

结合 CNN 和 WiFi 指纹库的室内定位算法^①



曹建荣^{1,2}, 张旭¹, 武欣莹¹, 吕俊杰¹, 杨红娟^{1,2}

¹(山东建筑大学信息与电气工程学院, 济南 250101)

²(山东省智能建筑技术重点实验室, 济南 250101)

通讯作者: 曹建荣, E-mail: 317370808@qq.com

摘要: 为了提高基于 WiFi 的室内定位的精度和降低计算时间, 本文提出一种卷积神经网络 (Convolutional Neural Networks, CNN) 结合传统指纹库的室内定位算法. 该算法基于接收信号强度指示 (Received Signal Strength Indication, RSSI) 数据, 首先利用卷积神经网络模型, 根据实时输入数据预判出待测点的初步位置. 在保证大范围预测的位置大概率正确的前提下, 再结合传统指纹库中的指纹点, 确定出精确度更高的最终预测位置. 实验结果表明, 在时效性达到要求的前提下, 累计误差在 1 m 以内的定位精度约有 65%, 累计误差在 1.5 m 以内的定位精度约有 85%, 且误差较为稳定.

关键词: 室内定位; 数据处理; RSSI; 指纹库; 卷积神经网络

引用格式: 曹建荣, 张旭, 武欣莹, 吕俊杰, 杨红娟. 结合 CNN 和 WiFi 指纹库的室内定位算法. 计算机系统应用, 2020, 29(7): 173-179. <http://www.c-s-a.org.cn/1003-3254/7492.html>

Indoor Location Algorithm Combining CNN and WiFi Fingerprint Database

CAO Jian-Rong^{1,2}, ZHANG Xu¹, WU Xin-Ying¹, LYU Jun-Jie¹, YANG Hong-Juan^{1,2}

¹(College of Information and Electrical Engineering, Shandong Jianzhu University, Jinan 250101, China)

²(Shandong Provincial Key of Intelligent Building Technology, Jinan 250101, China)

Abstract: In order to improve the accuracy of WiFi-based indoor positioning and reduce the calculating time, this study proposes an indoor location algorithm combining Convolutional Neural Networks (CNN) with traditional fingerprint library. Based on the Received Signal Strength Indication (RSSI) data, the algorithm first uses the CNN model to predict the initial position of the measured point according to the real-time input data. Under the premise that the large-scale prediction position is guaranteed to be correct, the fingerprint points in the traditional fingerprint database are combined to determine the final prediction position with higher accuracy. The results show that the location accuracy of the error within 1 m is about 65%, the location accuracy of the error within 1.5 m is about 85%, and the error is stable under the premise that the timeliness is required.

Key words: indoor location; data processing; RSSI; fingerprint library; Convolutional Neural Networks (CNN)

1 引言

基于 WiFi 的指纹库室内定位技术是室内定位领域中的重点研究内容之一. 指纹库室内定位技术主

要包含确定性算法、概率型算法、人工神经网络 3 个主要研究方向. 传统的基于指纹库的室内定位方法主要是指: 离线阶段先使用滤波方法 (高斯滤波

① 基金项目: 山东省重点研发计划 (2019GSF111054, 2019GGX104095); 山东省重大科技创新工程 (2019JZZY010120)

Foundation item: Key Research and Development Program of Shandong Province (2019GSF111054, 2019GGX104095); Major Science and Technology Innovation Engineering of Shandong Province (2019JZZY010120)

收稿时间: 2019-12-05; 修改时间: 2020-01-03; 采用时间: 2020-01-14; csa 在线出版时间: 2020-07-03

等)再使用均值求出指纹库;在线阶段使用确定性算法(K近邻等)或概率型算法(贝叶斯概率等)进行指纹匹配.随着科技的不断发展,各行各业对室内定位精度的要求越来越高^[1].传统的室内定位技术难以突破定位精度的瓶颈,而深度学习在图像和语音处理等方面的成熟应用使一些学者把深度学习引入基于WiFi的室内定位领域中,并做了较多的研究^[2,3].Nowicki M等人利用深度神经网络(Deep Neural Network, DNN)有效降低本地化系统成本,证明了堆叠自动编码器可以有效地减少特征空间及实现稳定和精确的分类^[4].Yu等人利用深度卷积神经网络(Deep Convolutional Neural Network, DCNN)和频谱图自动学习无线信道指纹特征,从图像层次表明谱图DCNN模型能够用于指纹识别,其无线信道分类准确率为96.46%^[5].Zhang等人首先采用4层深度神经网络,使用堆叠去噪自动编码器(Stacked Denoising Autoencoder, SDA)学习可靠特征,然后用隐马尔可夫模型(Hidden Markov Model, HMM)的精细定位器来平滑基于DNN的粗定位器的初始定位估计^[6].众多研究成果表明,基于深度学习的室内定位算法在定位精度上有显著提高.

