as Dataset3 and used to evaluate the positioning performance of the proposed WIO-EKF. The dataset acquisition location and reference points coordinate distribution is shown in Fig. 1.
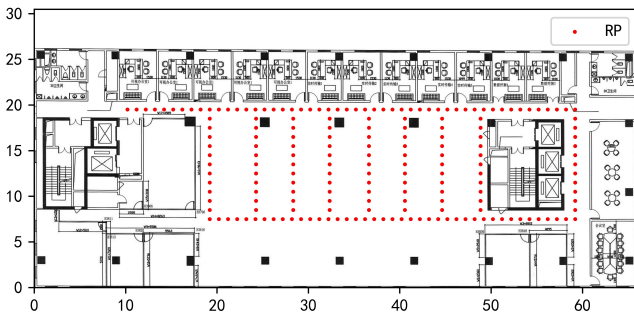

Fig. 1. Local dataset acquisition location and reference points coordinate distribution.

B. WiFi-Based Indoor Location

Firstly, we need to process the fingerprint data in the dataset so that the model can converge and achieve better positioning accuracy. The data processing flowchart is shown in Fig. 2. It mainly includes data augmentation and normalization.
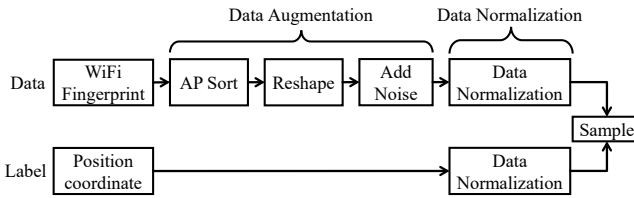

Fig. 2. Data processing flowchart.

1) Wi-Fi Fingerprint Data Augmentation:

AP Sort: The data sparsity problem of Wi-Fi fingerprint data will affect the model's performance. We want to use the AP sorting method to do data augmentation on Wi-Fi fingerprint data to form a globally sparse but locally dense data form. Assuming that the RP coordinates are known, we identify all RPs with RSS exceeding -110 dBm for AP_i (i.e., the i-th AP) and then calculate the average coordinates of RPs as the approximate coordinates for AP_i. Next, we sort

the APs based on their coordinates in a specific direction, transforming the original Wi-Fi fingerprints into data with spatial information about APs.

Reshape: To utilize 2D convolution for fingerprint data processing, the dimensions of the data are changed from 1×m to 3×n, where m represents the number of APs. The UJIIndoorLoc dataset covers three buildings on the Jaume I University campus in Spain. After obtaining the location of each AP, it was observed that the number of APs in buildings 1, 2, and 3 is as follows: 173, 146 and 146. In order to keep the number of APs unchanged after the input Wi-Fi fingerprints undergo dimension reduction in the encoder and upsampling in the decoder, we supplement each building's APs to 176, filling the added data with -110 dBm. Subsequently, the dimensions are transformed into 3×176.

Add noise: To enhance the model noise resistance and robustness, this study introduces noise to the original Wi-Fi fingerprint data, including masking and Gaussian noises. Masking noise randomly sets a portion of the input data to -110 dBm, simulating signal variations caused by human body obstruction. Gaussian noise, on the other hand, is a type of noise that follows a normal distribution and is used to simulate small-scale signal fluctuations over time. This study first adds Gaussian noise with a mean of 0 and a standard deviation of 3 to the data where the RSS values are not -110 dBm. Then, masking noise is applied to randomly set 30% of the data in each fingerprint to -110 dBm.

2) Wi-Fi Fingerprint Data Normalization:

Data normalization is performed on the fingerprint data to ensure that the RSS values, which typically range from -110 to 0 dBm, are within a suitable range for model convergence.

$$RSS_i = 1 - \frac{RSS_i}{RSS_{min}} \tag{1}$$

where RSS_{min} represents the minimum RSS value, which in this paper is -110 dBm . After normalization, the distribution of Wi-Fi fingerprint data becomes more uniform and no longer exhibits bias, and the variance is relatively small [48], thereby accelerating network training.

Additionally, the position coordinates also need to be normalized.

$$Pos_j = \frac{Pos_j}{Max_{pos} - Min_{pos}} \tag{2}$$

where Max_{pos} and Min_{pos} are the maximum and minimum position coordinates in the fingerprinting database, respectively.

3) The proposed CDAELoc network:

The parameter settings of the proposed model, including the CDAE and regression module, are listed in Table I.

The CDAE module utilizes Wi-Fi fingerprint data with artificially added Gaussian noise and masking noises to reconstruct the original Wi-Fi fingerprints. This process helps extract robust features from noisy Wi-Fi fingerprints. The regression module takes the feature maps extracted by the encoder as input data, and the labels are the position coordinates from the fingerprinting database. The objective is to predict the user's location. Each max-pooling layer and the last convolutional

TABLE I
STRUCTURE AND PARAMETERS OF CDAELOC

Layer Type	Filter Count	Filter Size	Stride Value	Output Size
CDAE: Encoder				
Conv	32	2×2	1	3×176×32
MaxPool	/	1×2	/	3×88×32
Conv	16	2×2	1	3×88×16
CDAE: Decoder				
UpSample	/	1×2	/	3×176×16
Conv	1	2×2	1	3×176×1
Regression module				
Conv	32	2×3	1	3×88×32
MaxPool	/	1×2	/	3×44×32
Conv	32	2×3	1	3×44×32
MaxPool	/	1×2	/	3×22×32
Conv	24	3×1	1	1×19×24
Flatten	/	/	/	456
MLP	/	/	/	2

layer are followed by a batch normalization (BN) layer and a rectified linear unit (ReLU) layer. The multilayer perceptron (MLP) consists of six fully connected layers, with hidden neuron numbers of 128, 256, 128, 64, 32, and 2. Except for the last layer, each fully connected layer is followed by a BN layer. The experimental parameter settings are shown in Table II. the activation function of the full connection layer adopts $tanh$.

