Incremental Strategy-based Residual Regression Networks for Node Localization in Wireless Sensor Networks

Dongyao Zou*, Guohao Sun, Zhigang Li, Guangyong Xi and Liping Wang
School of Computer and Communication Engineering, Zhengzhou University of Light Industry,
Zhengzhou 450000, China
[e-mail: zdy@zzuli.edu.cn]
*Corresponding author: Dongyao Zou

Received April 23, 2022; revised June 14, 2022; accepted July 2, 2022; published August 31, 2022

Abstract

The easy scalability and low cost of range-free localization algorithms have led to their wide attention and application in node localization of wireless sensor networks. However, the existing range-free localization algorithms still have problems, such as large cumulative errors and poor localization performance. To solve these problems, an incremental strategy-based residual regression network is proposed for node localization in wireless sensor networks. The algorithm predicts the coordinates of the nodes to be solved by building a deep learning model and fine-tunes the prediction results by regression based on the intersection of the communication range between the predicted and real coordinates and the loss function, which improves the localization performance of the algorithm. Moreover, a correction scheme is proposed to correct the augmented data in the incremental strategy, which reduces the cumulative error generated during the algorithm localization. The analysis through simulation experiments demonstrates that our proposed algorithm has strong robustness and has obvious advantages in localization performance compared with other algorithms.

Keywords: Wireless Sensor Networks (WSNs); Convolutional Neural Networks (CNN); Data Augmentation; Node Localization; Degree of intersection.

The research has been funded by Key scientific and technological projects in Henan Province, grant number 212102210407 and 192102210133.

http://doi.org/10.3837/tiis.2022.08.009 **ISSN : 1976-7277**

1. Introduction

Location information plays a pivotal part in sensor node applications. With the mature development of Artificial Intelligence (AI) and the popularity of the application of the Internet of Things, Wireless Sensor Networks (WSNs) as an essential way to sense external information have become a development trend of IoT and its applications [1]. WSNs as a distributed sensor network, by arranging it in a specific scenario, its sensor nodes can continuously perform a series of operations such as information collection and location monitoring, etc. [2,3]. At present, WSNs are widely used in various fields: industrial control, smart homes, military security, environmental monitoring, fine agriculture, etc. In these applications, sensor nodes must specify their location to detail “what is happening where” and thus achieve subsequent information processing [4,5].

However, in practical applications, sensor nodes are affected by their factors and environmental factors in the localization process, triggering problems such as limited network connectivity between nodes and hindered localization. Therefore, an efficient and energy-saving localization algorithm has gradually become the focus of research. There are two types of localization algorithms commonly used in WSNs: range-based and range-free [6]. Range-based node localization algorithms include angle of arrival (AOA) [7], time of arrival (TOA) [8], time difference of arrival (TDOA) [9], and methods based on received signal strength indication (RSSI) [10], etc. However, the range-based node localization algorithms require high hardware requirements and are susceptible to interference from external environmental factors in practical applications. So, range-free algorithms are now getting more attention. Range-free algorithms have the benefits of simple implementation and low hardware requirements, but these classical algorithms have the problem of high localization errors in practical applications [11]. This has been extensively researched and improved by scholars both nationally and internationally.

Currently, the most used range-free algorithms are still improved by using intelligent optimization algorithms based on the DV-Hop algorithm [12]. Reference [13] proposed a localization algorithm using time difference of arrival and frequency difference measurements of arrival. The introduction of free gradient effectively solves the problem that the Cuckoo search algorithm converges slowly and tends to fall into local optimality and further improves the localization accuracy. Reference [14] improved the DV-Hop by introducing the particle swarm optimization (PSO) to convert the node localization problem into a particle search problem and analyzed the localization performance of the optimized algorithm under different network expansion structures. Reference [15] proposed a WND-DV-Hop algorithm by constructing a weighting factor based on the beacon node hop counts, and the least squares method was used to correct hop counts between nodes. Reference [16] proposed a method to correct the average hops error in the DV-Hop using RSSI. The main idea is to optimize the algorithm by establishing the channel fading model of RSSI signal and polynomial approximation estimation, and then finally reduce the error by recursive calculation. But this algorithm is highly susceptible to environmental factors in practical scenarios. Reference [17] presents an improved regularized least-squares DV-HOP algorithm by combining double least squares with a statistical filtering optimization strategy, which effectively reduces the errors generated in the algorithm localization process. Reference [18] proposed a new range-free iterative localization algorithm. The non-convex problem in node localization is transformed into a convex optimization problem by matrix transformation and first-order Taylor expansion. Finally, the iterative successive approximation is used to improve the final localization accuracy of the algorithm and effectively solve the node localization problem in complex scenes.

Recently, many scholars have also started to use mobile nodes to solve the node localization problem. Reference [19] proposed a localization algorithm based on mobile anchor nodes (ANs) that allows them to fly in a C-shaped path in the 3D network. The coordinates of the unknown nodes (UNs) are calculated by obtaining the RSSI values between the mobile anchor node (AN) and the unknown node (UN) to establish the distance matrix between them. It can still maintain good localization performance in the presence of multipath fading of the signal. Reference [20] proposed a project to solve the localization of target nodes in the region by moving ANs. The localization accuracy is improved by selecting suitable ANs and allowing them to move irregularly in the region. Finally, the data are particle filtered for distributed localization optimal estimation. Reference [21] proposed a mobile node localization algorithm based on 3DWSN. By improving the Savarese algorithm, the 3D location is effectively acquired in the local localization phase. In addition, the algorithm eliminates the node singularity problem and improves the localization accuracy.

However, most of these algorithms first obtain the estimated distance between nodes by some methods and then perform node finding by artificial intelligence algorithms. There is a large cumulative error during the operation of the algorithms, which affects the final positioning accuracy. In addition, the localization algorithms using mobile nodes, although improved in localization accuracy, are not applicable to all application scenarios for scalability and cost reasons. Accordingly, we try to find a new effective localization scheme to solve these problems. Inspired by convolutional neural networks (CNN) in image applications [22], we apply them to WSNs to solve the node localization problem. CNN have excellent feature extraction ability and adaptability, can make a prediction of the coordinates of UNs based on the information of known nodes, which is very suitable for the node localization problem. So in this paper, we consider the node localization problem as a kind of regression problem with a feature dataset to predict the node location.

In summary, in this paper, we propose an incremental strategy-based residual regression network [23] for node localization in wireless sensor networks, which effectively reduces the cumulative error generated during the algorithm and improves the localization accuracy. The main contributions are:

- CNN is applied to the node localization problem of WSNs and designed a node localization prediction model based on residual regression. The regression fine-tuning of the prediction results based on the intersection of the predicted coordinates of the AN and the communication range of the real coordinates in the model, together with the loss function, greatly improves the positioning accuracy of the algorithm.
- An incremental strategy is proposed to augment the training data set, which better meets the training data set requirements of the model and guarantees the localization prediction accuracy of the model.
- A correction scheme is proposed. The correction coefficients are constructed to correct the augmented data based on the data information between nodes in a small area. This scheme reduces the cumulative error generated during the operation of the algorithm and better reflects the network characteristics of the node distribution area.

2. Related Work

We can commonly understand the node localization problem in WSNs as follows: Randomly deploy sensor nodes in an area, and the nodes can communicate with each other through neighboring nodes within their communication range only a few nodes (ANs) have known location information now, and an algorithm model needs to be built to get the coordinates of the remaining sensor nodes (UNs) in the region according to the information of the known ANs. Therefore, how well the algorithm model is built can directly affect the final localization results. The widely used range-free localization schemes are either improved by using the Distance vector routing algorithm (DV) [24] or by estimating the location coordinates of UNs by building RSSI fading models, but all these algorithms inevitably have some problems due to their characteristics.

The DV-Hop is used to measure the distance between sensor nodes based on the DV routing protocol. Its main idea is to consider each node as a router, and pass information (information including the best output path and distance-vector, etc.) between neighboring router nodes. Every once in a while, the router will send its information table to all neighboring nodes to reach each target node, thus ensuring that all router nodes save the best path to any node. Then, calculate the coordinates of the UN. However, by locating based on the data information between routing nodes, there will inevitably be large locating errors. Therefore, many scholars have also improved on this basis to achieve satisfactory results.

