

Fig.1 System model

图 1 系统模型

1.2 定位过程

如图 2 所示,我们提出的基于 CNN 的混合指纹定位系统的定位过程包括两个阶段:离线阶段和在线阶段.

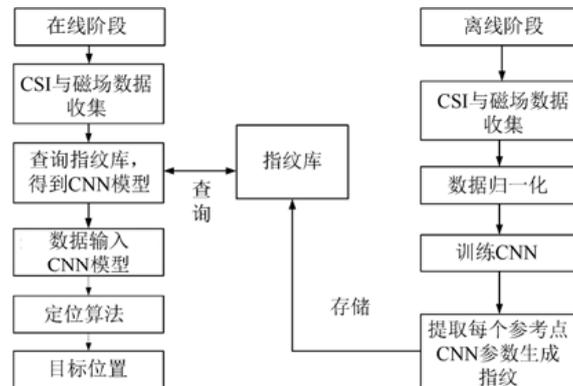


Fig.2 Localization process

图 2 定位过程

首先,我们在离线阶段时,使用每个参考点采集的 CSI 和磁场数据训练得到一个 CNN 模型.然后我们把这个模型的参数和参考点的坐标作为指纹存储到指纹库中.在实时定位阶段,采集到 CSI 和磁场数据后,我们查询指纹库,取出每个参考点的 CNN 模型参数生成 CNN 模型,然后将这个数据通过由每个参考点训练而来的 CNN 模型,将原始的数据和训练后的数据输入一个径向基函数,就可以分别得到目标在每个参考点的概率,通过贝叶斯法则,我们可以得到目标的估计位置.

在本文中,CSI 全称为信道状态信息,通过正交频分复用,可以将 CSI 从物理层解析出来,它代表着无线信号发射器和接收器之间的信道响应信息.在无线通信领域中,CSI 指的是通信链路的信道特性,描述了信号收发过程中信号的衰减因素,包括散射、环境衰减、距离衰减等.具体而言,CSI 属于网络七层协议中的物理层,它是一系列子载波的集合,每个子载波都有不同的相位和幅度,因此它可以很好的分辨多条路径.在平滑的衰落信道中,CSI 在频域中可以用如下公式表示.

$$y = Hx + n,$$

其中, $n$  表示高斯噪声, $H$  表示 CSI 信息, $y$  和  $x$  分别代表接收器和发射器的信号.

多个子载波上的 CSI 可以表示为

$$H = [H_1, H_2, \dots, H_i, \dots, H_N],$$

$$H_i = \|H_i\|^{j\angle H_i},$$

其中, $N$  表示根据驱动被分成的子载波组数,比如通常  $N=30$ , $H_i$  表示子载波  $i$  上的 CSI,  $\|H_i\|$  和  $\angle H_i$  分别表示子

载波  $i$  的幅度和相位.与数据链路层上的接收信号强度(RSS)相比,物理层上的 CSI 在无干扰情况下具有更好的稳定性.

## 2 基于 CNN 的指纹定位算法

### 2.1 数据归一化

在本文中,在采样点获得的数据由 3 部分组成:磁场的三维坐标( $m_1, m_2, m_3$ ),从  $L$  个 AP 中收集的 CSI 幅度 ( $H^1, H^2, \dots, H^L$ )以及采样点的位置坐标  $C=(x_c, y_c)$ .由于我们的输入数据来自于不同的量纲,为了更好地处理数据,加快训练网络的收敛性,我们需要将有量纲的数据转化为无量纲的数据,并将其映射到[0,1]区间内,具体计算公式如下:

$$data' = \frac{data - \min(data)}{\max(data) - \min(data)} \quad (1)$$

这里,  $data$  表示指纹中的每一个数据.  $data'$  是归一化后的采样数据,处理过后就可以用来作为 CNN 的输入了.

### 2.2 CNN 模型训练

在对数据进行预处理后,我们需要做的是用这些数据训练 CNN 网络,并提取权值作为指纹.卷积神经网络(CNN)最初是被用于手写字体识别.随着研究者对 CNN 网络性质的不断发掘,它的应用逐渐遍布很多领域,比如现在流行的面部解锁技术和图片识别技术等.如图 3 所示,卷积神经网络的典型结构通常包括输入层、卷积层、下采样层(池化层)、全连接层和输出层.