TABLE II
WI-FI FINGERPRINT LOCALIZATION EXPERIMENT PARAMETER SETTINGS

Hyperparameters	CDAE	Regression module
batch size	30	100
activation	ReLU	ReLU $tanh$
optimizer	Adam	Adam
learning rate (lr)	0.001	0.001
epochs	30	200
loss	MSE	MAE

C. IMU-Based Indoor Location

1) The proposed DbDIO network:

We drew inspiration from existing studies [33], [49], [50] and proposed a dual-branch deep inertial odometry (DbDIO) model. The overall architecture of the constructed network model is shown in Fig. 3.

The model takes normalized IMU data as input and predicts the average velocity within each time window as the label. The model consists of two branches, each composed of a 1D convolutional layer, a CBAM attention module [51], a BiLSTM, and a temporal attention (TA) module [33]. The convolutional kernel sizes differ between the two branches, allowing them to capture features of different scales. The
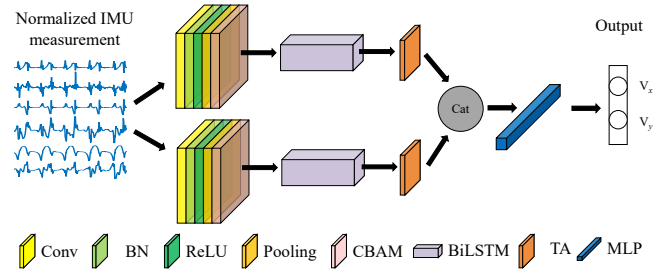

Fig. 3. The framework of the DbDIO model.

number of convolutional kernels is set to 64, and the stride is set to 2. The maximum pooling layer has a stride of 2. The CBAM attention module is used to refine the input features. Then, a BiLSTM is applied to each branch to extract long-term temporal features. The BiLSTM has two layers, with 64 hidden neurons in each layer, and a dropout rate of 0.4 to mitigate overfitting. The TA module calculates the weights for each time step by multiplying the feature vector of the last time step with the transpose of the entire feature map and applying the softmax function. The weights are then multiplied by the feature map and accumulated along the temporal axis. After concatenating the outputs of the two branches, three fully connected layers are used as the regression module, with 512, 512, and 2 hidden neuron numbers. A ReLU layer follows each fully connected layer except for the last layer, and a dropout rate of 0.5 is set. In addition, MSE is employed as the loss function to train the proposed DbDIO model. In addition, the epoch is set to 256, and the learning rate is set to 0.001.

2) IMU data preprocessing:

The IMU data processing proposed by DIO [32] was adopted in this paper. Specifically, the inertial signals are first rotated to the navigation coordinate system of the reference data collection device. Then, the normalized IMU data from the last and current time windows are extracted as the input for the model. The 2D average speed vector within the current time window is calculated as the label for the model. In this paper, time window is 200 (i.e. 1 s). Furthermore, the data augmentation method proposed by Wang et al [20] is also employed during model training. It involves adding random horizontal and gravity-aligned rotations to the IMU data, and applying the same horizontal rotations to the label values. Wand et al. [20] have demonstrated that predicting the velocity magnitude and the trigonometric values outperforms predicting the 2D average speed vector alone. However, this approach uses two separate, fully connected modules to predict velocity magnitude and trigonometric values, leading to an increased number of model parameters. In contrast, this paper focuses on predicting the 2D average speed vector; thus, the model only consists of a single fully connected module. Moreover, the number of hidden neurons in each layer of the BiLSTM is set to 64, significantly reducing the number of model parameters. The dual-branch framework of the DbDIO model enhances its performance.

D. Wi-Fi-IO Fusion localization using EKF

We use the CDAELoc network model to estimate global coordinates and the DbDIO network model to estimate relative motion paths. The fusion system incorporates an EKF to fuse the positioning results from both models. The DbDIO model provides accurate relative motion paths, but the initial heading of this path is unknown. This paper uses the first data from a mobile phone's built-in rotation vector sensor to perform a simple correction. However, the rotation vector sensor may be influenced by magnetic materials or may have inherent design flaws, resulting in an offset angle ψ between the measurement and the real heading. In this paper, we select the pedestrian's coordinates x, y, and the angle ψ as the system state vector:

$$X = \begin{bmatrix} x & y & \psi \end{bmatrix} \tag{3}$$

The observation vector is defined as Eq. (4):

$$Z = \begin{bmatrix} x^{wifi} & y^{wifi} & s^{imu} & t^{imu} \end{bmatrix}^T \tag{4}$$

where x^{wifi} and y^{wifi} represent the coordinates output by the CDAELoc model. s^{imu} and t^{imu} are the 2D displacement values obtained by accumulating the outputs of the DbDIO model. The system state equation is as Eq. (5):

$$
X_k = \begin{bmatrix} x_k \\ y_k \\ \psi_k \end{bmatrix}
$$
$$
= \begin{bmatrix} x_{k-1} + s_k^{imu} * \cos\psi_{k-1} - t_k^{imu} * \sin\psi_{k-1} \\ y_{k-1} + s_k^{imu} * \sin\psi_{k-1} + t_k^{imu} * \cos\psi_{k-1} \\ \psi_{k-1} \end{bmatrix} + W_{k-1} \tag{5}
$$

where W_{k-1} is the process noise. The Jacobian matrix can be represented as Eq. (6):

$$A = \begin{bmatrix} 1 & 0 & -s_k^{imu} * \sin\psi_{k-1} - t_k^{imu} * \cos\psi_{k-1} \\ 0 & 1 & s_k^{imu} * \cos\psi_{k-1} - t_k^{imu} * \sin\psi_{k-1} \\ 0 & 0 & 1 \end{bmatrix} \tag{6}$$

The observation matrix can be derived as Eq. (7):

$$H = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{7}$$

After the linearization of the system, the state equation and observation equation can be represented by Eq. (8) and Eq. (9), respectively.