To solve the problems of large cumulative errors and poor localization performance of range-free algorithms, we also try to find an efficient and low-cost solution to be applied to node localization in other fields. The development of AI and big data [25] has also focused more attention on deep learning related fields. Convolutional neural networks [26], with their excellent feature extraction capability and adaptability, have also started to be gradually applied to the node localization problem in WSNs. Unlike other localization algorithms, deep learning treats the node localization of sensors as a regression problem. The global UNs are predicted by feature extraction of data information of known nodes for localization. Reference [27] proposed a deep neural network-based sensor localization scheme. The optimal relationship existing between RSSI, and deployment nodes are predicted by constructing a deep neural network model. And for the sensor data loss problem, a method is proposed to reconstruct the lost data to achieve accurate localization. Reference [28] addresses the situation of node localization bias that exists in anisotropic networks. A distance estimation correction scheme for range-free algorithms constructed using neural networks is proposed to optimize the signal attenuation situation existing in anisotropic networks and further improve the localization performance in complex scenes is further improved. Reference [29] proposed a prediction algorithm using the stacked autoencoder model, and the model is trained with data between nodes. The superiority of this method is verified through simulation and analysis of different types of networks. Reference [30] established a deep learning model of denoising Auto-Encoder and trained the model by extracting fingerprint information from the RSSI to build a fingerprint database in 3D space. The experimental analysis demonstrated the good localization performance of the algorithm in both horizontal and vertical directions.

So in this paper, we propose a new algorithm that applies CNN to node localization. In this algorithm, we let the established model learn the relationship between the distance and the number of hops between nodes, and the prediction results are fine-tuned based on the intersection of the predicted coordinates of the AN with the communication range of the real coordinates, together with the loss function to improve the localization accuracy of the algorithm. In addition, we also propose an incremental strategy to augment the training dataset required by the model, which maximizes the training dataset requirement of the model. To ensure the accuracy of the augmented data, we propose a correction scheme to correct it, which reduces the cumulative error brought by it and thus improves the localization prediction performance of the model.

3. Incremental strategy-based residual regression networks

In summary, we propose an incremental strategy-based residual regression network for node localization in WSNs in this paper. This section describes this algorithm process in detail, which is divided into three stages: incremental acquisition of data information between nodes (training data set), training of the residual regression network model, and node localization. The global procedure of the algorithm is shown in **Fig. 1**.

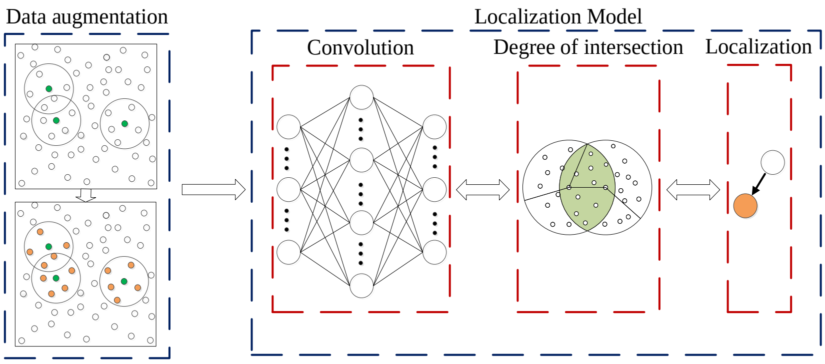

Fig. 1. Global procedure of the algorithm

3.1 Incremental strategy

Expanding the dataset required for model training is the most important purpose of the incremental strategy. The basic idea is to solve for the UN within their communication range using the information of the ANs according to the distribution state of the nodes in WSNs, to acquire the data between the nodes within the communication range. The advantage of this is that according to the local network characteristics in the area where the ANs are located, the cumulative error generated in the incremental process can be minimized, and the cumulative error generated in the positioning process of the network model designed in this paper can be reduced.

Assume that the amount of n nodes are stochastic distribution in a region denoted by set $\mathcal{Z}$; where the amount of ANs is m, denoted by set $\mathcal{A}$; the coordinates of ANs a_s, a_g are $(x_s, y_s), (x_g, y_g)$; the amount of UNs is $n-m$, denoted by set $\mathcal{U}$; the coordinates of UN u_k are (x_k, y_k); then for the sensor nodes in the region there are the following relations:

$$\begin{gathered}\mathcal{A}=\{a_1,\cdots,a_s,\cdots,a_g,\cdots,a_m\};\mathcal{U}=\{u_{m+1},\cdots,u_k,\cdots,u_n\}\\ \mathcal{Z}=\{\mathcal{A},\mathcal{U}\}=\{a_1,\cdots,a_s,\cdots,a_g,\cdots,a_m,u_{m+1},\cdots,u_k,\cdots,u_n\}\end{gathered} \tag{1}$$

After the deployment of sensor nodes in the region, the ANs firstly release their data message (including ANs identification, ANs location coordinates, and hop count) to the UNs in their communication range. **Table 1** represents the structural composition of the data message. If UNs receive the data message for the first time, it records this data message and adds 1 to the hop count (the initial hop count of the node is 0), and then continues to forward this data message to its neighboring nodes. If the UNs are not receiving the data message for the first time, it compares the previously saved data message with the minimum number of hops recorded in the currently received data message, selects the smaller hop count information to be saved, and continues to forward this data information to its neighboring nodes. According to the above steps, the ANs can obtain the distance data from other ANs by calculation, send this data message to the UNs in the region again in the form of broadcast, and get the data message between nodes in the whole region.

Table 1. Structural composition of the data message

Nodes identification	Average hop distance	Location coordinates	Minimum number of hops (initialized to 0)
ID	$Hopsize$	(x, y)	hop

The incremental strategy proposed is to get the data message between nodes in a small range. We need to find the UNs (data augmentation nodes) with the hop count of one in the saved data message and save this part of UNs into the set $\mathcal{Q}_T$. The discriminatory conditions are as follows:

$$\mathcal{Q}_T = \left\{ u_{m+1 \le k \le n} \wedge \left(\min\left(hop \right) = 1 \right) \right\}, s = 1, \cdots, m \tag{2}$$

The pseudo-code for saving the data augmentation nodes in the set $\mathcal{U}$ to the set $\mathcal{Q}_T$ is shown in **Algorithm 1**. $\mathcal{U}.(j)$ represents the element j of the set $\mathcal{U}$; $\mathcal{U}.hop(j)$ denotes obtaining the hop count information saved by the element j in the set $\mathcal{U}$.

Algorithm 1 Identify data augmentation nodes

Input: The set $\mathcal{U}$ of unknown nodes, number of elements $n - m$

Output: The set $\mathcal{Q}_T$ of data augmentation nodes

```
Begin
for j = 0; j ≤ n − m; j + + do
  if U.hop(j) == 1 do
      Q_T = U.(j)
end;
```

As shown in **Fig. 2**, the dataset expansion is divided into three key components: first, the UNs within their communication range are identified and selected according to the location of ANs; after that, correction coefficients are introduced to correct hop counts between nodes; finally, the data information between nodes is obtained by solving.

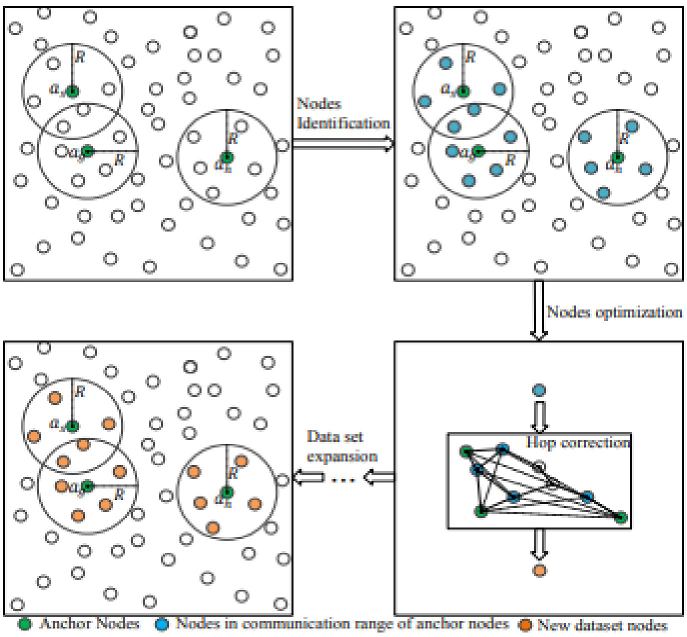

Fig. 2. Data augmentation process

3.1.1 Information acquisition of data augmentation nodes

In summary, we have identified the data augmentation nodes and stored them in the set $\mathcal{Q}_T$. We assume that u_k is the data augmentation node element in the set $\mathcal{Q}_T$. The set of ANs is known to be $\mathcal{A} = \{a_1, \cdots, a_s, \cdots, a_g, \cdots, a_m\}$, which contains ANs a_s, a_g with coordinates $(x_s, y_s), (x_g, y_g)$, and actual hop count between a_s and a_g is denoted by hop_{sg}. However, solving the information of data augmentation nodes directly by distance vector routing algorithm will produce a large error, so here we propose a correction scheme for the minimum number of hops among nodes in a small range. The ratio of the true distance d_{sg} among ANs with the communication radius R is defined as the ideal minimum hop count IH_{sg}. Moreover, the nodes solved in a small area are used as the model training data set, which well reflects the network characteristics of the area they are in and reduces the cumulative error in the model training process.

