

1-8-2019

An Improved Algorithm of VIRE Based on BP Neural Network

Hongshan Kong

Information Engineering University, Zhengzhou 450001, China;

Bin Yu

Information Engineering University, Zhengzhou 450001, China;

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the Artificial Intelligence and Robotics Commons, Computer Engineering Commons, Numerical Analysis and Scientific Computing Commons, Operations Research, Systems Engineering and Industrial Engineering Commons, and the Systems Science Commons

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation.

An Improved Algorithm of VIRE Based on BP Neural Network

Abstract

Abstract: Aimed at the problem that the proper threshold affects the accuracy of positioning in VIRE algorithm, a RFID positioning method combining BP neural networks with VIRE algorithm is presented and is named BP-VIRE. BP-VIRE utilizes the same reader's configuration and deployment as that of VIRE, and regards referenced tag RSSI and coordinate values as training samples, and constructs the positioning model based on neural networks with the help of BP algorithm which uses the method of error back propagation. According to the simulation results, compared with VIRE, BP-VIRE performance is better in the capability of positioning, esp. in the aspects of average deviation, maximum deviation, and boundary area deviation, so the improved method is more suitable for indoor positioning application in comparison with VIRE.

Keywords

indoor positioning, BP neural networks, virtual reference, radio frequency identification, received signal strength indicator

Recommended Citation

Kong Hongshan, Yu Bin. An Improved Algorithm of VIRE Based on BP Neural Network[J]. Journal of System Simulation, 2018, 30(9): 3586-3595.

一种基于BP神经网络的VIRE改进算法研究

孔红山, 郁滨

(信息工程大学, 郑州 450001)

摘要: 针对VIRE算法中阈值选择影响定位精度的问题, 提出一种BP神经网络与VIRE相结合的射频识别定位方法(简称BP-VIRE)。BP-VIRE采用与VIRE算法相同的阅读器配置与部署, 以参考标签的RSSI值和坐标值作为训练样本, 采用误差反传训练的BP算法构建基于神经网络的定位模型。仿真结果表明, 与VIRE相比, 本算法在平均误差、最大误差、边界区域误差等定位性能方面得到较大提升, 更适用于室内定位系统中的应用。

关键词: 室内定位; BP神经网络; VIRE; 射频识别; RSSI

中图分类号: TP391.9 文献标识码: A 文章编号: 1004-731X(2018)09-3586-10

DOI: 10.16182/j.issn1004731x.joss.201809046

An Improved Algorithm of VIRE Based on BP Neural Network

Kong Hongshan, Yu Bin

(Information Engineering University, Zhengzhou 450001, China)

Abstract: Aimed at the problem that the proper threshold affects the accuracy of positioning in VIRE algorithm, a RFID positioning method combining BP neural networks with VIRE algorithm is presented and is named BP-VIRE. BP-VIRE utilizes the same reader's configuration and deployment as that of VIRE, and regards referenced tag RSSI and coordinate values as training samples, and constructs the positioning model based on neural networks with the help of BP algorithm which uses the method of error back propagation. According to the simulation results, compared with VIRE, BP-VIRE performance is better in the capability of positioning, esp. in the aspects of average deviation, maximum deviation, and boundary area deviation, so the improved method is more suitable for indoor positioning application in comparison with VIRE.

Keywords: indoor positioning; BP neural networks; virtual reference; radio frequency identification; received signal strength indicator

引言

随着无线移动通信技术的发展及物联网技术在各个行业的应用, 室内定位备受人们关注。由于

室内环境复杂、多径传播和噪声干扰较大等问题, 如何实现精度高、速度快、开销少的定位系统显得尤为重要。当前, 主要有红外线、超声波、WLAN、ZigBee和射频识别(Radio frequency identification, 简称RFID)等室内定位技术等。红外线是较早用于室内定位的技术, 典型红外线定位系统有Active Badge^[1], 由于红外信号要求直线视距传输且易受外界光线干扰, 导致定位精确度有限。典型的超声波定位系统有Cricket^[2]和Active Bat^[3], 超声波定



收稿日期: 2017-01-09 修回日期: 2017-04-06;
作者简介: 孔红山(1981-), 男, 河南濮阳, 博士生, 副教授, 研究方向为室内定位; 郁滨(1964-), 男, 河南郑州, 博士, 教授, 博导, 研究方向为信息安全、无线网络安全技术、视觉密码等。

<http://www.china-simulation.com>

• 3586 •

位精确度高、抗干扰能力强,但构建超声波定位系统需要部署大量的基础设施,系统成本较高。典型的 WLAN 定位系统有 RADAR^[4],WLAN 定位可以利用现成的无线网络,易于构建,所需基站数目较少,但定位精确度不够理想,且对于体积小、能耗受限的设备是很不方便。

RFID 技术作为一种快速发展的技术,凭借其无接触通信、高数据速率和安全性、非视线识别、低成本等优点,成为室内定位的一个主要研究方向^[5-7]。由华盛顿大学的 Hightower 等提出的 SpotON^[8]是最早将 RFID 技术应用于室内定位的系统。该系统依据经验数据建立接收信号强度 RSSI (Received Signal Strength Indicator)与距离的关系,基于对 RSSI 的分析并结合聚合算法实现三维定位,定位精度也不够理想。密歇根州立大学的 LANDMARC^[9]引入了参考标签,通过计算待定位标签与参考标签 RSSI 之间的欧几里得距离,利用 k-近邻法选取 k 个最近邻参考标签实现定位,具有成本较低、容易实现和环境适应性较强特点,但易受多径效应影响,定位精度不够高。

VIRE 算法^[10]是由香港科技大学赵弋洋等人基于 LANDMARC 算法改进的定位方法,该算法引入虚拟参考标签以提高参考标签密度,采用小概率位置排除法去除冗余位置信息,利用加权 k-近邻法实现更准确的定位。小概率位置排除法中的阈值对定位精度的影响至关重要,若阈值太大,过多噪声虚拟标签会被选;若阈值太小,可能性大的虚拟标签会被忽略,这都会严重影响定位精度^[11]。文献[12]提出一种自适应的 VIRE 算法(AVIRE),可自适应选取合适的阈值;文献[13]采用非线性插值法计算虚拟标签的 RSSI 值,并用自动获取的动态阈值代替 VIRE 中的固定阈值。