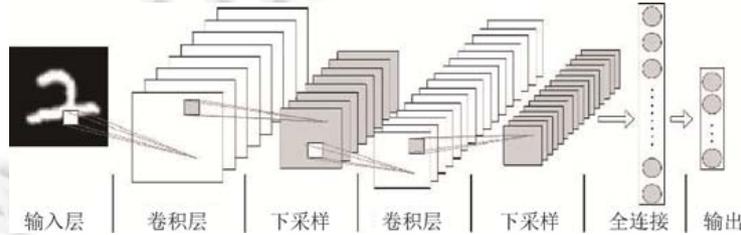


Fig.3 Structure of CNN model

图 3 CNN 模型结构

CNN 网络模型通过多个卷积层和池化层对输入的矩阵数据  $T$  进行处理,达到降维、升维或者特征提取的目的.CNN 网络中第  $i$  层的输出矩阵可以用  $H_t$  表示,这里  $H_{t_0}=T$ ,如果第  $i$  层是卷积层,那么  $H_t$  可以描述为

$$H_t = f(W_i \otimes H_{t_{i-1}} + b_i) \quad (2)$$

其中,  $W_i$  代表卷积网络中第  $i$  层卷积核的权值.  $\otimes$  表示卷积的意思:  $W_i$  与网络中第  $i-1$  层的特征矩阵  $H_{t_{i-1}}$  进行卷积,  $b_i$  是当前层的偏移向量,  $f(\cdot)$  是一个非线性函数的激励.

池化层的作用是减少特征矩阵的维数,在一定程度上保证特征的尺度不变性.通过池化可以减少卷积后矩阵的维度,常用的池化方法有两种:最大值子采样(max pooling)和均值子采样(average pooling),对卷积后的特征矩阵进行池化(也称“下采样”),可以得到一个池化后的特征矩阵  $H_t$ .

$$H_t = \text{pooling}(H_{t_{i-1}}) \quad (3)$$

输入矩阵  $T$  经过多对卷积层和池化层后,通过全连接层进行分类,完成从输入特征矩阵到输出  $Y_t$  的映射.

为了训练卷积神经网络,使它产生需要的输出,可以通过最小化网络的损失函数  $L(W, b)$  来训练 CNN 网络的权值.负对数似然(NLL)和均方误差(MSE)是卷积神经网络中常用的损失函数.

$$NLL(W, b) = -\sum_{i=1}^{|Y_t|} \log Y_t \quad (4)$$

$$MSE(W, b) = \frac{1}{|Y_t|} (Y_t - \hat{Y}_t)^2 \tag{5}$$

这里,  $\hat{Y}_t$  是指理想的输出矩阵,在本文中一般令  $\hat{Y}_t = T$  .

对于权值更新的算法,常用的更新算法是迭代的随机梯度下降法,通过不同的样本数据依次更新权值,使损失函数最小化:

$$W_i = W_i - \eta \frac{\partial L(W, b)}{\partial W_i} \tag{6}$$

$$b_i = b_i - \eta \frac{\partial L(W, b)}{\partial b_i} \tag{7}$$

这里,  $\eta$  表示学习速率,它决定了梯度下降的步伐.随着训练样本的迭代与权值的更新,只要网络的损失函数  $L(W, b)$  收敛于某个值或者小于设定的阈值,就可以认为参数的训练已经完成,可以将训练过程停止.这个时候 CNN 网络中的权值  $(W, b)$  会被固定下来,在测试数据的时候不会被更新.

考虑到本文使用的数据量不是特别大,我们最终使用的是 7 层的 CNN 模型,如图 4 所示.

由于 CNN 模型要求输入必须为矩阵,且这个矩阵行数必须等于列数,而我们获得数据只是一个一维的数组,为了满足 CNN 网络的要求,我们需要将我们的数据中填充 0 值,然后将其由一个数组转化为一个方阵,具体的构造过程如下所示.

第 1 步是构造数据.样本数据集中每条样本是一个一维数组,长度为  $(3+L)$ ,其中  $L$  为 AP 点的数目.首先,我们将样本集中的每一条数据都进行归一化处理;然后我们往样本数据中填充 0 值,直到样本数据长度恰好等于某个数的平方.最后我们将其转为方阵  $T$ .