$$d_{sg} = \sqrt{\left(x_s - x_g\right)^2 + \left(y_s - y_g\right)^2} \tag{3}$$

$$IH_{sg} = \frac{d_{sg}}{R} \tag{4}$$

Based on the actual and ideal hop counts hop_{sg}, IH_{sg} between ANs, their relative errors are defined as the hop deviation factor φ_{sg}.

$$\varphi_{sg} = \frac{hop_{sg} - IH_{sg}}{hop_{sg}} \tag{5}$$

Based on the hop count deviation factor φ_{sg}, the hop correction factor can be defined as β_{sg}. (As shown in **Fig. 3**). Through extensive experimental analysis, it is shown that the correction effect is best when $n = 2$.

$$\beta_{sg} = 1 - \varphi_{sg}{}^{n} \tag{6}$$

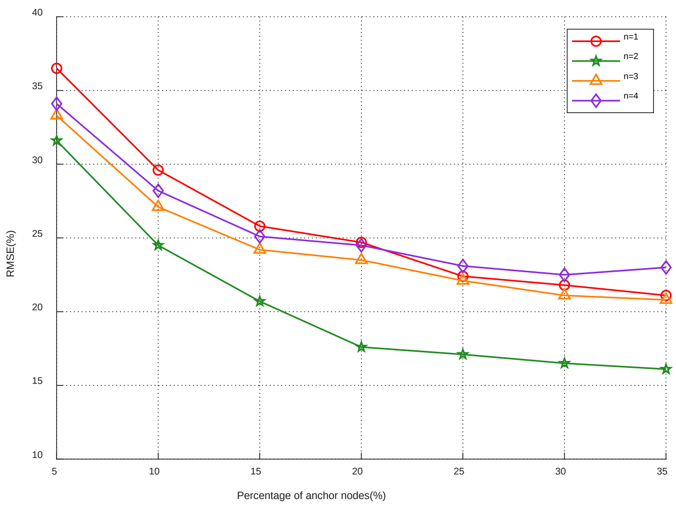

Fig. 3. Errors with different correction factors

We classify the hop count correction between nodes into three cases:

1) The hop count correction between ANs a_s, a_g. As shown in (7), the corrected hop count is defined as CH_{sg} based directly on the correction factor β_{sg}.

$$CH_{sg} = \beta_{sg} \times hop_{sg} \tag{7}$$

2) Hop count correction between the UN u_k and its nearest AN a_s. We use the average value of the correction coefficient between ANs to correct the hop count between the UN and its neighboring ANs. The average calculation can take full advantage of the hop count of the AN to better indicate the node distribution properties of the region where the UN is located, and the minimum hop count after correction is defined as CH_{ks}.

$$CH_{ks} = \frac{\sum_{s \neq g}^{m} \beta_{sg}}{m-1} \times hop_{ks} \tag{8}$$

3) Hop count correction between UN u_k and other AN a_g. In most cases, the paths between nodes in a small area overlap, so the correction coefficient between ANs is used directly to reflect the degree of deviation of the hop count between nodes better. As shown in (9), the corrected hop count is defined as CH_{kg} based on the calculation of the correction coefficient β_{sg}.

$$CH_{kg} = \beta_{sg} \times hop_{kg} \tag{9}$$

The average hop distance $AHopsize_I$ between ANs in the region can be calculated by using (10).

$$AHopsize_I = \frac{\sum_{s \neq g}^{m} \sqrt{d_{sg}}}{\sum_{s \neq g}^{m} CH_{sg}} = \sum_{s \neq g}^{m} \frac{\sqrt{d_{sg}}}{CH_{sg}} \tag{10}$$

The augmented node u_k acquires the estimated distance Ed_{ks}, Ed_{kg} from the ANs a_s, a_g based on its stored data information using (11) and (12).

$$Ed_{ks} = AHopsize_I \times CH_{ks} \tag{11}$$

$$Ed_{kg} = AHopsize_I \times CH_{kg} \tag{12}$$

3.1.2 Data Collection

Through the above steps, after completing the solution of the augmented nodes, the corrected hop count information between these data augmented nodes and ANs need to be saved into the vector as the training data for the model designed in this paper. For example, assume that $u_k, k = m+1, \cdots, m+j$ is an element in the set $\mathcal{Q}_{\boldsymbol{T}}$ of data augmentation nodes; then the hop count vector of u_k can be denoted as $\boldsymbol{CH}_k = \{CH_{k1}, \cdots, CH_{ks}, \cdots, CH_{km}\}$, where CH_{ks} is defined as the hop count between the data augmentation node u_k and the AN a_s obtained after correction, and the vector of hops between the AN a_s and other ANs can be denoted as $\boldsymbol{CH}_s = \{CH_{s1}, \cdots, CH_{sg}, \cdots, CH_{sm}\}$, and CH_{sg} is defined as the hop count between the AN a_s and a_g obtained after correction, and it should be noted that the AN with its own hop count $CH_{ss} = 0$. In addition, the distance vector between them is $\boldsymbol{Ed}_k = \{Ed_{k1}, \cdots, Ed_{ks}, \cdots, Ed_{km}\}$. The actual distance vector between the ANs is $\boldsymbol{d}_s = \{d_{s1}, \cdots, d_{sg}, \cdots, d_{sm}\}$. It should be noted that $d_{ss} = 0$.

The data information $\boldsymbol{CH}$ and $\boldsymbol{Ed}, \boldsymbol{d}$ of all nodes described above will be used as the initial data set (training and validation set) for training the model designed in this paper.

3.2 Residual regression prediction model for node localization in WSNs

3.2.1 Process of node localization prediction model

According to the calculation of the above formula, a prediction model of accurate localization of sensor network nodes based on residual regression is designed in this paper.

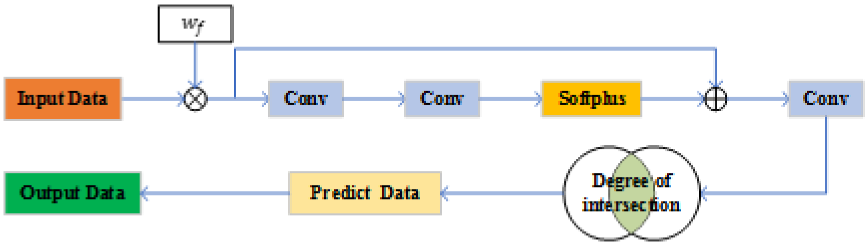

Fig. 4. Process of node localization prediction model

In this model, we first designed a weighting factor w_f to weight the AN samples and augmented node samples in the initial data set according to the node identifiers in the data samples, and the definition of w_f is shown in (13):

$$\omega_f = \begin{cases} \dfrac{2j}{m+j}, a_s \in \mathcal{A} \\ \dfrac{m}{m+j}, u_k \in \mathcal{Q}_T \end{cases} \tag{13}$$

Where a_s is the AN; u_k is the augmentation node; $\mathcal{A}$ denotes the data set of initial ANs and m denotes the amount of initial ANs; $\mathcal{Q}_T$ is the set of augmentation nodes and j denotes the amount of augmentation nodes.

After weighting the dataset, we used two convolution operators [31] to perform feature extraction on the input dataset to obtain the distribution characteristics of the space in which the nodes are located through feature acquisition; According to the regression properties of the localization model, we set the activation function Softplus [32], Softplus can better improve the nonlinear characteristics of the model and further enhance the robustness of the model compared with other activation functions; And the initial data set is fused with the extracted feature information by adding jump connections [33] to prevent problems such as data information loss during the convolution process; after that, the feature extraction is performed again by convolution on the output of the previous stage to further learn the mapping relationship between hop count and distance; and a concept based on the intersection of the predicted coordinate communication range of the AN with the actual coordinate communication range is proposed as residual compensation, which compensates for the information loss existing in the training process and further improves the localization prediction capability of the model.