神经网络有着很强的非线性信息处理能力,文献[14-16]尝试将神经网络引入室内定位系统中,主要研究信号强度 RSSI 与距离的非线性关系。VIRE 算法中阈值的选取是一个难点,且直接影响定位精度,本文研究如何利用神经网络

对 VIRE 算法进行改进,把 VIRE 算法阈值选择难题转化为应用人工神经网络进行室内定位,提出一种基于 BP 神经网络的 VIRE 改进算法(简称 BP-VIRE),解决 VIRE 算法中阈值选取造成的定位精度问题。

1 VIRE 算法原理及分析

1.1 基本原理

VIRE 系统定位模型如图 1 所示。VIRE 算法是在不改变 LANDMARC 算法中标签和阅读器位置的情况下,在真实参考标签间插入“虚拟参考标签”来增加已知的位置参考信息,在不改变 LANDMARC 算法中标签和阅读器位置的情况下,不增加任何参考标签,将每由 4 个参考标签覆盖的网格分为 $N \times N$ 个大小相同的虚拟网格单元,虚拟标签的值可以通过插值算法计算得来。

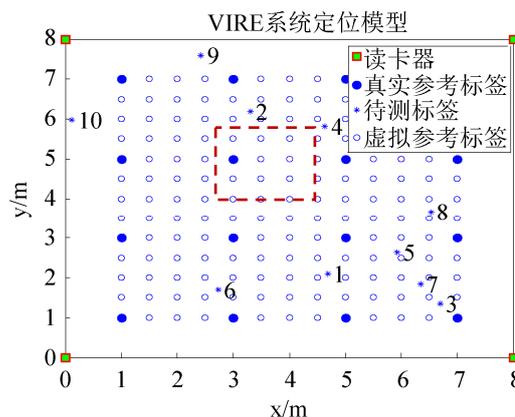


图 1 VIRE 系统定位模型
Fig. 1 VIRE system positioning model

当虚拟参考网格建立后,通过阅读器可读出每个标签的 RSSI 值,每个 RSSI 值都对应一个位置域。VIRE 算法引入“模糊地图”的概念,把定位区域分成很多小区域,每个小区域的中心对应一个参考标签。计算待定位标签与参考标签 RSSI 绝对差值,设定一个阈值,通过比较阈值和绝对差值建立获取待定位标签的最近邻参考标签。

VIRE 在 LANDMARC 的基础上对权值进行了改进,引入了两个权值 w_{1i} 和 w_{2i} , $w_i = w_{1i} \times w_{2i}$ 。

权值 w_{1i} 代表所选中的参考标签和待定位标签之间的 RSSI 差值, 差值越小, 代表区域越接近, 相应的权值越大; 权值 w_{2i} 代表挑选出的虚拟参考标签密度, 是一个与被选择的虚拟参考区域的密度相关的函数, 密度越大的区域所对应的 w_{2i} 越大, 表示待定位标签在该区域的可能性越大。待定位标签的坐标可由下式计算:

$$(x, y) = \sum_{i=1}^n w_i * (x_i, y_i) \quad (1)$$

1.2 算法分析

由 VIRE 的基本原理可知, 阈值是 VIRE 算法中最重要的参数之一, 直接关系着定位的精度。当阈值过大, 那么被挑选出的参考标签就越多, 很多无用的虚拟参考标签将对定位产生很大的干扰, 影响定位精度; 若阈值太小, 则会有很多携带有用信息的参考标签将被忽略, 将严重影响定位精度。另一方面, 在 VIRE 的权值计算方法中, 权值的准确性完全依赖所选参考标签, 参考标签又与阈值密切相关, 因此, 阈值又会造成权值的计算误差进而影响定位精度。最后, 合适阈值的选取是随着不同待定位标签区域、不同算法参数而变化的, 采取固定阈值也会影响定位精度。所以, VIRE 算法中阈值的选取是一个难点, 且直接影响定位精度, 归根结底是由于 VIRE 算法无法很好的解决有用定位数据的获取问题, 要么遗漏有用数据, 要么增加干扰数据, 并且通过权值进一步加大定位误差。

2 BP-VIRE 算法

BP 神经网络是人工神经网络中一种典型的信息修正模型。针对 VIRE 算法中阈值的选择问题, 引入 BP 神经网络, 在保留 VIRE 算法中参考标签、虚拟参考标签和阅读器位置信息的情况下, 以参考标签从读写器接收到的信号强度为输入、以参考标签位置坐标为输出, 通过误差反传训练的学习算法, 不断优化信号强度与坐标之间的非线性映射关系, 建立 BP 神经网络定位模型。利用所建模型, 通过待测标签的信号强度估计其坐标, 达到定位目的。

2.1 网络结构的确定

BP 神经网络模型拓扑结构包括输入层、隐含层和输出层。输入层的输入为 RFID 读写器在参考标签处的信号强度值, BP-VIRE 算法采用与 VIRE 算法相同的阅读器配置与部署, 即 4 个读写器部署在待定位区域的四个端点, 因此, 输入层神经元数目为 4。输出层为该参考标签的坐标, 包含 x 和 y 坐标值, 输出层神经元数目为 2。BP 神经网络至少有 1 个隐含层, 而 2 个隐含层的神经网络可以满足任何的分类要求, 当采用 1 个隐含层, 神经元个数利用经验公式 $\sqrt{n+m+\alpha}$ (n 为输入层神经元个数, m 为输出层神经元个数, α 为 [1,10] 之间的常数) 确定, 采用节点渐增法从 2 至 13 逐渐调节隐含层神经元个数, 发现拟合误差都较大; 网络结构采用 2 个隐含层, 采用节点渐增法对两隐含层神经元从 [2,13] 选取并组合测试, 拟合效果较好, 优于单隐层, 故确定 BP 神经网络结构采用双隐层, 如图 2 所示。