许多研究者都是通过使用复杂的网络结构和长时间调节网络参数寻求最优匹配结果.这无疑增加了前期离线阶段数据采集、训练调整的难度、在线阶段匹配的时间,同时对硬件也提出了更高要求^[7].本文提出了一种基于较少数据,简单计算的匹配算法,即用CNN算法完成初步定位后,再结合传统指纹库进一步确定精度的算法.该方法整体使用的数据量和计算复杂度相对较小,时效性较好,且定位精度进一步提高.

2 算法的基本原理

2.1 算法基本流程

本文提出的算法包括离线指纹库建立和在线定位计算两个阶段.离线阶段时,首先从服务器的数据库中获取AP接收到的全部信息,并预处理数据;然后基于预处理的RSSI数据训练CNN网络模型,得到符合定位精度的CNN定位模型;最后计算已处理的RSSI数据,完成指纹库的建立.在线阶段时,首先通过CNN定位模型初步预测待测点位置,得到最接近某个参考点的位置信息.然后结合传统指纹路径库中的数据,经过距离比例计算进一步缩小预测范围,最终得到较高精

度的待测点坐标.

2.2 原始数据预处理

2.2.1 数据获取

为保证获取的数据更接近实际情况,本文的数据直接从定位系统中实时存储定位信息的数据库中获取.离线阶段,首先获取未处理的UDP报文,完成UDP报文的校验、长度验证和舍弃不完整数据;其次拆分标准UDP报文的关键词.关键词包含4个,分别为上传此UDP报文的AP物理地址、时间戳、发送UDP报文内容的终端MAC地址及当前时刻的RSSI值;最后存入数据库中以备后用.

2.2.2 CNN模型数据处理

假设在被测区域中存在 U 个参考点和 V 个AP,则 V 个AP在第 j 个参考点处构成的一个RSSI值向量 r_j 可表示为:

$$r_j = [r_1, r_2, \dots, r_V], j = 1, 2, \dots, U \quad (1)$$

利用已处理的训练样本数据,把在每一个参考点处采集的 M 个RSSI值向量值整理为如下形式:

$$R_j = \begin{bmatrix} r(1,1) & r(1,2) & \dots & r(1,V) \\ r(2,1) & r(2,2) & \dots & r(2,V) \\ \dots & \dots & \dots & \dots \\ r(M,1) & r(M,2) & \dots & r(M,V) \end{bmatrix} \quad (2)$$

其中, j 取值为 $[1, U]$,且 j 为整数. $r_{(M,V)}$ 表示第 V 个AP在第 j 个参考点采集得到的第 M 个RSSI值. M 值的选择取决于获取的数据量,且 M 值为正整数的平方. V 的取值根据被测区域中实际使用AP的数量.

为简化后期计算,此处对上述 R_j 矩阵做形式和内容两方面变换.首先对 R_j 矩阵中每一个AP获取到的 M 个数据(即 R_j 矩阵中的每一列)均变形,变成 $\sqrt{M} \times \sqrt{M}$ 的矩阵形式.如本文选择 V 值为3,即变换后会形成3个矩阵,分别为 $A_{j,1}$ 、 $B_{j,2}$ 、 $C_{j,3}$;然后添加一个新的子矩阵 $D_{j,4}$,使 R_j 矩阵大小变为 $2\sqrt{M} \times 2\sqrt{M}$.矩阵 $D_{j,4}$ 统一填写相同值,这样既可以保证数据形式一致,也不会对数据内容造成影响.新的 R_j 矩阵大小由原来的 $3 \times M$ 变为 $2\sqrt{M} \times 2\sqrt{M}$.即 R_j 矩阵的形式为:

$$R_j = \begin{bmatrix} A_{j,1} & B_{j,2} \\ C_{j,3} & D_{j,4} \end{bmatrix} \quad (3)$$