3.2.2 Training

Pre-training and fine-tuning are the two main stages of the training process of the node localization prediction model for WSNs.

The pre-training stages perform feature extraction and optimization of the model parameters according to the information of the input data. Taking the hop counts as an example, suppose there exists a data set $\boldsymbol{CH}_u = \{CH_{k1}, \cdots, CH_{ks}, \cdots, CH_{km}\}$ with data volume m. The convolutional layer, for input $CH_{ks} \in \boldsymbol{CH}_u$, the feature extraction process can be expressed as:

$$\boldsymbol{Y}_{ks} = f\left(\boldsymbol{w}_c \cdot w_f CH_{ks} + \boldsymbol{b}_c\right) \tag{14}$$

$$f\left(CH_{ks}\right) = softplus\left(CH_{ks}\right) = \log\left(1 + e^{CH_{ks}}\right) \tag{15}$$

where $\boldsymbol{w_c}$ denotes the weight coefficient matrix in the feature extraction process, $\boldsymbol{b_c}$ denotes the bias vector, $\boldsymbol{Y_{ks}}$ denotes the new vector obtained after convolution on CH_{ks}, and $f(\cdot)$ denotes the activation function in the model.

In addition, we also assign a loss function $L\left(\boldsymbol{CH}_u, \boldsymbol{C\hat{H}}_u\right)$ in the model to measure the strengths and weaknesses of the localization prediction model to ensure that the parameters $\boldsymbol{\alpha}$ in the model are optimal.

$$\boldsymbol{\alpha} = \min L\left(CH_{ks}, C\hat{H}_{ks}\right) = \frac{1}{2}\sum_{s=1}^{m}\left\|CH_{ks} - C\hat{H}_{ks}\right\|^2 \tag{16}$$

where $\boldsymbol{C\hat{H}_u}$ denotes the output of the convolution and $\|\cdots\|$ denotes the 2-Norm.

During the training process, we represent the hop count vector and distance vector in the training dataset as matrices, defined as the AN dataset $\mathbf{CH}_{\mathrm{a}}, \mathbf{d}_{\mathrm{a}}$ and the augmented dataset $\mathbf{CH}_{\mathrm{u}}, \mathbf{Ed}_{\mathrm{u}}$, respectively, which are represented by matrices as follows:

$$\mathrm{CH_a} = \left[CH_1; CH_2; \cdots; CH_m\right] = \begin{bmatrix} CH_{11} & CH_{12} & \cdots & CH_{1m} \\ CH_{21} & CH_{22} & \cdots & CH_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ CH_{m1} & CH_{m2} & \cdots & CH_{mm} \end{bmatrix} \tag{17}$$

$$\mathrm{CH_u} = \left[CH_{m+1}; CH_{m+2}; \cdots; CH_{m+j}\right] = \begin{bmatrix} CH_{m+1,1} & CH_{m+1,2} & \cdots & CH_{m+1,m} \\ CH_{m+2,1} & CH_{m+2,2} & \cdots & CH_{m+2,m} \\ \vdots & \vdots & \ddots & \vdots \\ CH_{m+j,1} & CH_{m+j,2} & \cdots & CH_{m+j,m} \end{bmatrix} \tag{18}$$

$$\mathrm{Ed_u} = \left[Ed_{m+1}; Ed_{m+2}; \cdots; Ed_{m+j}\right] = \begin{bmatrix} Ed_{m+1,1} & Ed_{m+1,2} & \cdots & Ed_{m+1,m} \\ Ed_{m+2,1} & Ed_{m+2,2} & \cdots & Ed_{m+2,m} \\ \vdots & \vdots & \ddots & \vdots \\ Ed_{m+j,1} & Ed_{m+j,2} & \cdots & Ed_{m+j,m} \end{bmatrix} \tag{19}$$

$$\mathrm{d_a} = \left[d_1; d_2; \cdots; d_m\right] = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1m} \\ d_{21} & d_{22} & \cdots & d_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ d_{m1} & d_{m2} & \cdots & d_{mm} \end{bmatrix} \tag{20}$$

We define the parameters in the localization prediction model designed in this paper as $\boldsymbol{\rho} = \{\boldsymbol{w}_1, \boldsymbol{b}_1, \boldsymbol{w}_2, \boldsymbol{b}_2, \boldsymbol{w}_3, \boldsymbol{b}_3, \boldsymbol{w}_4\}$. $\boldsymbol{w_c}$ and $\boldsymbol{b_c}$ denote the weight coefficient matrix and deviation vector in the convolution of the Cth layer. The model is pre-training by a multi-layer logistic regression supervised learning, as well as fine-tuning the model parameters by back-propagating the intersection of the predicted coordinates of the ANs with the actual coordinates and the loss function as residuals [29]. In the pre-training process, the training data $\mathbf{CH}_{\mathrm{a}}, \mathbf{CH}_{\mathrm{u}}$ are firstly used as the initial input of the localization model, which is trained using (16) and updated with $\boldsymbol{w}_1$ and $\boldsymbol{b}_1$. After that, Layer-by-layer update of weights. In the fine-tuning phase, we propose a concept of intersection degree. As shown in **Fig. 5**, when the communication range overlap between the predicted and actual coordinates of the AN is higher, the model's positioning bias will be small. So in this model, we define the intersection degree $\boldsymbol{w}_4$ as the ratio of the communication area between the predicted coordinates $\hat{a}(\hat{x}, \hat{y})$ and the actual coordinates $a(x, y)$ of ANs, and the error between the predicted and true coordinates is smaller when $\boldsymbol{w}_4$ is closer to one.

$$\boldsymbol{w}_4 = \frac{S_{a\hat{a}}}{\pi R^2} \approx \frac{d_{a\hat{a}}}{R} = \frac{\sqrt{\left(x-\hat{x}\right)^2 + \left(y-\hat{y}\right)^2}}{R} \tag{21}$$

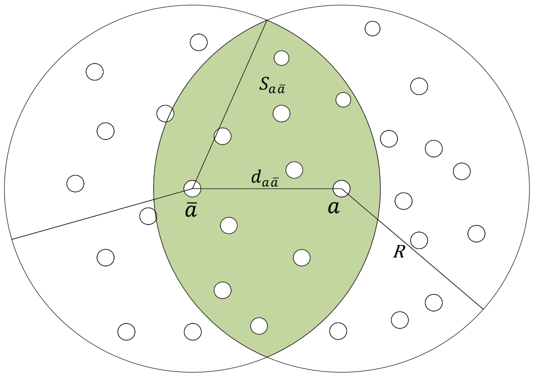

Fig. 5. Communication range intersection between actual and estimated coordinates

In the fine-tuning phase, we use $\mathbf{CH}_\mathrm{a}, \mathbf{d}_\mathrm{a}, \mathbf{CH}_\mathrm{u}, \mathbf{Ed}_\mathrm{u}$ as training data to optimize the parameters of the localization model.

$$\begin{aligned}\boldsymbol{\rho} &= \min\left(\frac{L\left(\boldsymbol{d}_a, \hat{\boldsymbol{d}}_a\right) + L\left(\boldsymbol{Ed}_u, \boldsymbol{E}\hat{\boldsymbol{d}}_u\right)}{w_4}\right) = \min\left(\frac{\sum_{s=1}^{m}\left\|\boldsymbol{d}_s - \hat{\boldsymbol{d}}_s\right\|^2 + \sum_{k=m+1}^{m+j}\left\|\boldsymbol{Ed}_k - \boldsymbol{E}\hat{\boldsymbol{d}}_k\right\|^2}{2\cdot\boldsymbol{w}_4}\right) \\ &= \min\left(\frac{\sum_{s=1}^{m}\left\|\boldsymbol{d}_s - V\left(\boldsymbol{CH}_s\right)\right\|^2 + \sum_{k=m+1}^{m+j}\left\|\boldsymbol{Ed}_k - V\left(\boldsymbol{CH}_k\right)\right\|^2}{2\cdot\boldsymbol{w}_4}\right)\end{aligned} \tag{22}$$

where $V(\cdot)$ denotes the transformation function of the data in the model from the hop count to the distance, $\widehat{\boldsymbol{d}}_s$ is the distance vector of AN a_s after model prediction, and $E\hat{d}_k$ is the distance vector of node u_k in the augmented dataset after model prediction.