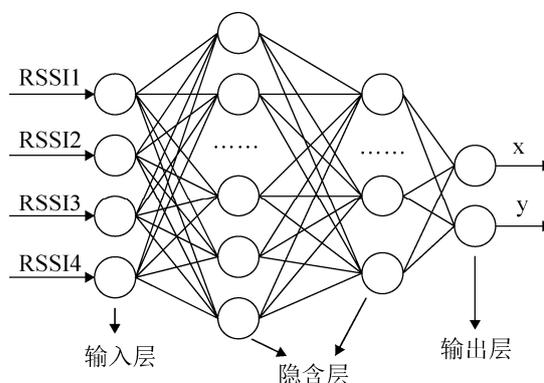


图 2 BP 神经网络结构

Fig. 2 BP neural network structure

2.2 学习算法

BP 神经网络的主要思想是将学习的过程分解为输入信息的正向传播和输出误差信息的反向传播两个阶段。在输入信息的正向传播阶段, 经输入层输入, 然后经过隐含层, 再传向输出层, 在输出节点产生输出信息。如果信息信号在输出层未达到实际的输出值, 则反向传播输出误差信息。在输出

误差信息的反向传播阶段, 未达到实际输出值的误差信号由输出端出发, 逐层反向传播, 在其传播过程中将误差值分配给各层的所有神经元, 然后再根据输出误差信息动态的调整各层神经元的连接权值和阈值。BP 神经网络正是通过这种反复的正向传播和反向调整, 各层神经元的连接权值和阈值得到不断调整, 当输出信号的误差值达到误差限定的范围时, 则停止学习。

首先设置变量和参数:

输入向量: $X_k = [x_1^k, x_2^k, x_3^k, x_4^k]$;

第一隐含层输出向量: $H_k = [h_1^k, h_2^k, \dots, h_m^k]$,

m 为第一隐含层神经元数目;

第二隐含层输出向量: $L_k = [l_1^k, l_2^k, \dots, l_n^k]$, n 为第二隐含层神经元数目;

输出向量: $Y_k = [y_1^k, y_2^k]$;

实际输出向量: $D_k = [d_1^k, d_2^k]$;

输入层至第一隐含层的连接权值: w_{ij}^k , $i=1, \dots, 4$, $j=1, \dots, m$;

第一隐含层至第二隐含层的连接权值: w_{2ij}^k , $i=1, \dots, m$, $j=1, \dots, n$;

第二隐含层至输出层的连接权值: w_{3ij}^k , $i=1, \dots, n$, $j=1, 2$;

第一隐含层输出阈值: θ_{1i}^k , $i=1, \dots, m$;

第二隐含层输出阈值: θ_{2i}^k , $i=1, \dots, n$;

输出层输出阈值: θ_{3i}^k , $i=1, 2$;

误差能量总和: $E(t)$, t 为学习过程的迭代次数;

输出层输出误差: e_i^k , $i=1, 2$;

参数 $k=1, 2, \dots, K$, K 为样本数量。

学习过程的算法步骤如下:

Step1: 初始化权值及阈值, 给 w_{ij}^k 、 w_{2ij}^k 、 w_{3ij}^k 和 θ_{1i}^k 、 θ_{2i}^k 、 θ_{3i}^k 赋予区间 $(-1, 1)$ 内的随机值。

Step2: 输入样本 (X_k, T_k) , 利用 X_k 、 w_{ij}^k 、 θ_{1i}^k 计算第一隐含层的输出 H_k , 计算公式为

$$h_j^k = f_1\left(\sum_{i=1}^4 w_{ij}^k \cdot x_i^k - \theta_{1j}^k\right), \quad j=1, \dots, m \quad (2)$$

传递函数采用正切 S 型函数 tansig , 表达式为

$$f_1(u) = \frac{2}{1 + e^{-2u}} - 1 \quad (3)$$

Step3: 用第一隐含层输出 H_k 、 w_{2ij}^k 、 θ_{2i}^k 计算第二隐含层的输出 L_k , 表达式为

$$l_j^k = f_1\left(\sum_{i=1}^m w_{2ij}^k \cdot h_i^k - \theta_{2j}^k\right), \quad j=1, \dots, n \quad (4)$$

传递函数采用正切 S 型函数 tansig , 表达式同式(3)。

Step4: 用第二隐含层输出 L_k 、 w_{3ij}^k 、 θ_{3i}^k 计算输出层的输出 Y_k , 表达式为

$$y_j^k = f_2\left(\sum_{i=1}^n w_{3ij}^k \cdot l_i^k - \theta_{3j}^k\right), \quad j=1, 2 \quad (5)$$

传递函数采用线性函数 purelin , 表达式为

$$f_2(u) = u \quad (6)$$

Step5: 利用实际输出向量 D_k 与输出向量 Y_k , 计算误差能量总和 $E(t)$ 和输出层误差 e_i^k , 表达式为

$$E(t) = \frac{1}{2} \sum_{i=1}^2 (d_i^k - y_i^k)^2, \quad i=1, 2 \quad (7)$$

$$e_i^k = d_i^k - y_i^k, \quad i=1, 2 \quad (8)$$

若 $E(t)$ 满足期望精度, 转到 **Step9**; 否则转到 **Step6**。

Step6: 判断 $t+1$ 是否大于最大迭代次数, 大于转至 **Step9**; 否则, 利用实际输出、输出层误差 e_i^k 与第二隐含层输出 L_k 来修正连接权 w_{3ij}^k 和阈值 θ_{3i}^k , 表达式为

$$w_{3ij}^k(t+1) = w_{3ij}^k(t) + \eta_1 \cdot \delta_{3j} \cdot l_i^k \quad i=1, \dots, n, \quad j=1, 2 \quad (9)$$

$$\theta_{3j}^k(t+1) = \theta_{3j}^k(t) + \eta_1 \cdot \delta_{3j}, \quad j=1, 2 \quad (10)$$