其中,矩阵 $A_{j,1}$ 、 $B_{j,2}$ 、 $C_{j,3}$ 、 $D_{j,4}$ 分别为以下形式:

$$\begin{aligned}
 A_{j,1} &= \begin{bmatrix} r_{(1,1)} & r_{(2,1)} & \cdots & R_{(\sqrt{M},1)} \\ r_{(\sqrt{M}+1,1)} & r_{(\sqrt{M}+2,1)} & \cdots & r_{(2\sqrt{M},1)} \\ \cdots & \cdots & \cdots & \cdots \\ r_{(M-\sqrt{M}+1,1)} & r_{(M-\sqrt{M}+2,1)} & \cdots & r_{(M,1)} \end{bmatrix} \\
 B_{j,2} &= \begin{bmatrix} r_{(1,2)} & r_{(2,2)} & \cdots & R_{(\sqrt{M},2)} \\ r_{(\sqrt{M}+1,2)} & r_{(\sqrt{M}+2,2)} & \cdots & r_{(2\sqrt{M},2)} \\ \cdots & \cdots & \cdots & \cdots \\ r_{(M-\sqrt{M}+1,2)} & r_{(M-\sqrt{M}+2,2)} & \cdots & r_{(M,2)} \end{bmatrix} \\
 C_{j,3} &= \begin{bmatrix} r_{(1,3)} & r_{(2,3)} & \cdots & R_{(\sqrt{M},3)} \\ r_{(\sqrt{M}+1,3)} & r_{(\sqrt{M}+2,3)} & \cdots & r_{(2\sqrt{M},3)} \\ \cdots & \cdots & \cdots & \cdots \\ r_{(M-\sqrt{M}+1,3)} & r_{(M-\sqrt{M}+2,3)} & \cdots & r_{(M,3)} \end{bmatrix} \\
 D_{j,4} &= \begin{bmatrix} r_{(1,4)} & r_{(2,4)} & \cdots & R_{(\sqrt{M},4)} \\ r_{(\sqrt{M}+1,4)} & r_{(\sqrt{M}+2,4)} & \cdots & r_{(2\sqrt{M},4)} \\ \cdots & \cdots & \cdots & \cdots \\ r_{(M-\sqrt{M}+1,4)} & r_{(M-\sqrt{M}+2,4)} & \cdots & r_{(M,4)} \end{bmatrix} \quad (4)
 \end{aligned}$$

经过上述变换,在某个参考点处来自3个AP的不同RSSI值就可以通过 R_j 矩阵表示在一张灰度图片上.在权衡数据量能够充分体现RSSI值动态变化和计算耗时两个问题的基础上,本文在训练CNN模型时,选择输入图片像素大小有 16×16 、 28×28 、 36×36 三种.

2.3 卷积神经网络训练

CNN模型一般由卷积层、池化层、全连接层3部分组成.不同的CNN模型通过对上述3部分的结构组合,参数修改等途径解决不同的问题.

从2.2节中可知,所用训练和测试的数据均为一张二维的、代表着一个个具体待测点的指纹灰度图片,这与使用CNN算法进行字符识别类似,即本文待测点处的位置编号与字符识别中的数字类别类似.因此以字符识别的两层卷积(Convolution)、两层池化(Pooling)、随机失活、两层全连接层(Fully Connected)的CNN模型为基础,通过改变卷积层数量、卷积核大小、激活函数以选择性能最佳CNN模型参数,进一步保证在后续参与定位时,定位精度的提高.图1为所使用的CNN模型的基本结构.

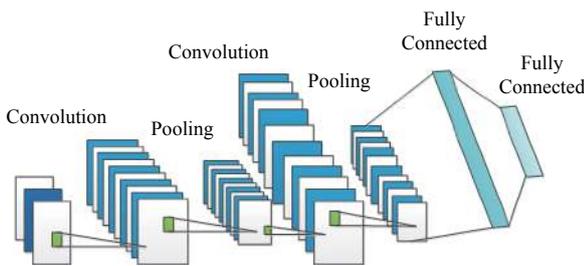


图1 CNN基本结构图

以 28×28 的图像为例,在第一层卷积层中,首先使用32个 5×5 的卷积核, 2×2 的最大池化(步长=1)对图像操作,形成32个 14×14 的特征图;然后再使用64个 5×5 卷积核, 2×2 的最大池化(步长=1)操作,形成 $7 \times 7 \times 64$ 的特征图;接着变换数据格式,生成 $1 \times (7 \times 7 \times 64)$ 的数据用于分类.使用的ReLU、Elu^[8]两种激活函数如下所示.