$$
X_k = \begin{bmatrix} 1 & 0 & -s_k^{imu} * \sin\psi_{k-1} - t_k^{imu} * \cos\psi_{k-1} \\ 0 & 1 & s_k^{imu} * \cos\psi_{k-1} - t_k^{imu} * \sin\psi_{k-1} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ \psi_{k-1} \end{bmatrix} + W_{k-1} \tag{8}
$$

$$
Z_k = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_k \\ y_k \\ \psi_k \end{bmatrix} + V_k \tag{9}
$$

The covariance matrices Q and R corresponding to the process noise W_{k-1} and observation noise V_k, respectively, are expressed by Eq. (10) and Eq. (11) to derive V_k:

$$Q = \begin{bmatrix} 0.01 & 0 & 0 \\ 0 & 0.01 & 0 \\ 0 & 0 & \left(\frac{45}{360}\right)^2 \end{bmatrix} \tag{10}$$

$$R = \begin{bmatrix} \sigma_x^2 & 0 & 0 & 0 \\ 0 & \sigma_y^2 & 0 & 0 \\ 0 & 0 & \sigma_s^2 & 0 \\ 0 & 0 & 0 & \sigma_t^2 \end{bmatrix} \tag{11}$$

where σ_x^2 and σ_y^2 are the variances of Wi-Fi positioning, and σ_s^2 and σ_t^2 are the variances of relative displacement in the inertial positioning module. The initial value of the state covariance matrix P is not critical and will converge to the true state after iterations. In this paper, it is set as Eq. (12):

$$P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{12}$$

IV. Experimental Results

A. Evaluation metrics

This paper uses APE to evaluate the performance of the proposed CDAELoc mode. PE refers to the position error, derived as the Euclidean distance between the estimated and real positions. A smaller APE indicates better positioning accuracy.

Two standard metrics proposed by [52], namely the Absolute Trajectory Error (ATE) and Relative Trajectory Error (RTE), are used to quantitatively evaluate the performance of the DbDIO model. ATE is defined as the RMSE between the entire estimated path and the ground truth path. RTE is defined as the RMSE within fixed time intervals, with each interval set to 1 s (200 samples).

The cumulative distribution function (CDF) of the positioning error is used to evaluate the performance of the proposed WIO-EKF. The CDF curve has the positioning error on the x-axis and the cumulative probability values on the y-axis. A steeper rise in the CDF curve indicates smaller positioning errors for most samples. We also use APE, min PE (Min_PE), and max PE (Max_PE).

B. Experimental Results of the CDAELoc model

1) Different Model Structures and Data Types:

To validate the effectiveness of the proposed regression network model based on CDAE and the sorting-based data augmentation method, this study conducted experiments using sorted and unsorted fingerprint data on three different model structures: (1) noise+CDAE+CNN: adding noise to the training data and using CDAE, (2) noise+CNN: only adding noise to the training data, and (3) CNN: normal CNN network. The experiments were conducted on Dataset1 and Dataset3; the results are shown in Table III.

Two following main findings were observed. Firstly, the proposed model structure achieved the smallest APE overall,

reducing APE by approximately 19 and 5% on Dataset1 and Dataset3, respectively. Adding noise to the training data simulated the characteristics of Wi-Fi signals, and using CDAE enabled the learning of more robust features, reducing the fluctuation of Wi-Fi signals and the impact of random occlusions caused by human bodies. Secondly, after sorting AP, the APE values dropped by approximately 27% on Dataset1. However, the performance improvement was smaller on Dataset3, which may be attributed to Dataset3 only covering one building, while Dataset1 covered three buildings. Sorting the APs effectively differentiated the Wi-Fi fingerprint between different buildings.

TABLE III
PERFORMANCE EVALUATION OF DIFFERENT FRAMEWORK AND DATA PREPROCESSING TYPES

Dataset	Method	Unsorted	Sorted
Dataset1	CNN	8.23	7.72
	Noise+CNN	15.36	7.92
	Noise+CDAE+CNN	**7.95**	**7.37**
Dataset3	CNN	4.25	4.30
	Noise+CNN	**3.93**	3.94
	Noise+CDAE+CNN	3.98	**3.83**

2) Different masking noise parameters:

As the operation time of AP devices increases, some devices may malfunction or be replaced with other AP devices, resulting in a growing disparity between the historical data in the fingerprinting database and the data collected in the online phase. To simulate this process, this study added masking noise with different masking levels to the test data. The masking level ranged from 0 to 0.5 with a step size of 0.05. The comparative methods used in the study were k-nearest neighbors (KNN), decision trees (DT), and random forests (RF). Fig. 4 shows the experimental results. The results proved that the proposed CDAELoc method performed best among all masking levels. When the parameter was set to zero, the APEs on Dataset1 for CDAELoc, KNN, DT, and RF were 7.37, 10.05, 12.07, and 9.48 m, respectively. On Dataset3, the APEs were 3.83, 4.99, 5.78, and 4.68 m, respectively. As the parameter value increased, all algorithms exhibited a general trend of decreasing performance. However, the proposed CDAELoc model exhibited the least performance degradation. Among the other three methods, KNN performed the best, reducing the errors on Dataset1 and Dataset3 by 10.94 and 2.3 m, respectively, when the parameter was set to 0.3. In contrast, the proposed algorithm reduced the errors by 3.72 and 0.88 m on Dataset1 and Dataset3, respectively. The experimental results proved that the proposed CDAELoc model exhibited robust noise resistance. Even with faulty or replaced APs in the fingerprinting database, the algorithm could still provide high positioning accuracy.

3) Comparison with other algorithms on Dataset1:

Table IV shows the evaluation of the positioning accuracy of various algorithms.