其中, 输出层局部梯度 $\delta_{3j} = y_j^k(1 - y_j^k)(y_j^k - d_j^k)$, $j=1, 2$, η_1 为第二隐含层至输出层的连接权值修正的学习系数。

Step7: 修正连接权 w_{2ij}^k 和阈值 θ_{2i}^k , 表达式为

$$w_{2ij}^k(t+1) = w_{2ij}^k(t) + \eta_2 \cdot \delta_{2j} \cdot h_i^k, \quad i=1, \dots, m, \quad j=1, \dots, n \quad (11)$$

$$\theta_{2j}^k(t+1) = \theta_{2j}^k(t) + \eta_2 \cdot \delta_{2j}, \quad j=1, \dots, n \quad (12)$$

其中, 第二隐含层局部梯度 $\delta_{2j} = l_j^k(1 - l_j^k) \sum_{i=1}^2 \delta_{3i} \cdot w_{3ij}^k$, $j=1, \dots, n$, η_2 为第一隐含层至第

二隐含层的连接权值修正的学习系数。

Step8: 修正连接权 w_{ij}^k 和阈值 θ_i^k , 表达式为

$$w_{ij}^k(t+1) = w_{ij}^k(t) + \eta_3 \cdot \delta 1_j \cdot x_i^k, \\ i = 1, \dots, 4, \quad j = 1, \dots, m \quad (13)$$

$$\theta_i^k(t+1) = \theta_i^k(t) + \eta_3 \cdot \delta 1_j, \quad j = 1, \dots, m \quad (14)$$

其中, 第一隐含层局部梯度 $\delta 1_j = h 1_j^k (1 - h 1_j^k)$

$\sum_{l=1}^n \delta 2_l \cdot w_{2jl}$, $j = 1, \dots, m$, η_3 为输入层至第一隐含层的连接权值修正的学习系数。

Step9: 判断训练样本是否全部完成, 若是, 学习结束; 否则, 随机选取下一个学习样本向量, 转到 Step2。

2.3 算法的稳定性分析

学习率取值与 BP-VIRE 算法的收敛性有十分重要的关系, 太小会使收敛速度很慢, 增加计算复杂度; 太大又会出现振荡现象而无法收敛, 因此, 必须合理选择 BP-VIRE 算法的学习率。

定理: BP-VIRE 算法中, 当 η_1 、 η_2 、 η_3 取值满足一定条件时, 算法是收敛的。

证明: 取 Lyapunov 函数为

$$V_i(k) = \frac{1}{2} e_i^2(k) \quad (15)$$

则有

$$\Delta V_i(k) = \frac{1}{2} e_i^2(k+1) - \frac{1}{2} e_i^2(k) \quad (16)$$

因为

$$e_i(k+1) = e_i(k) + \Delta e_i(k) =$$

$$e_i(k) + \left(\frac{\partial e_i(k)}{\partial L_k} \right)^T \Delta L_k \quad (17)$$

$$\text{而 } \Delta L_k = -\eta_1 \frac{\partial E}{\partial L_k} = -\eta_1 \frac{\partial E}{\partial e_i(k)} \cdot \frac{\partial e_i(k)}{\partial L_k} =$$

$$-\eta_1 e_i(k) \frac{\partial e_i(k)}{\partial L_k} \quad (18)$$

所以有

$$\Delta e_i(k) = \left(\frac{\partial e_i(k)}{\partial L_k} \right)^T \Delta L_k = \\ -\eta_1 e_i(k) \left\| \frac{\partial e_i(k)}{\partial L_k} \right\|^2 \quad (19)$$

式(16)可改写为

$$\Delta V_i(k) = \Delta e_i(k) \left[e_i(k) + \frac{1}{2} \Delta e_i(k) \right] = \\ -\eta_1 e_i(k) \left\| \frac{\partial e_i(k)}{\partial L_k} \right\|^2 \cdot \left[-\eta_1 + \frac{1}{2} \eta_1^2 \left\| \frac{\partial e_i(k)}{\partial L_k} \right\|^2 \right] = \\ \left\| \frac{\partial e_i(k)}{\partial L_k} \right\|^2 \cdot e_i^2(k) \left[-\eta_1 + \frac{1}{2} \eta_1^2 \left\| \frac{\partial e_i(k)}{\partial L_k} \right\|^2 \right] \quad (20)$$

由式(20)知, 要使算法收敛, 必有

$$-\eta_1 e_i(k) \left[-\eta_1 + \frac{1}{2} \eta_1^2 \left\| \frac{\partial e_i(k)}{\partial L_k} \right\|^2 \right] < 0 \quad (21)$$

因为神经网络学习率 $\eta_1 > 0$, 所以有

$$0 < \eta_1 < \frac{2}{\left\| \frac{\partial e_i(k)}{\partial L_k} \right\|^2} \quad (22)$$

因此, 当 η_1 取值满足式(22)的条件时算法是收敛的。

η_2 和 η_3 证明方法同 η_1 , 不再证明。

3 算法仿真及分析

为了衡量 BP-VIRE 算法定位性能, 采用仿真方法进行 MATLAB 算法实现, 并通过仿真结果与 VIRE 算法进行性能比较。

3.1 仿真场景及参数设置

利用 MATLAB 软件进行仿真实验, 仿真场景图如图 3 所示, 设定在 $8m \times 8m$ 的室内区域中: 4 个读写器放置在平面区域的四个端点上, 坐标分别为 1 号(0, 0)、2 号(0, 8)、3 号(8, 0)、4 号(8, 8), 10 个待测标签的坐标分别为(4.692, 2.091), (3.30, 6.19), (6.70, 1.34), (4.641, 5.819), (5.928, 2.633), (2.738, 1.692), (6.347, 1.833), (6.530, 3.670), (2.429, 7.59), (0.123, 5.976); 假设用于 VIRE 算法的 16 个真实参考标签均匀分布在(1, 1)、(1, 3)、(1, 5)、(1, 7)、(3, 1)、(3, 3)、(3, 5)、(3, 7)、(5, 1)、(5, 3)、(5, 5)、(5, 7)、(7, 1)、(7, 3)、(7, 5)、(7, 7)坐标点上, 将每由 4 个参考标签覆盖的网格分为 5×5 个大小相同的网格单元; 为了与 VIRE 算法性能进行比较, BP-VIRE 算法以同样密