ReLU 激活函数:

$$y = \begin{cases} 0 & (x \leq 0) \\ x & (x > 0) \end{cases} \quad (5)$$

Elu 激活函数:

$$y = \begin{cases} \gamma(e^x - 1) & (x \leq 0) \\ x & (x > 0) \end{cases} \quad (6)$$

ReLU 激活函数是分段线性函数,属于单侧抑制函数.它使神经元具有了稀疏激活性;Elu 激活函数存在负值,可以将激活单元的输出均值向0推近,具有批量归一化的效果,而且减少了计算量.

为了防止训练过拟合和训练出的模型相对简洁,在全连接层后添加Dropout层,并取值为0.5.训练模型使用反向传播算法训练整个网络.训练CNN模型能够找到代表参考的类与位置信息的非线性映射关系^[9],即当相邻迭代之间的损失函数下降到阈值以下或者满足迭代次数时,网络达到稳定则保存网络参数,停止训练;否则,从新的训练样本集中选择输入继续训练.通过多次训练后得出效果最好的CNN模型,形成基于卷积神经网络的指纹库模型.

2.4 指纹库数据处理

指纹库形成的基础是能够测出在定位点处具有代表性的RSSI值.现实中同一参考点处的RSSI值不同的原因主要有两个,分别为随时间、空间小范围内动态变化和信道传输错误出现偶然异常值.针对这两个问题,本文提出了基于限幅加权的滤波方法用于确定传统指纹库的指纹.对所有获取到的数据,做以下3步处理:首先使用 2δ 准则对处理的数据做限幅处理,去除因偶然因素引起的极端值;然后使用自然断点分类(Jenks Natural Breaks)^[10],得到若干代表性分类和有界终端点集合;最后对这些数据加权,得到某AP在此点最具代表性的RSSI值.

具体步骤如下:

Step 1. 计算高斯分布的标准差,筛选出符合 2δ 原则的集合 Ω_n ;

$$\begin{cases} u_0 = \frac{1}{n} \sum_{i=1}^n x_i \\ \sigma = \sqrt{\frac{1}{n} \left(\sum_{i=1}^n (x_i - u_0)^2 \right)} \\ \Omega_n = \{x_i | u_0 - 2\sigma \leq x_i \leq u_0 + 2\sigma\} \end{cases} \quad (7)$$

其中, $i \in [1, n]$, 且 i 为整数, x_i 为原始数据, n 、 u_0 、 σ 分别为原始数据的总量、平均值、标准差。

Step 2. 自然断点分类法与 K 均值聚类在一维数据聚类分析中意义是一样的. 从数据集本身出发, 按照类内方差最小, 类间方差最大的原则, 对数据迭代和移动聚类. 其中最主要的是确定聚类数量 k 值和使用方差拟合优度 (Goodness of Variance Fit, GVF) 算法判断分类效果好坏。

$$\begin{cases} \alpha = \frac{1}{N} \sum_{i=1}^N (y_i - u_e)^2 \\ \beta = \sum_{j=1}^k \frac{1}{N_j} \sum_{i=1}^{N_j} (y_{ij} - u_j)^2 \\ GVF = 1 - \frac{\beta}{\alpha} \end{cases} \quad (8)$$

其中, $i \in [1, N]$, $j \in [1, k]$, $i, j \in Z$; y_i 、 N 、 u_e 、 α 分别表示集合 Ω_n 中的第 i 个数据、数据数量、平均值、方差; y_{ij} 、 N_j 、 u_j 、 β 表示把集合 Ω_n 分成 K 类后, 第 j 类中的第 i 个数据、第 j 类中数据的数量、第 j 类中数据的平均值、 k 类方差的和。

GVF 用于判定不同分类的分类效果好坏. 在一定范围内, GVF 值越大, 其分类效果越好, 分类结果越具有代表性. 一般经验值指出, $GVF > 0.7$ 就可以接受分类结果。

通过 Step 2, 把集合 Ω_n 划分成了 k 类小集合 $\{\Theta_1, \Theta_2, \dots, \Theta_n\}$, 以及 $(k+1)$ 个各类的边界点组成的集合 $\{t_1, t_2, \dots, t_{k+1}\}$ 。

Step 3. 对计算出的 $\{\Theta_1, \Theta_2, \dots, \Theta_n\}$ 中的各类数据做加权处理, 得到此参考点处某一个 AP 的最终 RSSI 值。

$$\begin{cases} R_{(U,V)} = \sum_{j=1}^k w_j u_j \\ w_j = F(t_{j+1}) - F(t_j) \\ F(t_j) = P\{X \leq t_j\} \end{cases} \quad (9)$$