3.2.3 Node Localization

After the training of the localization model is finished, all the UNs $u_l, l \in \{m+1, m+2, \cdots, n\}$ in the region use the method in Section 3.1 to obtain the corrected hop count vector $\boldsymbol{CH}_l, \boldsymbol{CH}_l = \{CH_{l1}, CH_{l2}, \cdots, CH_{lm}\}$ between itself and each AN, which will be input to the localization model. The detailed process is as follows:

$$\boldsymbol{E}\hat{\boldsymbol{d}}_l = V\left(\boldsymbol{CH}_l\right) = f\left(\boldsymbol{w}_3\left(\left(\boldsymbol{w}_2\left(\boldsymbol{w}_1\left(w_f\boldsymbol{CH}_l\right) + \boldsymbol{b}_1\right) + \boldsymbol{b}_2\right) + w_f CH_l\right) + \boldsymbol{b}_3\right) = \left\{E\hat{d}_{l1}, \cdots, E\hat{d}_{ls}, \cdots E\hat{d}_{lm}\right\} \tag{23}$$

Where $f(\cdot)$ denotes the function in (14), $E\hat{d}_{ls}$ is the distance between the UN u_l and the AN a_s estimated by the model.

If the predicted coordinate of the UN u_l is $(\hat{x}_l, \hat{y}_l)$, the following set of distance equations can be constructed according to the coordinates of the AN a_s.

$$\begin{cases} (x_1-\hat{x}_l)^2+(y_1-\hat{y}_l)^2=E\hat{d}_{l1}^2 \\ \vdots \\ (x_s-\hat{x}_l)^2+(y_s-\hat{y}_l)^2=E\hat{d}_{ls}^2 \\ \vdots \\ (x_m-\hat{x}_l)^2+(y_m-\hat{y}_l)^2=E\hat{d}_{lm}^2 \end{cases} \tag{24}$$

Equation (24) is a nonlinear system of equations, which can be transformed into a linearized equation by subtracting the third equation from the first two equations:

$$EX=D \tag{25}$$

$$E=2\times\begin{bmatrix} (x_1-x_m) & (y_1-y_m) \\ \vdots & \vdots \\ (x_s-x_m) & (y_s-y_m) \\ \vdots & \vdots \\ (x_{m-1}-x_m) & (y_{m-1}-y_m) \end{bmatrix} \tag{26}$$

$$D=\begin{bmatrix} x_1^2-x_m^2+y_1^2-y_m^2+E\hat{d}_{lm}^2-E\hat{d}_{l1}^2 \\ \vdots \\ x_s^2-x_m^2+y_s^2-y_m^2+E\hat{d}_{lm}^2-E\hat{d}_{ls}^2 \\ \vdots \\ x_{m-1}^2-x_m^2+y_{m-1}^2-y_m^2+E\hat{d}_{lm}^2-E\hat{d}_{l(m-1)}^2 \end{bmatrix} \tag{27}$$

Solving (14) by the least squares method [34,35], the predicted coordinates can be obtained as $X=(\hat{x}_l,\hat{y}_l)$.

$$X=\left(E^TE\right)^{-1}E^TD=\begin{bmatrix}\hat{x}_l \\ \hat{y}_l\end{bmatrix} \tag{28}$$

After obtaining the predicted coordinates based on the distance predicted by the localization model, we define the error present in (28) as:

$$\delta_l=\frac{\sum_{s=1}^{m}\left(\sqrt{(\hat{x}_l-x_s)^2+(\hat{y}_l-y_s)^2}-E\hat{d}_{ls}\right)^2}{m} \tag{29}$$

The global error after localization prediction of all UNs in the region is defined as:

$$\delta=\frac{\sum_{l=m+1}^{n}\sum_{s=1}^{m}\left(\sqrt{(\hat{x}_l-x_s)^2+(\hat{y}_l-y_s)^2}-E\hat{d}_{ls}\right)^2}{n\cdot m-m^2} \tag{30}$$

4. Incremental strategy-based residual regression networks

In the simulation experiment part, we simulated a $100m\times100m$ area with 100 randomly deployed wireless sensor nodes.

The performance metrics of the algorithm are evaluated by changing the AN percentage and the sensor nodes' communication radius and comparing the DV-Hop [12], PSODV-Hop [14], WND-DV-Hop [15], and RANN [28] with the algorithm in this paper. Considering the interference factors in the real application scenario, we considered the Degree of Radio Irregularity (DOI) in our simulation and set it to 0.05 according to reference [36]. The schematic diagram of node distribution shown in **Fig. 6** and **Table 2** shows the detailed parameter settings for the simulation section. All data are averaged after 100 simulations.

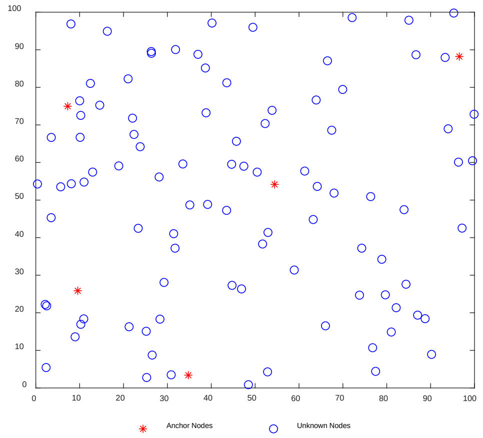

Fig. 6. Schematic diagram of node distribution

Table 2. The simulation experiment parameters

Parameter	Value
The area size	$100m \times 100m$
Number of sensor nodes, incremental step	$60 - 160{,}20$
Anchor node percentage, incremental step	$5\% - 35\%, 5\%$
Node communication radius, incremental step	$15 - 40m, 5m$
Number of simulation experiments	100
DOI	0.05

To verify the localization prediction performance of the model, we introduce the root mean square error and define the localization error equation as follows:

$$\text{RMSE} = \frac{1}{K}\sum\frac{\sum_{l=m+1}^{n}\sqrt{(\hat{x}_l - x_l)^2 + (\hat{y}_l - y_l)^2}}{(n-m)\cdot R} \tag{31}$$

where K denotes the amount of simulations, $(\hat{x}_l, \hat{y}_l)$ and (x_l, y_l) are the predicted and actual coordinates of the UN u_l, the amount of sensor nodes is n, the amount of AN is m, and the communication radius of the nodes is R.

4.1 Error analysis of the incremental stage

The size of the error generated in the incremental phase can play a crucial role in the prediction accuracy of our model. Therefore, seeking the optimal parameter settings to minimize the error can lead to better results for our model.

In section 3.1, we augmented the data set required for model training by an incremental strategy and in the process, introduced a correction coefficient to correct the minimum number of hops between nodes. However, influenced by factors such as hop distance error and network connectivity, we still need to find the optimal parameter values by magnifying the proportion of ANs in the region and increasing the communication radius of sensor nodes. As shown in **Fig. 7** & **Fig. 8**, we compare the localization performance before and after the hop count correction with DV-Hop, and we can observe that the localization error of all algorithms decreases as the set parameter value increases, and the localization effect after the hop count correction is obviously better than the other two algorithms, which can prove that our optimization strategy in the data augmentation stage is very effective.