度在边界周围部署参考标签, 加上 VIRE 参考标签共 225 个参考标签, 把 225 个参考标签信号强度及其坐标作为 BP-VIRE 算法训练样本。

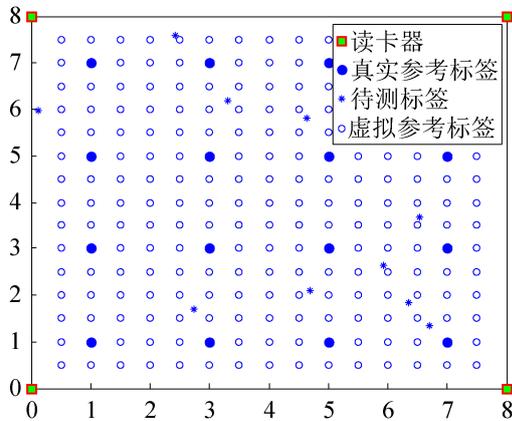


图 3 仿真场景图
Fig. 3 Simulation scenario

本仿真实验根据室内路径损耗模型^[17]计算参考标签的 RSSI 值, 公式为

$$\overline{PL}(d)_{dB} = \overline{PL}(d_0)_{dB} - 10 \cdot n \cdot \lg \frac{d}{d_0} + X_{\sigma}(dB) \quad (23)$$

式中: d 为信号源到目标的实际距离; d_0 为参考距离; n 为路径损耗因子; X_{σ} 为零均值高斯白噪声; $\overline{PL}(d_0)_{dB}$ 信号距离 d_0 处的平均 RSSI 值, $\overline{PL}(d)_{dB}$ 信号到达距离 d 处的平均 RSSI 值。仿真参数设置参考文献[6], d_0 为 0.1m, n 为 2.2, σ 为 2dB。

可通过公式(23)产生参考标签的 RSSI 值, 部分结果如表 1 所示。

3.2 VIRE 算法中阈值对定位性能影响

假设待测标签的实际坐标为 (x_0, y_0) , 定位坐标为 (x'_0, y'_0) , 为了衡量算法性能, 定义定位误差 e 的表达式为:

$$e = \sqrt{(x'_0 - x_0)^2 + (y'_0 - y_0)^2} \quad (24)$$

阈值是 VIRE 算法的重要参数, 直接影响着 VIRE 算法的定位精度。为了定量分析阈值选择对 VIRE 算法定位性能的影响, 分别对 10 个待测点在 VIRE 算法阈值区间为 [0.4, 6] 情况下进行仿真

定位, 计算 10 个待测点的平均误差, 结果如图 4 所示。

表 1 部分参考标签 RSSI 值
Tab. 1 RSSI value of partial reference tags

坐标	1号 RSSI	2号 RSSI	3号 RSSI	4号 RSSI
(0.5, 0.5)	12.278 7	0.067 7	0.067 7	0.031 8
(1.0, 0.5)	4.481 4	0.066 8	0.078 8	0.034 2
(1.5, 0.5)	2.090 7	0.065 2	0.092 6	0.036 7
(2.0, 0.5)	1.166 2	0.063 1	0.110 4	0.039 5
(2.5, 0.5)	0.730 8	0.060 6	0.133 4	0.042 4
(3.0, 0.5)	0.495 7	0.057 8	0.164 3	0.045 4
(3.5, 0.5)	0.356 0	0.054 8	0.206 6	0.048 5

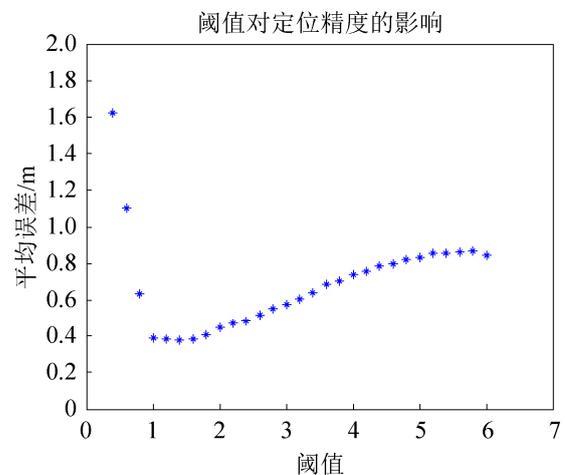


图 4 VIRE 算法中阈值与定位误差的关系曲线
Fig. 4 Relationship between threshold and positioning error

由图 4 可知, 阈值取值在 0.4~1 之间, 定位误差较大, 且对阈值变动比较敏感, 这是由于阈值太小导致部分携带有用信息的参考标签被忽略, 严重影响定位精度; 阈值取值在 1~2 之间, 定位误差较小, 且相对稳定, 取值比较适当; 阈值取值在 2~6 之间, 定位误差缓慢增加, 这是由于阈值较大, 一些无用的虚拟参考标签会被引入, 从而对定位产生干扰, 影响定位精度。

3.3 BP-VIRE 算法神经网络训练

利用 225 个参考标签的 RSSI 值和坐标值作为训练样本, 分别设置训练精度目标值为 0.000 01, 训练循环次数为 500。经多次 BP 神经网络训练并观察拟合效果, 第一隐含层神经元为 9, 第二隐含