其中, $j \in [1, k]$, $j \in Z$. $R_{(u,v)}$ 为第 U 考点的第 V 个 AP 最终值, w_j 为第 j 据的权值, 函数 $F()$ 为概率分布函数, 用于计算各类出现的概率。

通过以上 3 个步骤计算每一个参考点所有 AP 值

序列, 最终得到指纹库 Ψ 。

$$\Psi = \begin{bmatrix} R(1,1) & R(1,2) & \dots & R(1,V) \\ R(2,1) & R(2,2) & \dots & R(2,V) \\ \dots & \dots & \dots & \dots \\ R(U,1) & R(U,2) & \dots & R(U,V) \end{bmatrix} = \begin{bmatrix} \Psi_1 \\ \Psi_2 \\ \dots \\ \Psi_U \end{bmatrix} \quad (10)$$

为防止因信道重叠导致 RSSI 值在传输过程中出现误差和保证 RSSI 值动态变化的完整性^[11], V 值一般取为 3。

限幅加权的滤波方法最终目的是在保留了指纹自身的特征后, 最大限度地去除异常数据. 根据实际处理大量数据后, 无效数据约占中数据的 5%, 使用 2δ 准则, 能够直接去除来自异常情况、小概率出现的无效数据. 通过使用 GVF 算法和类内数据加权计算出最具代表性的指纹点的 RSSI 值. 以上 3 个步骤, 能够从参与定位的开始就保证了数据的有效性. 通过使用限幅加权的滤波方法能够保证参与整体定位的数据的有效性和提高最终的定位精度, 能够有效减少或避免在线定位时结果畸变。

2.5 在线定位

在线阶段定位共有 4 个步骤, 主要完成 CNN 模型的结果与传统指纹库指纹的结合计算. 4 个步骤如下:

Step 1. 预处理获取的实时数据, 然后使用 CNN 模型预测得到位置序号对应坐标点 $C(X_c, Y_c)$. 3 个 AP 在某一时刻获取的数据经过 2.4 节中的 Step1 初步去除异常数据后, 得到包含实时数据矩阵 H 。

Step 2. 实时矩阵 H 中的每一组数据均与指纹库 Ψ 中的所有指纹做欧式距离计算, 以最小欧式距离值对应的指纹为此组数据的接近指纹, 最终形成接近指纹库 Ψ' 。

Step 3. 统计出接近指纹库 Ψ' 同接近指纹数量排三位的接近指纹, 找到对应的坐标点集合 $\Gamma((X_a, Y_a), (X_b, Y_b), (X_d, Y_d))$. 计算出其各坐标点到 $C(X_c, Y_c)$ 的距离 l_a 、 l_b 、 l_d 及距离之和 l_s 。

Step 4. 计算出待测点的最终预测位置 (X_f, Y_f) , 计算公式如下:

$$\begin{cases} X_f = X_a l_a / l_s + X_b l_b / l_s + X_d l_d / l_s \\ Y_f = Y_a l_a / l_s + Y_b l_b / l_s + Y_d l_d / l_s \end{cases} \quad (11)$$

3 实验结果分析

3.1 实验设置

根据被测面积大小和 AP 的个数合理划分参考点

的位置. 本实验选取面积为 $14.4\text{ m}\times 8.6\text{ m}$ 的实验教室, 以 28 个 $2\text{ m}\times 2\text{ m}$ 的网格顶点, 作为训练 CNN 模型的指纹库参考点; 以 112 个 $1\text{ m}\times 1\text{ m}$ 的网格顶点作为传统指纹库参考点. 被测区域示意图如图 2(小圆点、大圆点分别为 $1\text{ m}\times 1\text{ m}$ 和 $2\text{ m}\times 2\text{ m}$ 的网格参考点) 所示. 在选取参考点时, CNN 模型指纹库的参考点位置和传统指纹库的参考点位置应完全不重合.

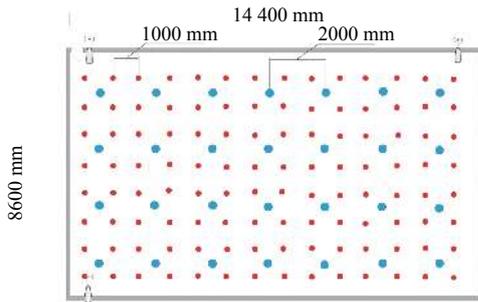


图2 被测区域示意图

移动终端信号的接收能力、所在高度、移动终端和 AP 朝向^[12]等因素均会引起 RSSI 值得变化. 因此在 3 个 AP 收集数据时, 使用 5 种不同型号的移动终端, 分别距地面垂直距离为 1 m、1.5 m、2 m 的高度以及 4 个朝向发射数据, 尽可能保证每一个参考点数据的完整性.