The listed algorithms utilized CNNs or autoencoders to construct their models. Among the compared algorithms,
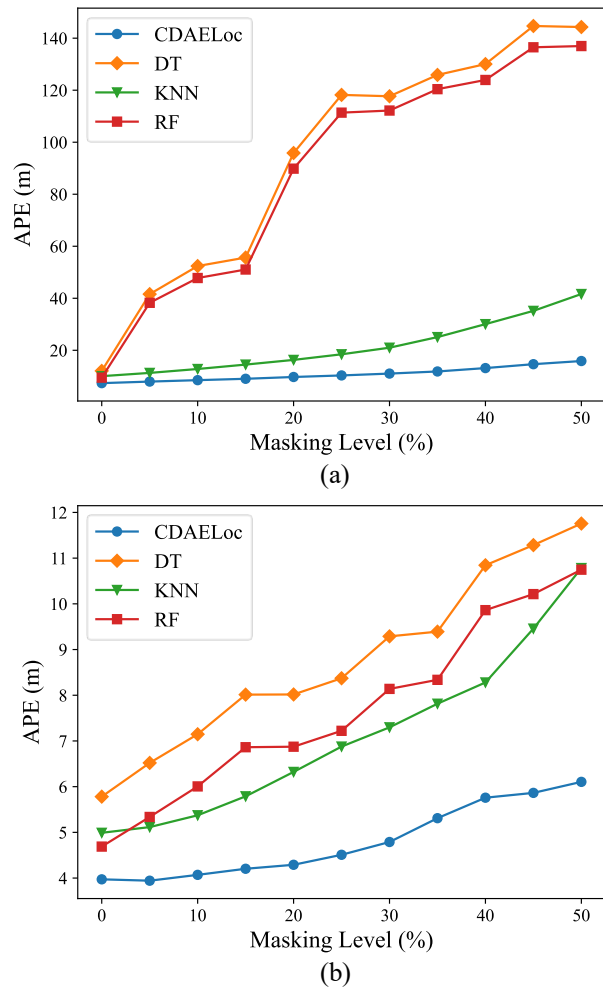

Fig. 4. CDF curves of different algorithms.

Tang's algorithm [15] achieved the best positioning accuracy among the autoencoder-based methods, with an APE of 8.42 m. CHISEL [13] achieved the best positioning accuracy among the CNN-based methods with an average error of 8.80 m. In contrast, the proposed algorithm in this study achieved an APE of 7.37 m, reducing the errors by 12.5% and 16.2% compared to those of Tang et al. [15] and CHISEL of Wang et al. [13], respectively. The experimental results demonstrate that the proposed CDAELoc algorithm outperforms other algorithms regarding positioning accuracy. Additionally, the proposed algorithm only requires a simple reordering of fingerprint data without complex data preprocessing. Moreover, the structure of the proposed CDAELoc model is simple, effectively improving the response speed of the algorithm.

C. Experimental Results of the DbDIO Model

1) Different branch numbers:

The proposed DbDIO model consists of two branches, each with a different convolution kernel size, allowing for the extraction of short-term features at different temporal scales. Table V presents the performance comparison between the single-branch and dual-branch network models. The table shows that the dual-branch network structure performs better

TABLE IV
COMPARISON WITH OTHER STUDIES ON DATASET1

Year	Studies	Model	APE (m)
2017	Scalable DNN [23]	SAE	9.29
2019	CNNLoc [21]	CNN	11.78
2021	CHISEL [22]	AE+CNN	8.80
2021	CCpos [25]	CDAE+CNN	12.4
2022	Tang et al. [24]	SAE+LSTM	8.42
2023	Alitaleshi et al. [26]	EA+CNN	8.34
2023	Zhao et al. [27]	GrowNet+ LSTM	8.49
	CDAELoc (This work)	**CDAE+CNN**	**7.37**

TABLE VI
PERFORMANCE COMPARISON WITH OTHER ALGORITHMS ON DATASET2

Metric	Method	Seen	Unseen	Mixed
	PDR	32.15	31.5	31.82
	PDR (Tango)	11.86	9.56	10.71
ATE (m)	R-RseNet	5.27	8.94	7.14
	DIO	**5.1**	8.19	6.67
	DbDIO (This work)	5.54	**7.81**	**6.67**
	PDR	26.21	25.18	25.69
	PDR (Tango)	9.14	7.03	8.09
RTE (m)	R-RseNet	4.22	6.68	5.47
	DIO	**3.68**	6.03	4.87
	DbDIO (This work)	3.84	**5.78**	**4.81**

overall, especially in the ATE metric, where the dual-branch structure shows an average reduction of 11.35%.

TABLE V
PERFORMANCE EVALUATION OF DIFFERENT NUMBERS OF BRANCHES

Metric	Number of branches	Dataset2	Dataset3
ATE (m)	Single	7.33	5.8
	Dual	**6.67**	**4.97**
RTE (m)	Single	4.95	**3.61**
	Dual	**4.81**	4.03

TABLE VII
COMPARISON OF MODEL COMPLEXITY

Method	Parameters	Model size	FLOPs
R-ResNet	4,634,882	17.76MB	38.25 M
DIO	6,143,747	23.52MB	73.6 M
DbDIO (This work)	**732,192**	**2.81MB**	**34.87 M**

In the table, FLOPs refer to the number of floating-point operations, where a smaller value indicates a lower computational load for the model. The table shows that the proposed model has the smallest number of parameters, model size, and FLOPs. Compared to the DIO model, the proposed model reduces the number of parameters by 88% and has a model size of only 2.81. This indicates that the model can be easily deployed on memory-constrained embedded devices or mobile devices.

D. WiFi-IO Fusion localization using EKF

1) Performance comparison between fusion systems and single techniques:

Fig. 5 shows the cumulative distribution function curves of positioning errors for the CDAELoc, DbDIO and WIO-EKF systems on Dataset3. As seen in Fig. 5, the probability of positioning errors of a single positioning technology was below 25% within 2 m, below 60% within 4 m, and over 30% for distances exceeding 5 m. In contrast, the WIO-EKF system achieved the following probabilities of positioning errors: about 40% within 2 m, below 85% within 4 m, and 6% for positioning over 5 m. These results indicate that the proposed WIO-EKF system exhibits higher positioning accuracy than the other two single techniques and achieves the best performance in terms of positioning accuracy.

