Research on *Indoor Patrol Robot Location based on BP Neural Network*

Liguo Dong 1*, Junhao Lv1

1South China Institute Of Software Engineering, Guangzhou University, Guangzhou, Guangdong, 510990, China

* E-mail: 1062491936@qq.com

**Abstract.** With the development of science and technology, patrol robots play a more and more important role in indoor monitoring. It is the location of the abnormal situation that the most important task of the patrol robot is to display accurately, so studying its location is of vital significance. In the case of known indoor maps, it is indispensable for patrol robots to select the odometer which achieves accurate positioning. The traditional odometer is mainly realized by means of displacement, laser signal or multi-sensor fusion. However, the positioning often fails due to problems such as mechanical structure, error accumulation, power consumption and heating. In addition, the phenomenon that peripherals occupy limited robot resources is also prominent. To this end, this paper proposes to use BP neural network instead of sensors to obtain the odometer. This method is that the laser data, constructing the map of the robot, and the BP neural network determined are combined to achieve precise positioning and reduce development costs. Experiments have proved that this method may be used to improve the endurance of the robot battery in addition to accurate positioning.

1. **Introduction**

Mobile robots are an important field in robot research. It is of vital practical value to develop mobile robots meeting different needs. Patrol robots research is one of the research branches on mobile robots. Though great achievements have been made in patrol robot research, many problems have appeared in the practical use. For example, in the case of known indoor maps, once the displacement odometer is used on patrol robots, the positioning failure occurs when the tracks slip or problems arise in mechanical structure. If the laser odometer is used, pose estimation failure will occur if the radiation distance of the laser radar is not great enough or the error accumulation is too big. So far, acquiring information through combination of the laser and the IMU is the acquisition mode of odometer with the optimal positional accuracy. However, the power consumption of the IMU will affect the endurance of the robot battery to some extent, which also merits attention in mobile robot research and development.

With the development of machine learning, it has become a reality to apply neural network to patrol robots. To sum up the above comparison, the paper proposes: on the basis of the known indoor maps, first, use emulation technique to obtain the data set of laser positioning and actual position; second, do model training of BP neural network; finally, with the confirmed information on BP neural network and the laser, odometer information is released, hence achieving precise localization.
2. Location method based on BP neural network

With the popularity of neural network research, many kinds of neural network have been proposed. As a matter of fact, the most widely used is the multi-layer perception based on classic BP algorithm (error back propagation algorithm). BP (Back Propagation) algorithm is featured by complete mathematical foundation and easy portability. As a classical algorithm, BP algorithm, compared with other neural network algorithms, is much simpler and consumes less computer memory. Therefore, it is more suitable for robotic equipment.

2.1. BP neural network

![BP neural network structure](image)

Figure 1. The three-layer perceptron structure

Among the applications of multi-layer perceptron, the three-layer perceptron structure shown in Figure 1 is the most common. The following describes the calculation formula for the weight adjustment of the three-layer BP algorithm. It is agreed in advance that in all derivation processes, the output layer will have \( j=0,1,2,...,m \), \( k=1,2,...,l \); the hidden layer will have \( i=0,1,2,...,n \), \( j=1,2,...,m \).

For the output layer, it can be written as

\[
\Delta w_{jk} = -\eta \frac{\partial E}{\partial w_{jk}} = -\eta \frac{\partial E}{\partial \text{net}_k} \frac{\partial \text{net}_k}{\partial w_{jk}}
\]  

(1)

For the hidden layer, it can be written as

\[
\Delta v_{ij} = -\eta \frac{\partial E}{\partial v_{ij}} = -\eta \frac{\partial E}{\partial \text{net}_j} \frac{\partial \text{net}_j}{\partial v_{ij}}
\]  

(2)

An error signal for the output layer or the hidden layer is define, let

\[
\delta_k = -\frac{\partial E}{\partial \text{net}_k}
\]  

(3)

\[
\delta_j = -\frac{\partial E}{\partial \text{net}_j}
\]  

(4)

For the output layer, it can be also written as

\[
\Delta w_{jk} = \eta \delta_k y_j
\]  

(5)

For the hidden layer, it can be also written as

\[
\Delta w_{ij} = \eta \delta_j y_i
\]  

(6)

It can be seen that the derivation will be completed if the error signal \( \delta_k \) and weight adjustment \( \delta_j \), in Equation (5) and Equation (6), are gained. Therefore, how to calculate \( \delta_k \) and \( \delta_j \) is the following step.