本文设定 CNN 模型指纹库参考点的个数为 28, 所以用于训练得数据集也应当包括 28 类. 每一个参考点训练集和测试集图片数量分别为 28×300 、 28×100 张, 且训练集和测试集的数据均来源于对 28 个参考点的均匀采样. 每张图片中包含 3 个 AP 在此参考点处采集的共 $3\times 14\times 14$ (以 28×28 的图片输入为例) 个 RSSI 值, 使用 Python 等软件搭建仿真实验平台.

3.2 实验结果及分析

通过 1.2.2 节的数据预处理过程, 可以获得 CNN 模型训练使用 16×16 、 28×28 、 36×36 三种格式的图片. 这 3 种输入图片的本质区别在于包含 RSSI 值的多少. 图片越大, 其相应的训练时间就会越长. 图 3 左侧为用于 CNN 模型训练的, 格式大小 28×28 的某参考点预处理的图片, 右侧是其局部放大图的截图. 图 3 左侧图像数据的 4 部分内容分别对应 3 个 AP 采集的此参考点的 RSSI 值矩阵和填补的空白值矩阵; 图 3 右侧图像为左侧图中某一个 AP 采集的数据放大图, 可以看出, 此 AP 采集的某参考点的 RSSI 值是动态变化的. 图 4 为指纹库中的某点数据预处理的过程结果

图. 图 4 中五角形点、三角形点、小圆点分别代表 3 个 AP 在被第一步限幅之后的数据, 大圆点代表第二步分类后的各类边界点, 各 AP 在此参考点处的最终 RSSI 值如图 4 中各类线所示.

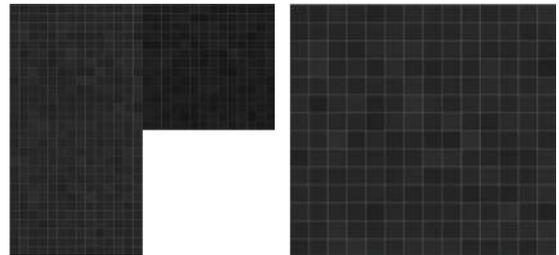


图3 某参考点采集数据成像图及局部图

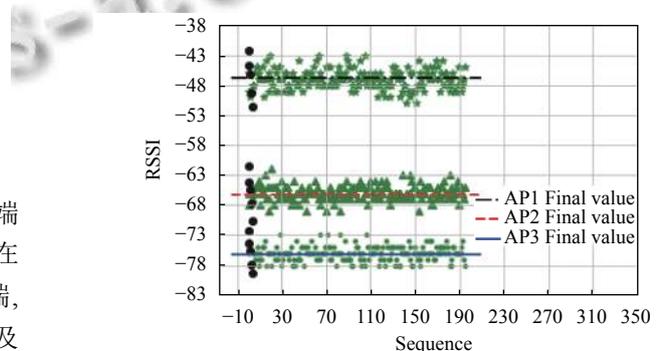


图4 某参考点指纹库数据处理过程图

根据 CNN 模型设置, 采用不同卷积核大小、激活函数等得到训练的时间、准确率如表 1 所示, 网络损失收敛图如图 5 所示.

表1 不同参数性能比较

算法		16×16		28×28		36×36	
		Tr A	Te A	Tr A	Te A	Tr A	Te A
ReLU	3×3	0.93	0.902	0.91	0.907	0.95	0.912
	5×5	0.91	0.877	0.93	0.914	0.96	0.915
Elu	3×3	0.89	0.890	0.92	0.891	0.94	0.868
	5×5	0.90	0.881	0.94	0.904	0.97	0.890

通过表 1 的数据可以看出, 当输入数据包含信息增加, 其训练准确率越高. 但是测试集并未随之增加而显著提高; 对于同一批数据, 准确率会随着卷积核模板的规格增加也呈现上升的趋势.

图 5 是以 28×28 的图像、 5×5 卷积模板为例, 分别在 ReLU、Elu 两个激活函数下的网络损失收敛图. 结合表 1 的数据可以看出, 在相同的数据基础上, Elu 比 ReLU 少约 200 轮训练达到最佳收敛值, 而且训练效果相当.