第 2 步是输入数据.将方阵数据作为输入,经过两个卷积层和池化层,还有一个全连接层之后,我们可以得到一个最终的输出  $Y_t$ .

第 3 步为训练权值.使用误差反向传播算法<sup>[15]</sup>对 CNN 网络的权值进行更新.

最终得到数据库中第  $i$  个参考点的指纹为  $F(i) = \{(W_{1i}, b_{1i}), (W_{2i}, b_{2i}), (W_{3i}, b_{3i}), (W_{4i}, b_{4i}), (W_{5i}, b_{5i}), C_i\}$ .

### 2.3 指纹匹配

在线阶段中,我们运用一种概率的方法去匹配指纹库,并估计用户的位置.基于贝叶斯法则,我们可以计算测试点位于采样点  $i$  的概率  $\text{Pr}$ .

$$\text{Pr}(C_i | T) = \frac{\text{Pr}(C_i) \text{Pr}(T | C_i)}{\sum_{i=1}^{N_c} \text{Pr}(C_i) \text{Pr}(T | C_i)} \tag{8}$$

这里,  $N_c$  表示参考点的数量,  $C_i$  是参考点  $i$  在指纹库中的位置坐标,  $\text{Pr}(C_i)$  是目标位于参考点  $C_i$  处的先验概率,我们可以假设  $\text{Pr}(C_i)$  服从均匀分布.那么公式(8)可以简化为



Fig.4 Structure of CNN model used in this paper

图 4 本文的 CNN 结构模型

$$\Pr(C_i | T) = \frac{\Pr(T | C_i)}{\sum_{i=1}^{N_c} \Pr(T | C_i)} \quad (9)$$

基于深度网络的模型,我们可以认为  $\Pr(T | C_i)$  是一种径向基函数<sup>[16]</sup>,用来描述输入数据  $T$  在参考点  $C_i$  处的输出  $Y_i$  与它本身的相似度.它可以被表示为

$$\Pr(T | C_i) = \exp\left(-\frac{1}{\lambda_T \sigma_T \|T - Y_i\|}\right) \quad (10)$$

这里,  $\sigma_T, \lambda_T$  分别为输入数据  $T$  的方差和方差参数.

最终,目标的估计位置为所有参考点的位置的加权平均:

$$\hat{C} = \sum_{i=1}^{N_c} \Pr(C_i | T) C_i \quad (11)$$

### 3 实验和结果分析

#### 3.1 实验环境与数据处理

实验中使用了 3 个无线路由器作为数据发送器,两个型号是 TP-LINK 66E4,一个型号是 360 mini 安全路由器. TP-LINK 路由器有 3 根天线,360 路由器有两根天线,因此我们一共使用了 8 根发送天线,它们都工作在 2.4GHz 频段.实验中使用一台 DELL 笔记本电脑作为数据接收端.我们在实验前就在里面装上了 Intel 5300 无线网卡,就可以在电脑上接收并处理 CSI 信号.

如图 5 所示,我们在实验室里进行实验,房间里摆放着桌子,电脑和一些 AP 设备.我们设置了 3 个 AP 点来覆盖整个房间.

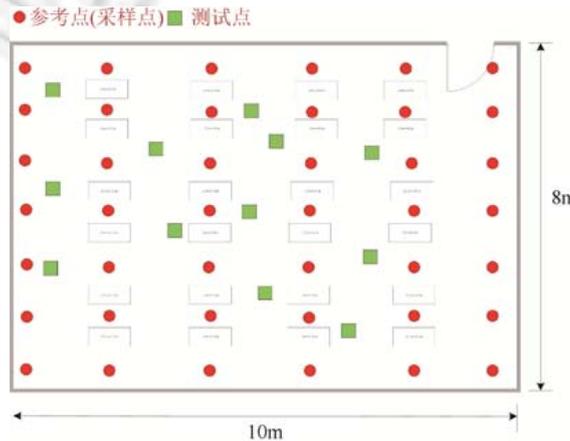


Fig.5 Plan of laboratory

图 5 实验室平面图

数据收集的步骤如下.

(1) 确定实验区域,以墙脚为原点,建立坐标系并划分网格.由于实验环境中每块地板砖的边长是相同的,我们每隔一块地板砖取一个点作为参考点,依次编号并记录下它的位置.