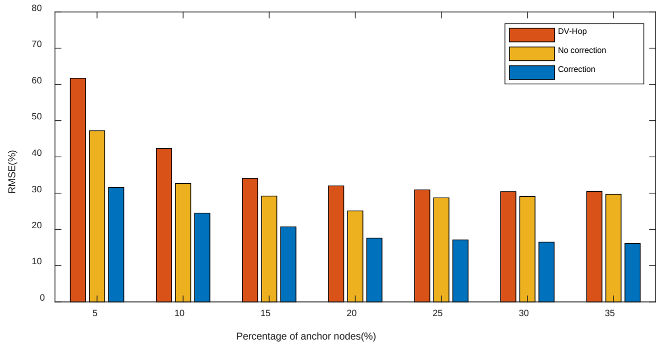

Fig. 7. Influence of anchor nodes percentage on incremental stage

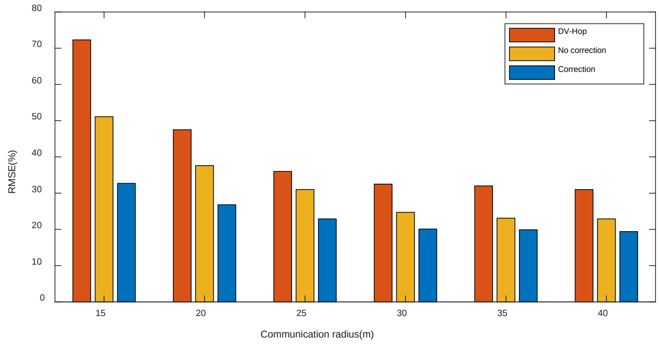

Fig. 8. Influence of nodes communication radius on incremental stage

As shown in **Fig. 7** & **Fig. 8**, the final positioning error can be significantly reduced by hop count correction. In **Fig. 7**, we set the communication radius of the AN to 30 m. By increasing the proportion of ANs (incremental step of 5 m), we can observe an overall decreasing trend of localization error for the three algorithms. However, when the percentage of ANs is greater than 25%, the positioning error before hop count correction starts to rise again. This is because an increase in the proportion of ANs means that the amount of amplified data is also increasing, and the cumulative error between nodes is also increasing, which results in the accuracy of the amplified data decreasing as the proportion of ANs increases. Moreover, an excessively high AN proportion does not improve the localization accuracy significantly. Also, it causes a great waste of resources, so selecting an appropriate AN proportion will also improve the localization performance of the algorithm to a certain extent. In **Fig. 8**, we set the percentage of ANs to 20%, increase the communication radius of the nodes (incremental step of $5m$), and decrease the localization error of the three algorithms. After the node communication radius reaches $30m$, the AN can achieve full coverage of the area, the network connectivity between the nodes in the area reaches the best state, and the decreasing trend of localization error tends to level off at this time. Therefore, considering when the percentage of ANs is 20% , and the communication radius is $30m$, the error of the augmented data is minimized, and the data optimization of the incremental phase is optimized.

4.2 Localization error analysis of the model

For range-free node localization algorithms, the ratio of ANs, the communication radius of the nodes, and the number of nodes in the region all impact the localization performance of the algorithm. We analyzed the effects of each of these parameters on the localization performance of the model in a variety of cases and compared it with several excellent localization algorithms.

4.2.1 Influences of anchor nodes percentage on model localization error

ANs play a crucial role in node localization. If the percentage of ANs is too low, the coverage of UNs in the region is small, and the final localization accuracy of the algorithm is poor. If the percentage of ANs is too high, the accuracy of algorithm localization will be relatively improved, but the cost of network and manual deployment will be higher. So, the optimal AN ratio can make sure the algorithm's accuracy and control the cost. To analyze the influence of AN proportion on the model localization error, we set the maximum communication radius R of the nodes to $30m$, the trend of AN percentage from 5% to 35%, and the incremental step is 5%, and 100 simulation experiments are conducted for each algorithm to take the average value.

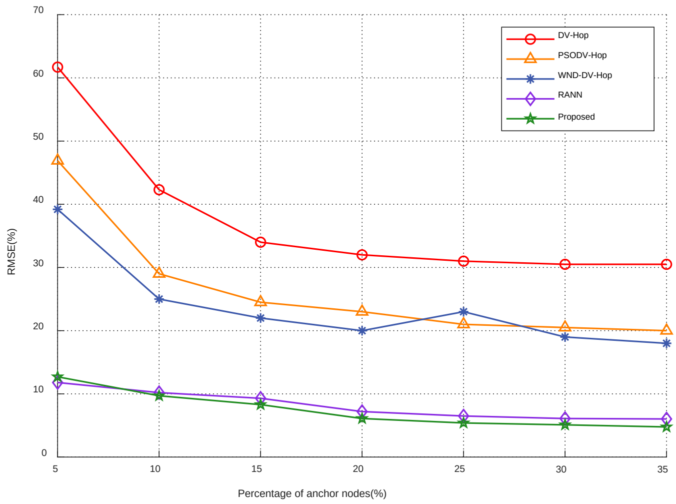

Fig. 9. Influences of anchor nodes percentage on model localization error

The algorithm in this paper is significantly better than other algorithms. The RANN algorithm is proposed as a de-correction mechanism by using artificial neural networks, but this method causes data loss in the convolution process and needs to take into account the signal fading in practical application scenarios. There are also higher-order operations in this algorithm, which will increase the computational load of the node and shorten the lifetime of the node. In addition, through **Fig. 9** we see that the first three algorithms are strongly influenced by the proportion of ANs. The algorithm localization error decreases sharply at the beginning of the increase of the ANs percentage, and when the AN percentage reaches 20%, the decreasing trend starts to level off, in which there is also a sudden change in localization error. Our algorithm model in this paper is trained and validated based on the data between some nodes through small data augmentation, so it is less influenced by the proportion of ANs, and the global localization error decreasing trend is more modest. By comparing and analyzing the simulation data under each parameter, it can be found that the proposed algorithm model in this paper outperforms several other algorithms in the field of overall localization performance, with a 18.9% improvement than PSODV-Hop.

Table 3 shows the positioning errors of the proposed algorithm model with different AN percentages. The data in the table are obtained by averaging the localization error through 100 simulation experiments with the communication radius set to $30m$.

Table 3. Localization errors at different anchor node percentages

Anchor nodes percentage (%)	5	10	15	20	25	30	35
Minimum error (m)	4.115	3.973	3.785	3.247	2.745	2.312	2.267
Maximum error (m)	4.733	4.516	4.465	3.957	3.267	3.118	2.959
Average error (m)	4.465	4.286	4.126	3.319	2.934	2.471	2.366

4.2.2 Influences of nodes communication radius on model localization error

The communication radius of a node affects the connectivity between nodes in a region. The larger the communication radius, the greater the node coverage, the greater the number of neighboring nodes, and the better the connectivity of the network. But an increase in the communication radius often means higher hardware requirements for the sensor nodes. We set the percentage of ANs to 20%, the trend of node communication radius from $15m$ to $40m$, the incremental step is $5m$, and 100 simulation experiments are conducted for each algorithm to take the average value, and analyze its effect on the algorithm localization error.

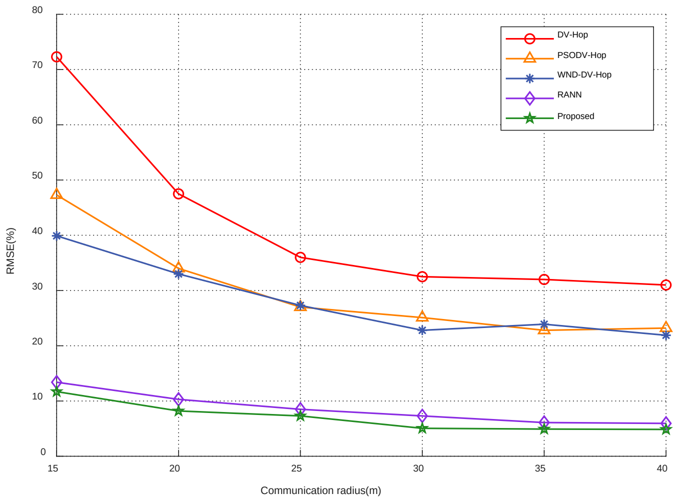

Fig. 10. Influences of nodes communication radius on model localization error

By increasing the communication radius of the sensor nodes, the positioning error of the algorithms all show a decreasing trend. The positioning effect of the model in this paper is obviously better than several other algorithms, which can keep below 10%. The first three algorithms show a sharp decrease in positioning error at the beginning of the radius increase, and the trend of positioning error tends to level off after the communication radius increases to $30m$. This is because these several positioning algorithms without ranging utilize distance vector routing algorithms for positioning. When the communication radius of the nodes increases to a certain level, the communication range of the ANs can basically achieve network communication coverage of the whole area, thus achieving better positioning results. Compared with these algorithms, RANN and the algorithmic model in this paper are less affected by the communication radius and have significantly welled localization effects. However, only the irregularity of the node distribution area is considered in the RANN algorithm, ignoring the data loss caused by back-propagation and communication success rate between nodes during model training and the gradient disappearance problem in the subsequent prediction process of node data by the artificial neural network. The training data of the proposed algorithm model are obtained by augmenting the network characteristics in a small area according to the incremental strategy. In **Fig. 10**, the localization error of the model remains basically unchanged after the communication radius reaches $30m$. Moreover, after augmenting the data by incremental strategy, the error of the proposed algorithm can be maintained within 10%, which improves the localization performance by 22.8% compared with PSODV-Hop. Therefore, the overall analysis shows that the algorithm model is significantly more advantageous in the evaluation of each parameter.

As shown in **Table 4**, the localization error of the model during different communication radius. The data in the table are obtained by averaging the positioning error through 100 simulation experiments when the percentage of ANs is set to 20%.