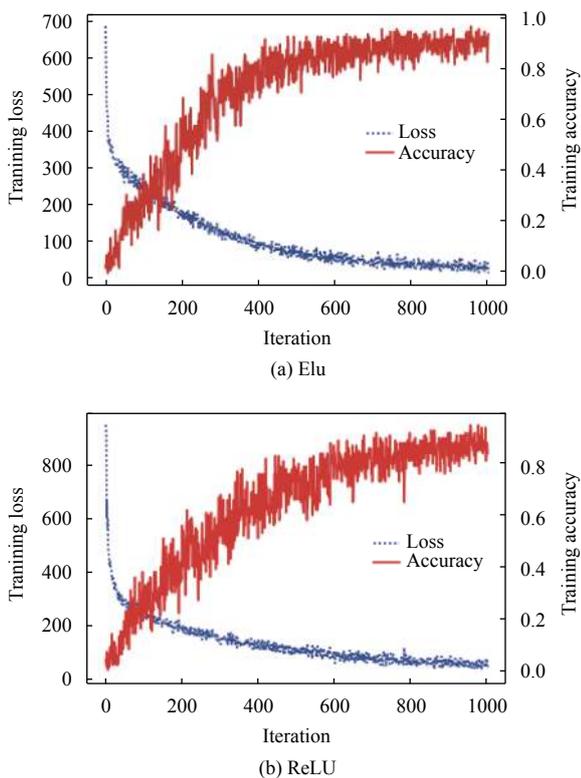


图5 ReLU、Elu 激活函数下网络收敛图

考虑到在实际应用中采集的数据数量和匹配时间周期两个主要因素, 本文最终选取 Elu 激活函数、 28×28 的输入数据、 5×5 的卷积模板的 CNN 模型, 并在此模型下保存最佳训练参数。

本文通过实验得到的定位误差累计分布图如图 6 所示, 其中横坐标表示定位误差值, 纵坐标表示定位误差的累积比例。图 6 中 CNN1-JK 曲线为本文算法结果; CNN1 曲线为只使用本文算法中的 CNN 模型得到的结果; CNN2 模型曲线是以本文 CNN 模型为基础, 提高了分类精度 (112 类) 的算法结果; G-K 曲线为使用高斯滤波结合 K 近邻算法得到的结果。

从图 6 中可以看出, 从 CNN1 和 CNN2 模型的结果中可以看出, 增加 3 倍分类精度后, 定位误差并没有成比例显著降低, 但是 CNN2 模型前期训练的训练耗时却是 CNN1 模型的两倍到三倍; 在单纯考虑累积误差时, 本文算法结果与 CNN2 模型结果相比, 累积误差约在 0.75 m 之后本文算法体现出定位精度优势; 本文算法结果与传统的 G-K 算法结果相比, 明显的提高了定位精度。通过实验结果对比发现, 本文的算法误差有近 50% 在 0.5–1 m 内, 约 90% 均小于 1.5 m, 相比于同

类型单纯的卷积神经网络和传统算法, 准确度比较高, 误差稳定性较好。

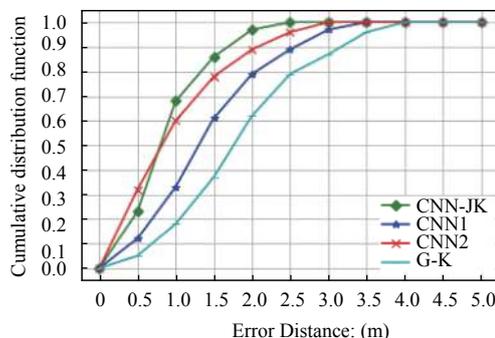


图6 定位误差累计分布函数图

图 7 为测试点测试的实际结果。区域 I–IV 分别是待测点在大圆点连线中心点、与大小圆点重合、小圆点连线中心点、一般位置的预测结果 (三角形点为实际位置、五角形点为预测位置)。实验结果大部分均在 0.5–1.5 m 之间, 实际结果与图 6 的误差累积结果基本相同, 精度在可控范围内。