层神经元为5时效果最好,均方差变化情况和回归分析图如图5所示。从图5(a)可以看到,横坐标为训练次数,纵坐标为均方差,均方差随着训练次数的增加而逐渐减小,当训练次数为454次的时候,均方差已基本达到稳定值;在图5(b)中,横坐标为目标值,纵坐标为预测值,如果预测值和目标值完全一致,则预测值应该全部落到45°线上,从图5(b)可以看出拟合曲线效果很好,得到的预测数据和目标值非常接近,表明了训练网络的良好预测性能。

神经网络训练各层之间的权重和阈值如表2~4所示。

3.4 BP-VIRE 算法性能分析

利用训练好的神经网络模型,输入10个待测点的RSSI值,定位结果如图6所示,定位误差如表5所示。

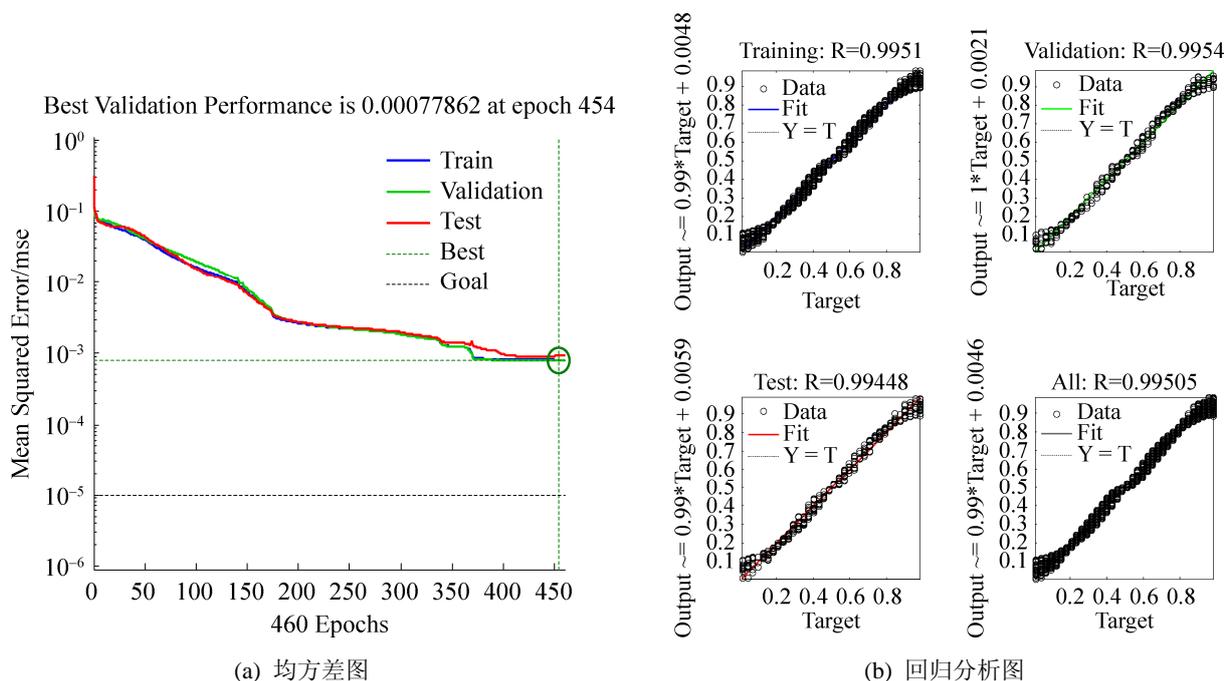


图5 神经网络训练效果
Fig. 5 Neural network training effect

表2 输入层到第一隐含层的权重和第一隐含层阈值

Tab. 2 Weight of the input layer to the first hidden layer and threshold of the first hidden layer

		$w1_{ij}$	$\theta1_i$		
0.701 5	1.854 2	-0.003 2	-1.558 1	-2.324 4	
1.670 1	-2.405 4	-0.067 1	0.479 2	-1.233 5	
1.231 6	0.681 5	0.993 4	1.698 3	-1.206 2	
9.476 1	2.819 9	-4.482 3	-8.792 2	-1.075 0	
13.154 8	5.767 4	-6.646 3	-13.306 4	-1.028 6	
-1.047 8	1.143 0	1.751 6	0.954 5	-0.670 8	
9.830 9	-6.407 0	7.985 8	-8.839 6	2.556 3	
3.827 3	17.240 6	-18.525 9	-3.076 5	-0.494 9	
2.692 1	-2.430 6	2.527 6	-1.902 5	0.940 7	

表 3 第一隐含层到第二隐含层的权重和第二隐含层阈值

Tab. 3 Weight of the first hidden layer to the second hidden layer and threshold of the second hidden layer

				w_{2ij}					θ_{2i}
0.067 5	-0.054 7	1.255 9	0.063 5	0.636 4	-0.078 1	-0.080 6	-0.417 2	0.446 3	1.206 8
0.013 9	-2.176 7	-0.838 6	10.092 7	7.834 5	0.484 4	14.345 9	1.507 8	-0.360 6	-0.813 0
-0.132 6	0.243 2	-0.005 3	-0.056 0	-4.116 5	-0.707 7	8.425 1	-12.515 2	0.750 6	-0.117 3
0.701 2	1.052 3	-0.159 3	-2.897 6	3.643 4	0.290 5	-0.585 2	-1.031 1	4.478 3	1.076 9
-0.487 3	-0.641 1	-1.002 2	10.581 3	16.306 8	0.232 9	4.611 7	16.319 5	1.384 8	-1.380 8

表 4 第二隐含层到输出层的权重和输出层阈值

Tab. 4 Weight of the second hidden layer to the output layer and threshold of the output layer

			w_{3ij}			θ_{3i}
2.036 7	-0.947 9	0.976 5	-0.505 4	0.414 9	-0.021 0	
-0.186 1	-0.526 4	-0.610 3	-0.940 7	-0.016 6	-0.003 6	