For the output layer, \( \delta_k \) can be expanded to
\[ \delta_k^o = - \frac{\partial E}{\partial \text{net}_k} = - \frac{\partial E}{\partial o_k} \frac{\partial o_k}{\partial \text{net}_k} = - \frac{\partial E}{\partial o_k} f'(\text{net}_k) \]  

(7)

For the hidden layer, \( \delta_j^y \) can be expanded to

\[ \delta_j^y = - \frac{\partial E}{\partial \text{net}_j} = - \frac{\partial E}{\partial y_j} \frac{\partial y_j}{\partial \text{net}_j} = - \frac{\partial E}{\partial y_j} f'(\text{net}_j) \]  

(8)

For the output layer, we can get

\[ \frac{\partial E}{\partial o_k} = -(d_k - o_k) \]  

(9)

For the hidden layer, we can get

\[ \frac{\partial E}{\partial y_j} = - \sum_{k=1}^{l} (d_k - o_k)^2 f'(\text{net}_k)w_{jk} \]  

(10)

Then

\[ \delta_k^o = (d_k - o_k)o_k(1-o_k) \]  

(11)

\[ \delta_j^y = \left[ \sum_{k=1}^{l} (d_k - o_k)^2 f'(\text{net}_k)w_{jk} \right] f'(\text{net}_j) = \left( \sum_{k=1}^{l} \delta_k^o w_{jk} \right)y_j(1-y_j) \]  

(12)

Finally, the formula, the weight adjustment, which is from the BP learning algorithm of the three-layer perceptron, is:

\[ \begin{align*}
\Delta w_{jk} &= \eta \delta_j^y y_j - \eta (d_k - o_k) o_k (1 - o_k) y_j \\
\Delta v_{ij} &= \eta \delta_j^y x_i - \eta (\sum_{k=1}^{l} \delta_k^o w_{jk}) y_j (1 - y_j) x_i
\end{align*} \]  

(13)

2.2. Acquisition of training data

In the course of the research, 4*5*0.8m indoor model is built and Gazebo is used to draw a simulation map. In addition, a plug-in for path drawing is added in Rviz. A path is drawn in the map. The path is located as close as possible to the middle of the map to ensure that the robot can reach all positions safely. As shown in Figure 2, the white is safe areas, the gray risk areas, and the black danger areas.

After compilation of the script file, the robot is made to create random points along the drawn path as navigation target points. Navigation package of DWA in ROS is adopted to enable the robot to autonomously navigate to these target points. Among them, global cost maps and local cost maps are used for the robot to avoid obstacles. In the simulation process, information about noise-containing laser and the robot pose is acquired.
The above-mentioned information about the noise-containing radar is not the original laser data, so coordinate transformation needs to be changed, as shown in Figure 3 and Equation 14.

\[
\begin{align*}
  x &= r \cdot \cos(\omega) \cdot \sin(\alpha) \\
  y &= r \cdot \cos(\omega) \cdot \cos(\alpha) \\
  z &= r \cdot \sin(\omega)
\end{align*}
\]  

(14)

In Equation 14, \( r \) is the linear distance between radar and the obstacle observed by the laser radar, \( \omega \) is the vertical angle, and \( \alpha \) is the rotation angle value. In two-dimensional space, \( \omega = 0 \). The value of \( r \) can be acquired through subscribing to laser data, mileage data and then coordinate transformation.

In the end, both 362*1 laser data set and 362*2 pose data set in robot two-dimensional space are acquired.

2.3. Model determining

Due to the limited memory of the robot equipment, the BP network model should be kept as simple as possible under the premise of ensuring model accuracy. In the neural network, the 2 hidden layers with appropriate activation functions can represent any decision boundary of arbitrary precision, and can fit any smooth mapping of any precision. At the same time, it is necessary to learn a complex description with one layer, so the number of hidden layers in the network is 3.
The range of hidden layer neurons is determined according to the empirical formula on StackOverFlow,

\[ N_h = \frac{N_i}{(\alpha \times (N_j + N_o))} \]  

In Equation 15, \( N_h \) is the number of neurons in the input layer; \( N_j \) is the number of neurons in the output layer; \( N_o \) is the number of samples in the training set; \( \alpha \) is an arbitrary and variable value, usually the range can be 2-10.

Then, MATLAB is used to build a neural network model, the training set randomly selects 350 samples from the data set, and the rest are used as test set samples.

![Figure 4. Model training process](image1)

![Figure 5. Model error](image2)

After simulation, when the model error is less than 0.05, the research team believes the model is successfully trained, and finally the hidden layer size is obtained, as shown in Figures 4 and 5.

In order to improve the calculation speed of the model and be more suitable for robotic devices, ReLU(x), which is much faster than the sigmoid function and the tanh function, is adopted as the activation function, and MSE as the loss function, as shown in Equation 16 and Equation 17.

\[ \text{ReLU}(x) = \max(x,0) \]  

\[ \text{MSE} = \frac{1}{N} \sum_{x \in \mathcal{X}} (x_p(x) - x_g(x))^2 \]  

3. Experiment and analysis

In the built indoor model, the hexapod patrol robot is used to observe the base_link in Rviz for the realization of robot position measurement, as shown in Figure 6.
Position at the start taken as the original point, according to the walking conditions shown in Figure 7 and Figure 8, coordinate values are observed respectively after the robot returns to the original point with three different odometers. The positioning accuracy is reflected by the deviation displacement. As shown in Table 1 and Table 2.