Table VIII presents the statistical summary of the average, minimum, and maximum positioning errors of the three localization techniques. Since both the DbDIO model and the WIO-EKF system have their initial coordinates set to the real pedestrian coordinates, the Min_PE metric is not calculated. The WIO-EKF system exhibits an APE of 2.53 m, outperforming the CDAELoc and DbDIO models by about 34% and 42%, respectively. The WIO-EKF also significantly

2) Comparison with other algorithms on Dataset3:

This study compared the proposed DbDIO model with other deep learning algorithms on Dataset2, and the comparison results are presented in Table VI. PDR and PDR (Tango) are both PDR methods. Still, they differ in terms of the source of input data. PDR utilizes data from a handheld mobile phone, while PDR (Tango) utilizes data from a Tango mobile phone. Both methods have a step length set to 0.67 m, while the direction estimation data and gait detection data are sourced from the mobile phone's game rotation vector sensor and step counter sensor, respectively. Compared to PDR, PDR (Tango) demonstrates superior performance. This is because the Tango phone is tightly bound to the user's body, allowing the rotation vector information outputted by the phone to approximate the user's posture changes. In contrast, a handheld phone can be freely swung during data collection, resulting in significant deviations in direction information from the user's true heading. Incorrect heading data leads to a significant decrease in algorithm performance. The PDR (Tango) algorithm also requires the device to remain relatively fixed to the user's body, while the proposed algorithm, ResNet, and DIO utilize IMU data from handheld devices. The user has more freedom to move the phone, resulting in fewer constraints and higher performance. Among these five methods, the proposed DbDIO algorithm achieved the best performance in almost all results, particularly on unseen data. Compared to the best-performing DIO algorithm among the other methods, the proposed DbDIO algorithm achieved a reduction of 4.52 and 4.15% in ATE and RTE metrics, respectively.

Table VII presents a complexity comparison between the proposed model and two other models.

Fig. 5. CDF curves of different algorithms.

Fig. 6. Comparison of trajectories of different algorithms in a specific sample.

TABLE VIII
PERFORMANCE COMPARISON OF DIFFERENT ALGORITHMS

Method	APE (m)	Min_PE (m)	Max_PE (m)
CDAELoc (Wi-Fi)	3.83	0.09	15.01
DbDIO (IMU)	4.39	-	14.98
WIO-EKF (fusion)	**2.53**	-	**7.48**

reduces Max_PE. The Max_PE for DbDIO is 14.98m, which can be attributed to the accumulation of errors during path calculation by inertial odometry and significant offset in initial heading estimation using the rotation vector sensor.

Fig. 6 shows the computed localization results for a data sequence in different algorithms. The Wi-Fi fingerprint localization results are somewhat scattered and discontinuous, but they are generally distributed around the actual trajectory; the trajectory calculated by the inertial odometry module is very similar in shape to the real trajectory, indicating that the deep inertial odometry network model proposed in this paper has high performance. However, due to the large error between the initial heading output by the rotation vector sensor and the real heading, the entire trajectory deviates significantly from the real trajectory, with an average positioning error of about 8.8 m; from the Fig 6, it can be seen that the EKF fusion localization results have a high degree of overlap with the real trajectory, with an average positioning error of about 1.6 m, which is an 81.8% improvement compared to the inertial odometry, indicating that the EKF fusion system can effectively reduce the impact of the initial heading error on positioning performance.

2) The impact of the initial heading error of odometry:

To study the robustness of the WIO-EKF system to initial heading errors, this paper used different offset values to the initial heading of the DbDIO. The offset values ranged from -180 to 180 degrees, with a step size of 10 degrees. The counterclockwise direction was considered positive. Fig. 7 shows the experimental results: as the offset value increased, the APE of DbDIO initially dropped and then grew. The best
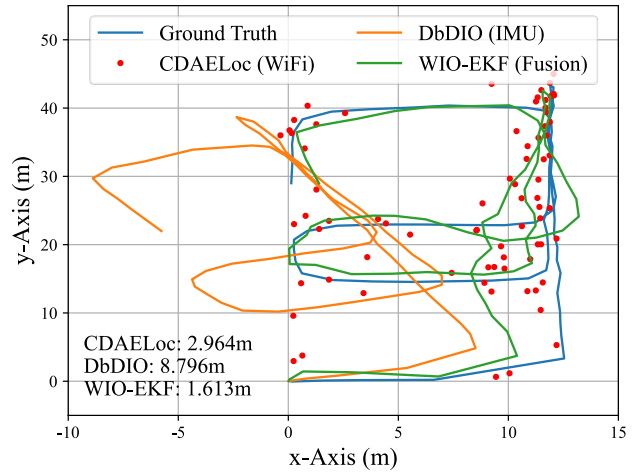
positioning performance was achieved at an offset value of about 0 degrees, and the positioning error exhibited significant variation. However, the CDAELoc value did not affect the offset model and consistently maintained a stable positioning error. The WIO-EKF system provided the best positioning for offset values ranging from -150 to 150 degrees. Overall, its positioning error was smaller than that of the CDAELoc model. However, when the offset angle exceeded 150 degrees or was less than -150 degrees, the WIO-EKF positioning performance was inferior to that of CDAELoc. This was primarily due to the time required for the WIO-EKF system to adjust, resulting in a performance loss in the early stages of localization.