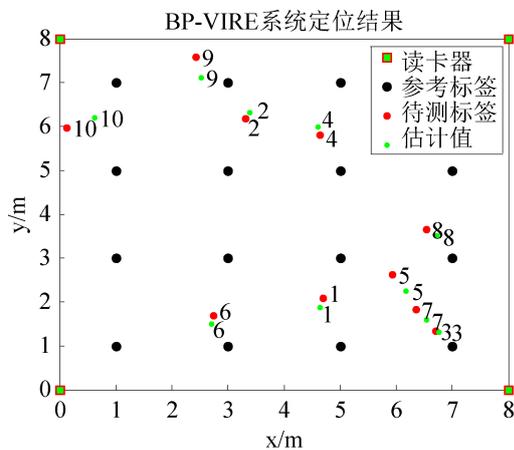


图 6 BP-VIRE 系统定位结果

Fig. 6 BP-VIRE system positioning result

由图 6 和表 5 可以看出, BP-VIRE 定位效果非常好, 平均误差仅为 0.1296 m; 最大误差为待测点 10 的误差 0.5216 m, 约为平均误差的 4 倍, 这是由于其 X 轴坐标为 0.123, 非常靠近定位区域边界处, 一方面边界处参考标签 RSSI 值比较敏感, 另一方面边界处的训练样本缺失, 造成误差较大; 与待测点 10 类似, 待测点 9 的 Y 轴坐标为 7.590, 也靠近定位区域边界处, 其定位误差为 0.3087 m, 边界区域两个待测点 9、10 的误差均值为 0.4151 m, 要大于定位的平均误差; 其余 8 个待测点定位误差都很小, 3 号和 7 号在 0.1 左右, 其余 6 个待测点更是都小于 0.06 m, 表明 BP-VIRE 算法具有非常好的定位性能。

表 5 BP-VIRE 系统定位误差

Tab. 5 BP-VIRE system positioning error

待测点	实际坐标	估计坐标	误差
1	(4.692, 2.091)	(4.676 7, 2.048 1)	0.045 5
2	(3.30, 6.19)	(3.292 2, 6.242 5)	0.053 1
3	(6.70, 1.34)	(6.590 5, 1.437 6)	0.146 6
4	(4.641, 5.819)	(4.647 6, 5.833 7)	0.016 1
5	(5.928, 2.633)	(5.930 9, 2.661 7)	0.028 9
6	(2.738, 1.692)	(2.754 1, 1.666 7)	0.03
7	(6.347, 1.833)	(6.261 2, 1.913 4)	0.117 6
8	(6.530, 3.670)	(6.545 1, 3.693 5)	0.027 9
9	(2.429, 7.590)	(2.632 9, 7.358 2)	0.308 7
10	(0.123, 5.976)	(0.644 1, 5.998 5)	0.521 6

3.5 BP-VIRE 与 VIRE 性能比较

为了更好的说明 BP-VIRE 算法性能, 现与 VIRE 算法进行性能比较。由于对少量标签进行试验随机性较大, 可信度不够高, 因此需要采用大量标签进行随机试验, 利用累积分布函数(cumulative distribution function, CDF)对 BP-VIRE 与 VIRE 定位算法进行性能评价, 表达式为

$$F(x) = P\{X \leq x\} \quad (25)$$

X 为定位误差的随机变量, x 表示误差值, 可以用来表示误差的分布情况。

仿真场景和其他参数不变, 随机产生 1000 个待测点(如图 7), 仿真比较 BP-VIRE 与 VIRE 均在 $N=10$ 下的性能。绘制两种算法下 1 000 个随机产

生的待测点的定位误差比较图如图 8(a)所示, 两种算法的 CDF 曲线比较如图 8(b)所示。

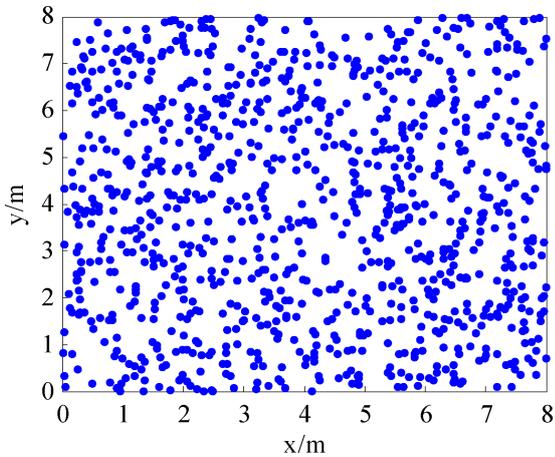
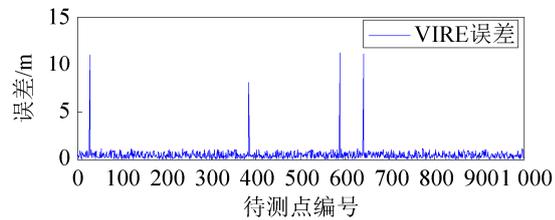
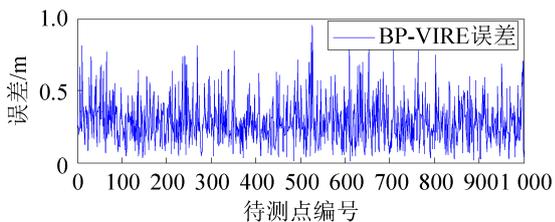
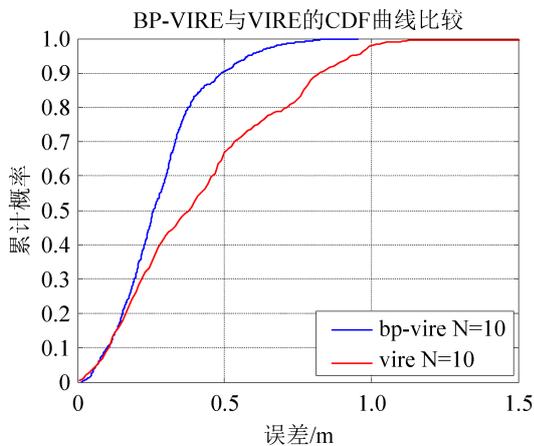