(2) 摆放并固定好 3 个 AP 的位置,把它们全部开启.

(3) 在电脑上重装 Ubuntu 系统,安装好 Linux-802.11n-csi-tool.实验人员甲在每个参考点上停留,把电脑摆在同样高度的凳子上,且保持电脑朝向一致.然后操作采集 500 个 CSI 数据包时停止.每个数据包中的 CSI 以

三维数组的形式展现,每一列依次表示发送天线、接收天线和 30 个子载波.然后把数据包以该参考点的编号存储.实验人员乙根据坐标系向甲提供每个参考点的物理位置,并检查核对数据存储时是否命名正确.

(4) 随机选取 20 个位置当做测试点,实验人员乙确定并记录其真实位置,实验人员甲按照步骤(3)进行实验,采集完毕后依然由实验人员乙核对是否命名正确.

(5) 全部采集完毕后,换回 Windows 系统,把数据导入 MATLAB 中进行滤波等处理.磁场数据数目为 3,从 3 个 AP 的一共 8 根天线中收集 CSI 的幅值,每根天线包含 30 个不同子载波的幅值,一共有 240 个 CSI 幅值,加在一起一共有 243 个数据.

(6) 将输入数据补 0,增加至 256 个数据,通过 MATLAB 里面的 Reshape()函数将数据转化为  $16 \times 16$  的方阵,将这个数据存储为训练样本.

(7) 重复上述步骤,在每个参考点收集 1 000 次训练样本数据.

CNN 网络的参数设置为

(1) 我们第 1 层卷积层的卷积核大小为  $3 \times 3$  的方阵,使用 100 个卷积核,卷积步长设置为 1.(2) 第 2 层池化层的大小为  $2 \times 2$ ,这里的池化层为最大池化层,池化的步长为 1,池化后经过 Relu 函数输出.(3) 第 3 层卷积层的卷积核大小为  $3 \times 3 \times 100$  的三维矩阵,使用 200 个卷积核,卷积步长为 1.(4) 第 4 层的池化层与第 2 层的池化层设置一样.(5) 第 5 层全连接层的大小为 400 个神经元,经过 Relu 函数输出.(6) 第 7 层的输入层大小设置为 243 个神经元.

为了应用于室内定位,收集的 CSI 数据应该排除多路径干扰.在时域中,多径信道  $h(\cdot)$  的总体响应表示为

$$h(\tau) = \sum_{k=0}^{L_p-1} \alpha_k \delta(\tau - \tau_k) \quad (12)$$

其中,  $L_p$  是多径信道分量的数量,  $\alpha_k$  和  $\tau_k$  分别表示第  $k$  个路径的幅度和时间延迟<sup>[17]</sup>.通过对 CSI 信号进行快速傅里叶逆变换(IFFT)转换,我们可以得到时域中的  $h(\tau)$ ,它反映了不同时延下的信号强度,因此可以粗略地区分视距路径和多路径成分.我们设置一个阈值并过滤掉过多的脉冲,以获得近似的视距路径.

由于磁场数据是变化的,我们取每 2 分钟的平均值作为代表数据.

## 3.2 实验结果

### 3.2.1 基于 CNN 指纹定位的定位性能

在本小节中,我们与多层的 BP 神经网络相比较来评估我们提出的基于 CNN 网络的指纹定位算法.我们在相同的环境下,进行了 6 次对比试验,每次实验中测量 10 次定位结果,取 10 次定位结果的误差的平均值作为这次试验的定位误差.实验结果如图 6 所示.