Fig. 8 shows a path comparison of one sequence with clockwise heading angle offsets of -120°, 60°, 0°, 60°, 120°, and 180°. It can be observed that the six paths calculated by the WIO-EKF system exhibited slight differences only at the beginning, while the later paths were essentially consistent.
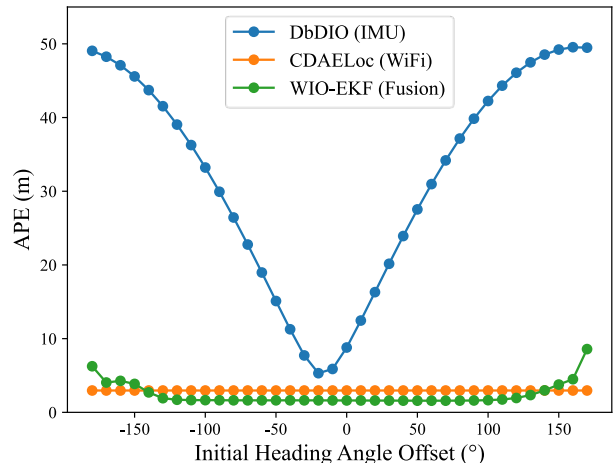

Fig. 7. Performance evaluation of different initial yaw angle offsets of the DbDIO network.

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2024.3386889
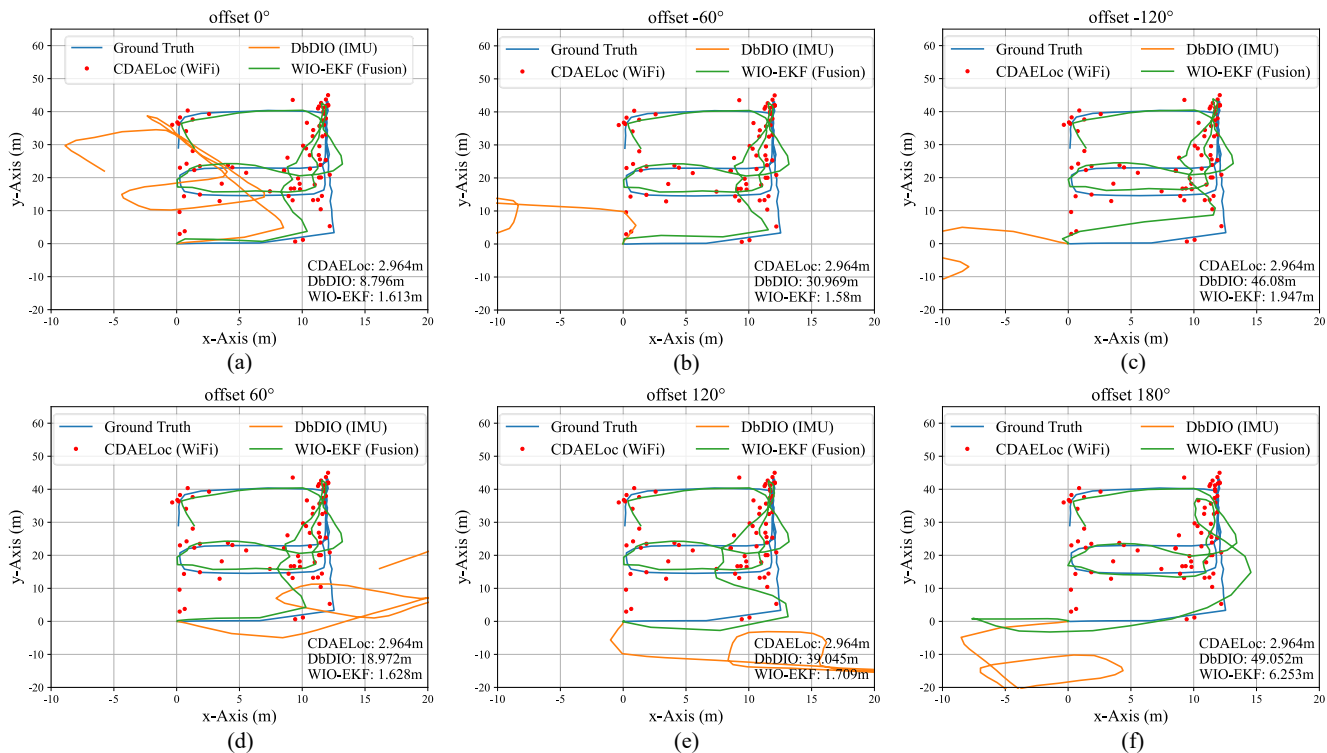
10

Fig. 8. Path comparison of one sequence with different offsets of the initial heading angle: (a) 0°; (b) -60° (c) -120°; (d) 60°; (e) 120°; (f) 180°.

V. CONCLUSION

In the context of Wi-Fi-based localization, this study proposed a Wi-Fi fingerprint data augmentation method, which augmented the original Wi-Fi fingerprints by incorporating coarse AP spatial information, thereby enriching the spatial characteristics of the data. Compared to conventional machine learning methods, the proposed CDAELoc network model can provide reliable localization results even when some APs fail or are replaced. In IMU-based localization, this paper proposed DbDIO model, utilizing two branches with different convolution kernel sizes to extract features of varying temporal scales. Each branch comprised a convolutional layer and a BiLSTM layer to capture short-term and long-term temporal features. The DbDIO network outperformed those of other available studies with a BiLSTM layer to capture short-term and long-term temporal features, providing a parameter reduction of up to 88%. This study proposed a Wi-Fi fingerprinting and deep inertial odometry fusion algorithm based on EKF for multimodal sensor fusion localization. The proposed WIO-EKF system utilized the predictions of CDAELoc and DbDIO as the system's observations. Experimental results proved that the fusion algorithm achieved a lower APE than Wi-Fi fingerprinting and inertial odometry alone, with 34% and 42% improvements, respectively. Moreover, the WIO-EKF system exhibited robustness to initial heading errors of the inertial odometry, implying a certain level of independence from magnetometers, which was not achievable by conventional PDR/Wi-Fi fusion localization.

Compared to other models, the proposed one significantly reduced the parameter count by an order of magnitude. How-

ever, as the reduction in FLOPs was slight, running DbDIO on resource-constrained smart devices still poses a challenge. Future work will apply lighter data-driven inertial odometry models to three-dimensional multimodal sensor fusion and real-time indoor positioning systems.