图7 随机产生 1 000 个待测点
Fig. 7 1 000 random samples



(a) 1 000 个样点误差比较图



(b) CDF 曲线比较图

图8 BP-VIRE 与 VIRE 算法性能比较

Fig. 8 Performance comparison of BP-VIRE and VIRE

由图 8(a)可知, BP-VIRE 最大误差为 0.9546, 平均误差为 0.281 6 m, 大于 1 m 的点数为 0; VIRE 最大误差为 11.227 9, 平均误差为 0.462 5 m, 大于 1 m 的点数为 16, 大于 7 m 的点位 4。由图 8(b)可知, BP-VIRE 在误差 < 0.5 m 的概率约为 90%, 而 VIRE 在误差 < 0.5 m 的概率约为 67%; BP-VIRE 在误差 < 1 m 的概率为 100%, 而 VIRE 在误差 < 1 m 的概率约为 98.4%。可以得出: 与 VIRE 相比, ①与 BP-VIRE 平均误差降低了 39.11%, 大大提高了定位精度; ② BP-VIRE 最大误差降低了 91.5%, 仅为 VIRE 的 1/11, 解决了 VIRE 算法边界定位误差较大的问题; ③误差小于 0.5 m 的概率提高 23%, BP-VIRE 定位性能更加稳定可靠。

4 结论

本文针对目前室内定位效果较好的 VIRE 算法中存在的不足, 提出了一种基于 BP 神经网络的 RFID 室内定位算法 BP-VIRE。BP-VIRE 能有效克服 VIRE 的不足, 实现高效稳定的高精度定位。但是, 网络的学习和记忆具有不稳定性, 也就是说, 改变系统参数或增加了学习样本, 训练好的网络需要从头开始, 对于以前的权值和阈值是没有记忆的。由于实验条件的限制, 本文实验数据是通过室内路径损耗模型来获取, 并未在真实环境中验证, 如何使本算法能够在真实的不同复杂程度的实际环境中应用, 实验验证真实环境下算法的可行性和有效性是下一步研究的重点。

参考文献:

- [1] Want R, Hopper A, Falcao V, et al. The active badge location system[J]. ACM Transactions on Information Systems (S1046-8188), 1992, 10(1): 91-102.
- [2] Priyantha N B, Chakraborty A, Balakrishnan H. The Cricket location-support system[C]. Proceedings of MOBICOM 2000, 2000: 32-43.
- [3] Harter A, Hopper A, Steggle P, et al. The anatomy of a context-aware application[J]. Journal of Wireless Networks (S1022-0038), 2002, 8(2/3): 187-197.
- [4] Bahl P, Padmanabhan VN. RADAR: An in-building RF-based user location[C]. Proceedings of IEEE

- INFOCOM 2000, 2000: 775-784.
- [5] 石雪军, 纪志成. 基于射频识别的室内定位系统算法研究[J]. 系统仿真学报, 2015, 27(6): 1294-1300.
Shi Xuejun, Ji Zhicheng. Indoor Positioning System Algorithm Based on RFID[J]. Journal of System Simulation, 2015, 27(6): 1294-1300.
- [6] 史伟光. 基于射频识别技术的室内定位算法研究[D]. 天津: 天津大学, 2012.
Shi Weiguang. Research of indoor localization algorithm based on radio frequency identification technology[D]. Tianjin: Tianjin University, 2012.
- [7] 李程, 钱松荣. 射频识别动态定位方法[J]. 通信学报, 2013, 34(4): 144-148.
Li Cheng, Qian Songrong. RFID dynamic positioning method [J]. Journal on Communications, 2013, 34(4): 144-148.
- [8] Ni I M, Liu Yun-hao, Lau Y C, et al. LANDMARC: Indoor location sensing using active RFID[J]. Journal of Wireless Networks (S1022-0038), 2004, 10(6): 701-710.
- [9] JIANG Xue-jing, LIU Ye, WANG Xiao-lei. An enhanced approach of indoor localization sensing using active RFID[C]. IEEE WASE International Conference on Information Engineering, 2009: 169-172.
- [10] Zhao Yi-yang, Liu Yun-hao, Ni I M. VIRE: Active RFID-based localization using virtual reference elimination[C]. Proc of International Conference on Parallel Processing, 2007: 56.
- [11] Kontkanen P, Myllymaki P, Roos T. Topics in probabilistic location estimation in wireless networks[C]. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 2004, 33(39): 1052-1056.
- [12] 朱凤娟. 射频识别技术的室内定位算法研究[D]. 广州: 华南理工大学, 2010.
Zhu Fengjuan. Research of indoor positioning algorithm based on the RFID technology[D]. Guangzhou: South China University of Technology, 2010.
- [13] 陈冰. 无源 RFID 定位系统搭建与 VIRE 算法优化[D]. 天津: 天津大学, 2011.
Chen Bing. The design of passive RFID location system and improvement of VIRE algorithm[D]. Tianjin: Tianjin University, 2011.
- [14] Kuo R J, Chang J W. Intelligent RFID positioning system through immune-based feed-forward neural network[J]. Journal of Intelligent Manufacturing (S1572-8145), 2015, 26(4): 1-13.
- [15] Hong B W, Huang Y J, Chen C Y, et al. Fuzzy neural network based RFID positioning and navigation method for mobile robots[J]. Research Journal of Applied Sciences, Engineering and Technology (S2040-7459), 2013, 6(7): 1233-1239.
- [16] 陈增强, 国峰. 基于模糊神经网络建模的 RFID 室内定位算法[J]. 系统科学与数学, 2014, 34(12): 1438-1450.
CHEN Zengqiang, GUO Feng. An RFID indoor location algorithm based on fuzzy neural network model[J]. Journal of System Science and Mathematical Science, 2014, 34(12): 1438-1450.
- [17] Rappaport T. Wireless Communications, Principle and Practice[M]. IEEE Press, Prentice Hall, 1996: 3.