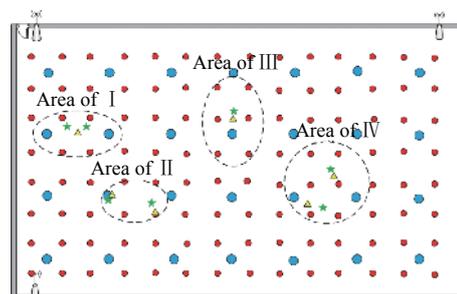


图7 典型实际定位数据结果示意图

本文算法利用 CNN 模型能够较好地学习 RSSI 的波动特性的特点, 首先使用 CNN 模型预测粗略位置, 再配合传统指纹库指纹简单计算确定最终定位。选择以字符识别的 CNN 模型为基础, 更符合指纹定位的数据特点; 使用限幅加权的滤波方法, 从数据参与定位的源头上减少了因异常值引起的定位即便问题的出现。整体上一方面可以提高定位范围准确性, 防止发生 RSSI 值局部畸变而导致整体定位错误; 另一方面整体使用数据量较少, 定位速度较快。

4 结论

将深度学习应用在室内定位领域中提高定位精度有着重要的意义。本文提出一种卷积神经网络结合传

统指纹库的室内定位算法. 首先利用卷积神经网络模型, 根据实时输入信号强度指示 RSSI 数据预判出待测点的初步位置, 再结合传统指纹库中的指纹点, 确定出精确度更高的最终预测位置. 实验结果表明, 在时效性达到要求的前提下, 误差在 1 m 以内的定位精度约为 65%, 误差在 1.5 m 以内的定位精度约为 85%, 定位精度高于传统的基于指纹库的定位方法, 且误差较为稳定.

参考文献

- 1 Jung SH, Moon BC, Han D. Unsupervised learning for crowdsourced indoor localization in wireless networks. *IEEE Transactions on Mobile Computing*, 2016, 15(11): 2892–2906. [doi: [10.1109/TMC.2015.2506585](https://doi.org/10.1109/TMC.2015.2506585)]
- 2 Liu H, Darabi H, Banerjee P, *et al.* Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 2007, 37(6): 1067–1080. [doi: [10.1109/TSMCC.2007.905750](https://doi.org/10.1109/TSMCC.2007.905750)]
- 3 Mainetti L, Patrono L, Sergi I. A survey on indoor positioning systems. *Proceedings of the 22nd International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*. Split, Croatia. 2015. 111–120.
- 4 Nowicki M, Wietrzykowski J. Low-effort place recognition with WiFi fingerprints using deep learning. *arXiv: 1611.02049*, 2016.
- 5 Yu YL, Liu FX, Mao S. Fingerprint extraction and classification of wireless channels based on deep convolutional neural networks. *Neural Processing Letters*, 2018, 48(3): 1767–1775. [doi: [10.1007/s11063-018-9800-1](https://doi.org/10.1007/s11063-018-9800-1)]
- 6 Zhang W, Liu K, Zhang WD, *et al.* Deep neural networks for wireless localization in indoor and outdoor environments. *Neurocomputing*, 2016, 194: 279–287. [doi: [10.1016/j.neucom.2016.02.055](https://doi.org/10.1016/j.neucom.2016.02.055)]
- 7 Nuño-Maganda MA, Herrera-Rivas H, Torres-Huitzil C, *et al.* On-device learning of indoor location for WiFi fingerprint approach. *Sensors*, 2018, 18(7): 2202. [doi: [10.3390/s18072202](https://doi.org/10.3390/s18072202)]
- 8 Yang L, Fan CX, Li Y, *et al.* Improving deep neural network with multiple parametric exponential linear units. *Neurocomputing*, 2018, 301: 11–24. [doi: [10.1016/j.neucom.2018.01.084](https://doi.org/10.1016/j.neucom.2018.01.084)]
- 9 Ferreira BV, Carvalho E, Ferreira MR, *et al.* Exploiting the use of convolutional neural networks for localization in indoor environments. *Applied Artificial Intelligence*, 2017, 31(3): 279–287.
- 10 North MA. A method for implementing a statistically significant number of data classes in the jenks algorithm. *Proceedings of the 6th International Conference on Fuzzy Systems and Knowledge Discovery*. Tianjin, China. 2009. 35–38.
- 11 Wu B, Ma ZX, Poslad S, *et al.* An efficient wireless access point selection algorithm for location determination based on RSSI interval overlap degree determination. *Proceedings of IEEE 2018 Wireless Telecommunications Symposium (WTS)*. Phoenix, AZ, USA. 2018. 1–8.
- 12 Lee YC, Park SH. RSSI-based fingerprint map building for indoor localization. *Proceedings of the 10th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*. Jeju, Republic of Korea. 2013. 292–293.