Table 1. Statistics of positioning accuracy under the first walking condition

| Odometer Type | Starting Coordinates (m) | Returning Coordinates (m) | Displacement Deviation (cm) |
|---------------|--------------------------|---------------------------|----------------------------|
| Lidar         | (0.0016,0.0035)          | (-0.011,0.018)            | 1.92                       |
| Lidar and IMU | (0.0068,0.0032)          | (-0.0086,-0.0055)         | 1.77                       |
| Ours          | (0.0057,0.0036)          | (-0.0091,0.0062)          | 1.78                       |

Table 2. Statistics of positioning accuracy under the second walking condition

| Odometer Type | Starting Coordinates (m) | Returning Coordinates (m) | Displacement Deviation (cm) |
|---------------|--------------------------|---------------------------|----------------------------|
| Lidar         | (0.0050,0.0039)          | (0.097,0.088)             | 12.5                       |
| Lidar and IMU | (-0.046,-0.0090)        | (-0.045,-0.041)           | 3.2                        |
| IMU           | (0.0049,0.0031)          | (-0.024,-0.0039)          | 3.0                        |

Table 1 and Table 2 compared, it can be seen that under the simple walking condition, the accuracy of the three is relatively high. However, as the walking distance becomes larger and the movement...
becomes complicated, large accumulated error appears in the laser radar odometer while the other two are not large. Under the second walking condition, the accuracy is even higher than that of the IMU if our method is adopted. This indicates the reasonable training of the BP neural network model and its effective integration with the information from laser radar.

The patrol robot's endurance of battery has been a great concern in the design. For this reason, the hexapod robot, with the full battery and multi-thread, is made to move back and forth to compare the impacts of different odometers on battery endurance of robots, as shown in Table 3.

| Method      | Battery Capacity (mAh) | Starting Time (s) | End Time (s) | Battery Life (s) |
|-------------|------------------------|-------------------|--------------|------------------|
| Lida rand IMU | 8400           | 0                 | 1758         | 1758             |
| Ours        | 8400           | 0                 | 1926         | 1962             |

Through experiments, it is found that the robot with IMU equipment has a shorter battery life. This means that the IMU consumes more battery power, which leads to the deterioration of the robot's endurance, especially in the more threads, hence an outstanding problem. In contrast, under the same condition, power endurance may be greatly improved if our method is used.

4. Conclusion

In the case of known indoor maps, we propose the location method of indoor patrol robots based on BP neural network. It turns out to an efficient and reliable method. With the laser information of the robot in the map construction and the BP neural network determined, precise location is realized. So far, a lot of experiments on robotic equipment have been done by the research team. This paper comes to the conclusion that compared with the existing odometers, if the method proposed in this paper is used and meets the accuracy requirements, the robot will have greater endurance, hence reducing the cost of engineering development.

Fund project

Foundation for Characteristic Innovation Projects in Higher Education of Guangdong, China (No. KTSCX 245); Foundation for Young Innovative Talents in Higher Education of Guangdong, China (No.2017KQNCX272, No.2018KQNCX389, 2018KQNCX391); Foundation for Construction of Intelligent Scientific Research and Application Team (No. ST201903).

References

[1] Li L., Wu H., Chen Y. (2008) Research based on BP neural network algorithm and derivation. China Water Transport. 8(1): 161-162.
[2] Han, L. (2006) Artificial Neural Network Tutorial. Beijing University of Posts and Telecommunications Press, Beijing.
[3] Montemerlo, M. (2002). A factored solution to the simultaneous localization and mapping problem. Aaii, Edmonton.
[4] Durrant-Whyte, H., & Bailey, T. (2006). Simultaneous localization and mapping: part i. IEEE Robotics & Automation Magazine, 13(2), 99-110.
[5] Dissanayake G, Huang S, Wang Z, et al. (2011) A review of recent developments in Simultaneous Localization and Mapping. In: CIIEEE International Conference on Industrial and Information Systems. Kandy. 477-482.
[6] Dissanayake, M.W.M., Gamini, Newman, Paul, & Clark, et al. (2001). A solution to the simultaneous localization and map building (slam) problem. IEEE Transactions on Robotics & Automation, 17(3): 229-241.
[7] Guivant, J. E., & Nebot, E. M. (2001). Optimization of the simultaneous localization and map-building algorithm for real-time implementation. IEEE Transactions on Robotics and Automation, 17(3), 242-257.
[8] José A. Castellanos, José Neira, & Juan D. Tardós. (2001). Multisensor fusion for simultaneous localization and map building. IEEE Transactions on Robotics, 17, 908-914.

[9] Cadena, C., Carlone, L., Carrillo, H., Latif, Y., & Leonard, J. J. (2016). Simultaneous localization and mapping: present, future, and the robust-perception age. IEEE Transactions on Robotics, 32(6).

[10] LeCun, Y. Bengio, Y. & Hinton, G. (2015) Deep learning. Nature, 444-436.

[11] Cybenko, G. (1989) Approximations by superpositions of sigmoidal functions, Mathematics of Control, Signals, and Systems, 2 (4): 303-314.

[12] Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. Neural Networks, 4(2), 251-257.

[13] Hinton, G. E., Osindero, S. & Teh, Y. W. . (2006). A fast learning algorithm for deep belief nets. Neural Computation, 18(7), 1527-1554.