REFERENCES

[1] I. Ashraf, M. Kang, S. Hur, and Y. Park, "Minloc: Magnetic field patterns-based indoor localization using convolutional neural networks," *IEEE Access*, vol. 8, pp. 66 213–66 227, 2020.

[2] J. Kunhoth, A. Karkar, S. Al-Maadeed, and A. Al-Ali, "Indoor positioning and wayfinding systems: a survey," *Human-centric Computing and Information Sciences*, vol. 10, no. 1, pp. 1–41, 2020.

[3] A. Mittal, S. Tiku, and S. Pasricha, "Adapting convolutional neural networks for indoor localization with smart mobile devices," in *Proceedings of the 2018 on Great Lakes Symposium on VLSI*, 2018, pp. 117–122.

[4] Y. Dong, T. Arslan, and Y. Yang, "An encoded lstm network model for wifi-based indoor positioning," in *2022 IEEE 12th International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2022, pp. 1–6.

[5] H. Yan, Q. Shan, and Y. Furukawa, "Ridi: Robust imu double integration," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 621–636.

[6] S. Herath, S. Irandoust, B. Chen, Y. Qian, P. Kim, and Y. Furukawa, "Fusion-dhl: Wifi, imu, and floorplan fusion for dense history of locations in indoor environments," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 5677–5683.

[7] J.-O. Nilsson, I. Skog, P. Händel, and K. Hari, "Foot-mounted ins for everybody-an open-source embedded implementation," in *Proceedings of the 2012 IEEE/ION Position, Location and Navigation Symposium*. Ieee, 2012, pp. 140–145.

[8] C. Kilic, N. Ohi, Y. Gu, and J. N. Gross, "Slip-based autonomous zupt through gaussian process to improve planetary rover localization," *IEEE robotics and automation letters*, vol. 6, no. 3, pp. 4782–4789, 2021.

[9] C. Chen, P. Zhao, C. X. Lu, W. Wang, A. Markham, and N. Trigoni, "Deep-learning-based pedestrian inertial navigation: Methods, data set, and on-device inference," *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4431–4441, 2020.

This article has been accepted for publication in IEEE Internet of Things Journal. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2024.3386889

11

[10] W. Liu, D. Caruso, E. Ilg, J. Dong, A. I. Mourikis, K. Daniilidis, V. Kumar, and J. Engel, "Tlio: Tight learned inertial odometry," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 5653–5660, 2020.

[11] J. Yu, Z. Na, X. Liu, and Z. Deng, "Wifi/pdr-integrated indoor localization using unconstrained smartphones," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, no. 1, pp. 1–13, 2019.

[12] Z. Jiangtao and W. Honglan, "An improved wknn algorithm using in indoor positioning," in *2019 6th International Conference on Information Science and Control Engineering (ICISCE)*. IEEE, 2019, pp. 136–140.

[13] L. Siyang, R. de Lacerda, and J. Fiorina, "Wknn indoor wi-fi localization method using k-means clustering based radio mapping," in *2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring)*. IEEE, 2021, pp. 1–5.

[14] B. Wang, "Research on indoor positioning method with smartphone based on wifi and inertial sensors (in chinese)," Ph.D. dissertation, Harbin Engineering University, 2021.

[15] J. Torres-Sospedra, D. Quezada-Gaibor, G. M. Mendoza-Silva, J. Nurmi, Y. Koucheryavy, and J. Huerta, "New cluster selection and fine-grained search for k-means clustering and wi-fi fingerprinting," in *2020 International Conference on Localization and GNSS (ICL-GNSS)*. IEEE, 2020, pp. 1–6.

[16] Y. Wang, C. Xiu, X. Zhang, and D. Yang, "Wifi indoor localization with csi fingerprinting-based random forest," *Sensors*, vol. 18, no. 9, p. 2869, 2018.

[17] M. S. Anwar, F. Hossain, N. Mehajabin, M. Mamun-Or-Rashid, and M. A. Razzaque, "A comparative study on gaussian process regression-based indoor positioning systems," in *2018 International Conference on Innovation in Engineering and Technology (ICIET)*. IEEE, 2018, pp. 1–5.

[18] H. Rizk, M. Torki, and M. Youssef, "Cellindeep: Robust and accurate cellular-based indoor localization via deep learning," *IEEE Sensors Journal*, vol. 19, no. 6, pp. 2305–2312, 2018.

[19] A. Sahar and D. Han, "An lstm-based indoor positioning method using wi-fi signals," in *Proceedings of the 2nd International Conference on Vision, Image and Signal Processing*, 2018, pp. 1–5.

[20] R. Wang, H. Luo, Q. Wang, Z. Li, F. Zhao, and J. Huang, "A spatial–temporal positioning algorithm using residual network and lstm," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 11, pp. 9251–9261, 2020.

[21] X. Song, X. Fan, C. Xiang, Q. Ye, L. Liu, Z. Wang, X. He, N. Yang, and G. Fang, "A novel convolutional neural network based indoor localization framework with wifi fingerprinting," *IEEE Access*, vol. 7, pp. 110 698–110 709, 2019.

[22] L. Wang, S. Tiku, and S. Pasricha, "Chisel: Compression-aware high-accuracy embedded indoor localization with deep learning," *IEEE Embedded Systems Letters*, vol. 14, no. 1, pp. 23–26, 2021.

[23] K. S. Kim, S. Lee, and K. Huang, "A scalable deep neural network architecture for multi-building and multi-floor indoor localization based on wi-fi fingerprinting," *Big Data Analytics*, vol. 3, pp. 1–17, 2018.

[24] Z. Tang, S. Li, K. S. Kim, and J. Smith, "Multi-output gaussian process-based data augmentation for multi-building and multi-floor indoor localization," in *2022 IEEE International Conference on Communications Workshops (ICC Workshops)*. IEEE, 2022, pp. 361–366.