Table 4. Relationship between model localization error and node communication radius

Node communication radius (m)	15	20	25	30	35	40
Minimum error (m)	5.915	4.472	3.526	2.839	2.318	2.016
Maximum error (m)	6.833	5.235	4.386	3.673	3.951	4.017
Average error (m)	6.218	4.716	4.013	3.185	3.714	3.373

4.2.3 Influences of anchor nodes percentage on model localization error

The node distribution density is intimately related to the whole amount of nodes. When the whole amount of nodes is small, the distribution density of nodes in the region will be sparse, the connectivity between nodes will be reduced, and the localization accuracy of the algorithm will be poor. When the whole amount of nodes is large, the load of the network increases, which also affects the localization performance of the algorithm. So, to analyze the influence of the distribution density of sensor nodes in the region on the algorithm localization error. We set the proportion of ANs to 20%, the node communication radius to $30m$, the trend of the whole amount of sensor nodes in the region to $60-160$ (node distribution density of $0.01-0.016$), and the incremental step size to 20. Each algorithm is simulated 100 times to take its average value as the average localization error.

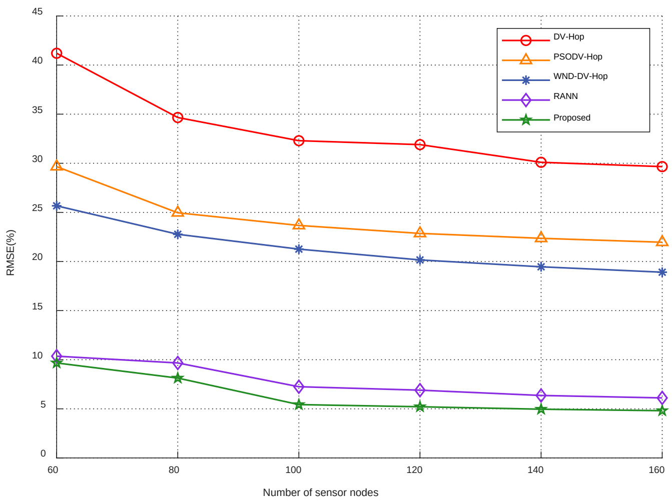

Fig. 11. Influence of the whole number of nodes on model localization error

When the node distribution density in the region is incremented from $0.01-0.016$, the localization effect of different algorithms are shown in **Fig. 11**. By increasing the amount of sensor nodes in the region, the localization errors of the exemplified algorithms are gradually reduced, and the algorithm is more excellent than several other algorithms in the case of sparse or dense node distribution, with a 17.8% improvement in localization performance compared to PSODV-Hop. This also shows that, by establishing a residual regression-based node localization model for WSNs, the node localization is treated as a regression problem solved with good localization accuracy. Unlike the intelligent optimization algorithm, this algorithmic model optimizes the model parameters through residual regression, which can effectively avoid the problem of falling into local optimal solutions. The analysis of the parameters through simulation experiments shows that, based on the parameter settings of the algorithm model in this paper, when the node distribution density is $0.01-0.012$, the localization accuracy of the model is more excellent than other related algorithms, and the best localization performance is achieved at this time.

4.3 Algorithm complexity analysis

The issue of energy consumption is also a significant element to be evaluated in the positioning of nodes in WSNs. The lower algorithm complexity reduces energy consumption, increases positioning speed, and extends the lifetime of the sensor nodes. According to the localization characteristics of our design algorithm model, the time complexity evaluation of the algorithm can be divided into data augmentation, model training, and localization.

Assuming that there are m ANs, $n-m$ UNs in the region, the augmented data nodes are k, the amount of iterations in model training is T, and there are X neurons in the hidden layer, the complexity of the data augmentation phase can be denoted as $O(mk)$, the complexity of model training as $O(TmkX)$, and the complexity of node localization as $O(m(n-m))$. According to the node distribution characteristics of WSNs, the amount of ANs is small, the augmented data nodes $k \ll n-m$ and the overall structure of the model, and the amount of iterations are independent of the nodes, the training and localization of the model can be completed by the aggregation nodes, and the nodes only need to complete the hop count and distance estimation between themselves and the ANs, and the complexity notes $O(m(n-m))$.

5. Conclusion

In this paper, to solve the problems of large cumulative errors and poor localization performance of range-free algorithms. We propose an incremental strategy-based residual regression network for sensor node localization, which treats node localization as a regression problem with a feature data set. The incremental strategy is used to establish the hop-count and distance relationships between nodes in a small range to train the model, and a correction scheme is used to reduce the cumulative error generated therein. During the training of the model, we fine-tune the prediction results according to the intersection of the predicted coordinates of the AN with the communication range of the real coordinates and the loss function. Simulation shows that the algorithm effectively reduces the cumulative errors generated during the operation and has high localization accuracy.

In future research, we will continue to focus on node localization optimization schemes in WSNs, with a bias toward solving sensor node localization problems in irregular regions.

References

[1] S. C. Mukhopadhyay, S. K. S. Tyagi, N. K. Suryadevara, V. Piuri, F. Scotti and S. Zeadally, "Artificial Intelligence-Based Sensors for Next Generation IoT Applications: A Review," *IEEE Sens. J.*, vol. 21, no. 22, pp. 24920-24932, Nov. 2021. Article (CrossRef Link)

[2] N. B. Gaikwad, V. Tiwari, A. Keskar and N. Shivaprakash, "Heterogeneous Sensor Data Analysis Using Efficient Adaptive Artificial Neural Network on FPGA Based Edge Gateway," *KSII Trans. Internet Inf. Syst.*, vol. 13, no. 10, pp. 4865-4885, 2019. Article (CrossRef Link)

[3] Akram J, Munawar HS, Kouzani AZ, Mahmud MAP, "Using Adaptive Sensors for Optimised Target Coverage in Wireless Sensor Networks," *Sensors*, vol. 22, no. 3, p. 1083, 2022. Article (CrossRef Link)
[4] M. Khan, B. N. Silva, C. Jung and K. Han, "A context-Aware Smart Home Control System based on ZigBee Sensor Network," *KSII Trans. Internet Inf. Syst.*, vol. 11, no. 2, pp. 1057-1069, 2017. Article (CrossRef Link)
[5] S. Vikrant, P. R. B, B. H. S and P. D, "Policy for planned placement of sensor nodes in large scale wireless sensor network," *KSII Trans. Internet Inf. Syst.*, vol. 10, no. 7, pp. 3213-3230, 2016. Article (CrossRef Link)
[6] V. C. S. R. Rayavarapu and A. Mahapatro, "A Novel Range-Free Anchor-Free Localization In WSN Using Sun Flower Optimization Algorithm," in *Proc. of 2021 Advanced Communication Technologies and Signal Processing (ACTS)*, pp. 1-6, 2021. Article (CrossRef Link)
[7] W. Ding, S. Chang and J. Li, "A Novel Weighted Localization Method in Wireless Sensor Networks Based on Hybrid RSS/AoA Measurements," *IEEE Access*, vol. 9, pp. 150677-150685, 2021. Article (CrossRef Link)
[8] Xiong, W., Schindelhauer, C., So, H.C. et al, "Maximum Correntropy Criterion for Robust TOA-Based Localization in NLOS Environments," *Circuits Syst. Signal Process.*, vol. 40, pp. 6325-6339, 2021. Article (CrossRef Link)
[9] X. Ma, T. Ballal, H. Chen, O. Aldayel and T. Y. Al-Naffouri, "A Maximum-Likelihood TDOA Localization Algorithm Using Difference-of-Convex Programming," *IEEE Signal Process Lett.*, vol. 28, pp. 309-313, 2021. Article (CrossRef Link)
[10] Balakrishnan, A., Ramana, K., Nanmaran, K. et al., "RSSI Based Localization and Tracking in a Spatial Network System using Wireless Sensor Networks," *Wireless Pers Commun.*, vol.123, pp. 879–915, 2022. Article (CrossRef Link)
[11] W. Wu, X. Wen, H. Xu, L. Yuan and Q. Meng, "Accurate Range-free Localization Based on Quantum Particle Swarm Optimization in Heterogeneous Wireless Sensor Networks," *KSII Trans. Internet Inf. Syst.*, vol. 12, no. 3, pp. 1083-1097, 2018. Article (CrossRef Link)
[12] Niculescu, D., Nath, B., "DV Based Positioning in Ad Hoc Networks," *Telecommunication Syst.*, vol. 22, pp. 267–280, 2003. Article (CrossRef Link)
[13] Abd El Aziz, M, "Source localization using TDOA and FDOA measurements based on modified cuckoo search algorithm," *Wireless Netw.*, vol. 23, no. 2, pp. 487-495, 2017. Article (CrossRef Link)
[14] A. Hadir, Y. Regragui and N. M. Garcia, "Accurate Range-Free Localization Algorithms Based on PSO for Wireless Sensor Networks," *IEEE Access*, vol. 9, pp. 149906-149924, 2021. Article (CrossRef Link)
[15] Fengrong Han, Izzeldin Ibrahim Mohamed Abdelaziz, Xinni Liu and Kamarul Hawari Ghazali, "An Enhanced Distance Vector-Hop Algorithm using New Weighted Location Method for Wireless Sensor Networks," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 11, no. 5, 2020. Article (CrossRef Link)
[16] Messous S, Liouane H, Cheikhrouhou O, Hamam H, "Improved Recursive DV-Hop Localization Algorithm with RSSI Measurement for Wireless Sensor Networks," *Sensors*, vol. 21, no. 12, p. 4152, 2021. Article (CrossRef Link)
[17] H. Liouane, S. Messous, O. Cheikhrouhou, M. Baz and H. Hamam, "Regularized Least Square Multi-Hops Localization Algorithm for Wireless Sensor Networks," *IEEE Access*, vol. 9, pp. 136406-136418, 2021. Article (CrossRef Link)
[18] Y. Jin, L. Zhou, L. Zhang, Z. Hu and J. Han, "A Novel Range-Free Node Localization Method for Wireless Sensor Networks," *IEEE Wireless Commun. Lett.*, vol. 11, no. 4, pp. 688-692, April. 2022. Article (CrossRef Link)
[19] Iram Javed, Xianlun Tang, Kamran Shaukat, Muhammed Umer Sarwar, Talha Mahboob Alam, Ibrahim A. Hameed, Muhammad Asim Saleem, "V2X-Based Mobile Localization in 3D Wireless Sensor Network," *Secur. Commun. Netw.*, vol. 2021, 2021. Article (CrossRef Link)