[25] F. Qin, T. Zuo, and X. Wang, "Ccpos: Wifi fingerprint indoor positioning system based on cdae-cnn," *Sensors*, vol. 21, no. 4, p. 1114, 2021.

[26] A. Alitaleshi, H. Jazayeriy, and J. Kazemitabar, "Ea-cnn: A smart indoor 3d positioning scheme based on wi-fi fingerprinting and deep learning," *Engineering Applications of Artificial Intelligence*, vol. 117, p. 105509, 2023.

[27] Y. Zhao, W. Gong, L. Li, B. Zhang, and C. Li, "An efficient and robust fingerprint based localization method for multi floor indoor environment," *IEEE Internet of Things Journal*, 2023.

[28] C. Chen, X. Lu, A. Markham, and N. Trigoni, "Ionet: Learning to cure the curse of drift in inertial odometry," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[29] S. Herath, H. Yan, and Y. Furukawa, "Ronin: Robust neural inertial navigation in the wild: Benchmark, evaluations, & new methods," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 3146–3152.

[30] H. Liu, X. Wei, M. Perusquía-Hernández, N. Isoyama, H. Uchiyama, and K. Kiyokawa, "Duet: Improving inertial-based odometry via deep imu online calibration," *IEEE Transactions on Instrumentation and Measurement*, 2023.

[31] X. Deng, S. Wang, C. Shan, J. Lu, K. Jin, J. Li, and Y. Guo, "Data-driven based cascading orientation and translation estimation for inertial navigation," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 3381–3388.

[32] Y. Wang, H. Cheng, C. Wang, and M. Q.-H. Meng, "Pose-invariant inertial odometry for pedestrian localization," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–12, 2021.

[33] Y. Wang, H. Cheng, and M. Q.-H. Meng, "A2dio: Attention-driven deep inertial odometry for pedestrian localization based on 6d imu," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 819–825.

[34] R. Gravina, P. Alinia, H. Ghasemzadeh, and G. Fortino, "Multi-sensor fusion in body sensor networks: State-of-the-art and research challenges," *Information Fusion*, vol. 35, pp. 68–80, 2017.

[35] A. Poulose and D. S. Han, "Hybrid indoor localization using imu sensors and smartphone camera," *Sensors*, vol. 19, no. 23, p. 5084, 2019.

[36] N. Yu, X. Zhan, S. Zhao, Y. Wu, and R. Feng, "A precise dead reckoning algorithm based on bluetooth and multiple sensors," *IEEE Internet of Things Journal*, vol. 5, no. 1, pp. 336–351, 2017.

[37] D. Feng, J. Peng, Y. Zhuang, C. Guo, T. Zhang, Y. Chu, X. Zhou, and X.-G. Xia, "An adaptive imu/uwb fusion method for nlos indoor positioning and navigation," *IEEE Internet of Things Journal*, 2023.

[38] J. L. Carrera, Z. Li, Z. Zhao, T. Braun, and A. Neto, "A real-time indoor tracking system in smartphones," in *Proceedings of the 19th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, 2016, pp. 292–301.

[39] H. Zou, Z. Chen, H. Jiang, L. Xie, and C. Spanos, "Accurate indoor localization and tracking using mobile phone inertial sensors, wifi and ibeacon," in *2017 IEEE International Symposium on Inertial Sensors and Systems (INERTIAL)*. IEEE, 2017, pp. 1–4.

[40] A. G. Abdellatif, A. A. Salama, H. S. Zied, A. A. Elmahallawy, and M. A. Shawky, "An improved indoor positioning based on crowd-sensing data fusion and particle filter," *Physical Communication*, vol. 61, p. 102225, 2023.

[41] M. Nurpeiissov, A. Kuzdeuov, A. Assylkhanov, Y. Khassanov, and H. A. Varol, "End-to-end sequential indoor localization using smartphone inertial sensors and wifi," in *2022 IEEE/SICE International Symposium on System Integration (SII)*. IEEE, 2022, pp. 566–571.

[42] Z. Xu, B. Huang, B. Jia, and G. Mao, "Enhancing wifi fingerprinting localization through a co-teaching approach using crowdsourced sequential rss and imu data," *IEEE Internet of Things Journal*, 2023.

[43] J. Torres-Sospedra, R. Montoliu, A. Martínez-Usó, J. P. Avariento, T. J. Arnau, M. Benedito-Bordonau, and J. Huerta, "Ujiindoorloc: A new multi-building and multi-floor database for wlan fingerprint-based indoor localization problems," in *2014 international conference on indoor positioning and indoor navigation (IPIN)*. IEEE, 2014, pp. 261–270.

[44] Localization via wi-fi fingerprinting. Accessed on January 18, 2024. [Online]. Available: https://github.com/ryanmclark/Localization_via_WiFi_Fingerprinting

[45] Google arcore. Accessed on January 18, 2024. [Online]. Available: https://developers.google.com/ar/

[46] Apple arkit. Accessed on January 18, 2024. [Online]. Available: https://developer.apple.com/arkit/

[47] Arengine. Accessed on January 18, 2024. [Online]. Available: https://developer.huawei.com/consumer/cn/hms/huawei-arengine/

[48] D. Singh and B. Singh, "Investigating the impact of data normalization on classification performance," *Applied Soft Computing*, vol. 97, p. 105524, 2020.

[49] G. Lai, W.-C. Chang, Y. Yang, and H. Liu, "Modeling long-and short-term temporal patterns with deep neural networks," in *The 41st international ACM SIGIR conference on research & development in information retrieval*, 2018, pp. 95–104.

[50] A. Hoseini, S. Hosseini-Zahraei, and A. Akbarzadeh, "Fuzzy-based gait events detection system during level-ground walking using wearable insole," in *2022 29th National and 7th International Iranian Conference on Biomedical Engineering (ICBME)*. IEEE, 2022, pp. 333–339.

[51] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.

[52] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of rgb-d slam systems," in *2012 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2012, pp. 573–580.