[20] Luo Q, Liu C, Yan X, Shao Y, Yang K, Wang C, Zhou Z, "A Distributed Localization Method for Wireless Sensor Networks Based on Anchor Node Optimal Selection and Particle Filter," *Sensors*, vol. 22, no. 3, p. 1003, 2022. Article (CrossRef Link)
[21] Iram Javed, Xianlun Tang, Muhammad Asim Saleem, Muhammad Umer Sarwar, Maham Tariq, Casper Shikali Shivachi, "3D Localization for Mobile Node in Wireless Sensor Network," *Wirel Commun Mob Comput.*, vol. 2022, 2022. Article (CrossRef Link)
[22] Y. -B. Liu, M. Zeng and Q. -H. Meng, "Unstructured Road Vanishing Point Detection Using Convolutional Neural Networks and Heatmap Regression," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1-8, 2021. Article (CrossRef Link)
[23] Quan, Y., Li, Z., Chen, S. et al., "Joint deep separable convolution network and border regression reinforcement for object detection," *Neural Comput & Applic.*, vol. 33, pp. 4299–4314, 2021. Article (CrossRef Link)
[24] J. Vivarekar, S. T. Sonnis and D. A. Roy, "Time-Triggered Distance Vector Routing Protocol For Mobile Ad-hoc Networks," in *Proc. of 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pp. 01-07, 2021. Article (CrossRef Link)
[25] Tien, J.M, "Internet of Things, Real-Time Decision Making, and Artificial Intelligence," *Ann. Data. Sci.*, vol. 4, no. 2, pp. 149-178, 2017. Article (CrossRef Link)
[26] M. A. M. Sadr, J. Gante, B. Champagne, G. Falcao and L. Sousa, "Uncertainty Estimation via Monte Carlo Dropout in CNN-Based mmWave MIMO Localization," *IEEE Signal Process Lett.*, vol. 29, pp. 269-273, 2022. Article (CrossRef Link)
[27] Zhishu Shen, Tiehua Zhang, Atsushi Tagami, Jiong Jin, "When RSSI encounters deep learning: An area localization scheme for pervasive sensing systems," *J. Netw. Comput. Appl.*, vol. 173, pp. 102852, 2021. Article (CrossRef Link)
[28] A. El Assaf, S. Zaidi, S. Affes and N. Kandil, "Robust ANNs-Based WSN Localization in the Presence of Anisotropic Signal Attenuation," *IEEE Wireless Commun. Lett.*, vol. 5, no. 5, pp. 504-507, Oct. 2016. Article (CrossRef Link)
[29] Z. Yan, X. Liu, W. Ji, Y. Liu, G. Han and Y. Xie, "Stacked Auto-Encoders Based Localization without Ranging over Internet of Things," *IEEE Internet Things J.*, vol. 9, no. 10, pp. 7826-7841, 2022. Article (CrossRef Link)
[30] C. Xiao, D. Yang, Z. Chen and G. Tan, "3-D BLE Indoor Localization Based on Denoising Autoencoder," *IEEE Access*, vol. 5, pp. 12751-12760, 2017. Article (CrossRef Link)
[31] C. Hu, X. Wu and Z. Shu, "Bagging deep convolutional autoencoders trained with a mixture of real data and GAN-generated data," *KSII Trans. Internet Inf. Syst.*, vol. 13, no. 11, pp. 5427-5445, 2019. Article (CrossRef Link)
[32] Zhao, H., Liu, F., Li, L. et al, "A novel softplus linear unit for deep convolutional neural networks," *Appl Intell*, vol. 48, pp. 1707-1720, 2018. Article (CrossRef Link)
[33] Oyebade K. Oyedotun, Kassem Al Ismaeil, Djamila Aouada, "Training very deep neural networks: Rethinking the role of skip connections," *Neurocomputing*, vol. 441, pp. 105-117, 2021. Article (CrossRef Link)
[34] Wang, Y, "Linear least squares localization in sensor networks," *J Wireless Com Network.*, vol. 51, 2015. Article (CrossRef Link)
[35] L. Zhang, T. Zhang and H. -S. Shin, "An Efficient Constrained Weighted Least Squares Method With Bias Reduction for TDOA-Based Localization," *IEEE Sens. J.*, vol. 21, no. 8, pp. 10122-10131, April. 2021. Article (CrossRef Link)
[36] Q. Xiao, B. Xiao, J. Cao and J. Wang, "Multihop Range-Free Localization in Anisotropic Wireless Sensor Networks: A Pattern-Driven Scheme," *IEEE Trans. Mobile Comput.*, vol. 9, no. 11, pp. 1592-1607, Nov. 2010. Article (CrossRef Link)

Dongyao Zou received the B.S. degree in industrial automation from Zhengzhou University, in 1996, and the M.S. degree in microelectronics and solid-state electronics and the Ph.D. degree in circuits and systems from the Beijing University of Posts and Telecommunications, in 2005 and 2008, respectively. Since July 2008, he has been engaged in teaching and research work with the School of Computer and Communication Engi neering, Zhengzhou University of Light Industry. His research interests include IoT information sensing, processing and node localization technologies.

Guohao Sun received the B.S. degree in Network Engineering from the School of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou, China, where he is currently pursuing the M.S. degree. His research interests include wireless communication technologies, information processing, and node localization in wireless sensor networks.

Zhigang Li received the Ph.D. degree from Beijing University of Posts and Telecommunications (BUPT) in 2016. He is currently a research staff of Zhengzhou University of Light Industry (ZZULI), Henan, China. His current research interests include mobile computing, cellular network and C-V2X.

Guangyong Xi received the B.S. degree in engineering surveying from Hohai University, in 2001, and M.S. degree and the PhD degree in engineering surveying from Hohai University, in 2004 and 2011, respectively. He is an associate professor in the School of Computer and Communication Engineering at Zhengzhou University of Light Industry. His research focuses on IoT information sensening, processing and precise indoors wireless positioning.

Liping Wang received the Ph.D. degree in electronic technology from Huazhong University of Science and Technology, Wuhan, Hubei, China, in 2016, M.S. degree in electrical engineering from Hunan University, Changsha, Hunan, China, in 2006, and B.S. degree in electronic technology from Xinyang Normal University, Xinyang, Henan. She has been engaged in teaching and research work with the School of Computer and Communication Engineering, Zhengzhou University of Light Industry since 2017. Her research interests include embedded systems and wireless sensor network.