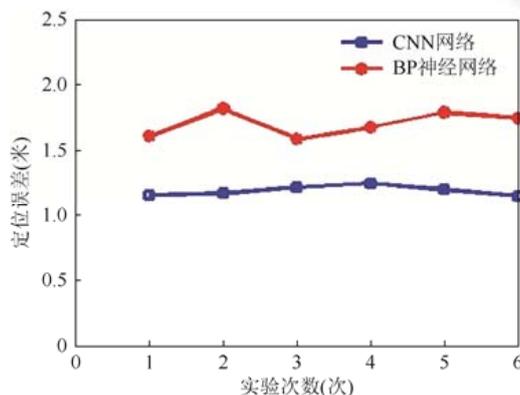


Fig.6 Comparison of localization based on CNN and BP neural network

图 6 基于 CNN 定位与 BP 神经网络定位性能对比

从图6中我们可以发现,伴随着试验次数的增多,我们的提出的CNN指纹定位系统非常稳定,定位的误差只有微小的浮动,保持在1.2m左右.而基于BP网络的定位算法抗干扰性比较差,在第2次试验、第3次试验中其定位误差均有较大幅度的跳变.除了稳定性,在定位精度方面,CNN网络在定位精度上也优于BP神经网络约30%.这是因为CNN网络相对于BP网络而言,可以使用更少和更精准的参数来描述这个数据的特征,减少了测量误差和干扰对定位结果的影响.

### 3.2.2 训练样本数对定位结果的影响

为了研究不同训练样本数对定位结果的影响,我们从总样本库里随机抽取100,300,500,700组样本去训练CNN网络和BP网络的参数,然后保存这些参数值为指纹进行定位.实验结果如图7所示.

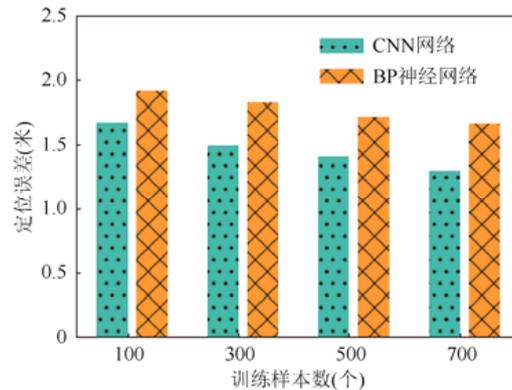


Fig.7 Impact of number of samples on localization based on CNN and BP neural network

图7 样本数量对CNN网络和BP网络性能的影响

显然,从图7中我们可以发现,随着训练样本数量的增加,CNN网络与BP网络的定位精度都不断地增加,并且CNN网络的定位精度一直优于BP网络.在训练样本数量为100的时候,两种方法的定位效果最差,CNN网络和BP网络的定位误差分别为1.7m和1.9m.而在训练样本数量为700的时候,定位效果最好,CNN网络和BP网络的定位精度分别增加了20%和15%.这是因为随着训练样本数量的增加,CNN网络和BP网络从样本中训练得来的权值也越来越准确,但是BP网络由于其结构复杂,容易在训练时引入误差,因此它的整体定位性能比我们的CNN网络的定位性能差.

从上述两个实验的结果可以看出BP神经网络和CNN网络的区别:(1)在BP神经网络中,每一层的神经元和另外一层的神经元之间,是全连接的状态,需要训练参数数量巨大,很容易在参数中引入误差量.而在CNN网络中,由于卷积层和池化层的存在,相隔的两层之间只有相近的神经元之间才有连接,属于局部连接,这使得CNN网络的架构复杂度大大降低,也减少了需要训练的参数数量.(2)与BP网络不同,CNN网络的卷积层和池化层都是特征提取和降维的过程,经过卷积和池化之后,增强了数据表达的特征并削弱了隐藏的噪声信号,使输出结果更加稳定.

### 3.2.3 与传统定位算法的比较

本小节中,我们比较了CNN网络与传统定位算法的定位性能,这里我们选取一种直接匹配的无需学习算法即K最邻近(KNN)算法<sup>[18]</sup>和一种浅学习算法即支持向量机(SVM)算法<sup>[19]</sup>与我们的CNN网络定位相比较.实验结果如图8所示.

这里我们引入了一种新的判断标准,即标准差来衡量定位的性能.标准差指的是多次测量时,测量的结果偏离均值的程度.从图8中可以发现,CNN网络定位结果的均方误差和标准差分别为1.24m,0.58m,优于SVM的1.60m,1.02m和KNN的2.2m,1.8m.这是因为KNN是一种无学习的算法,无法利用原始数据的特征,在定位时仅仅利用了表面的、带有误差的信息,所以定位的结果比较差.SVM是一种浅学习方法,尽管它可以利用数据的一

些特征,但是它对于复杂数据的表示能力有限,对复杂的非线性问题无法得出最优解.而 CNN 网络作为一种深度学习网络,通过多层参数对数据进行学习,可以表示数据中更为深层次的特征,因此利用它进行定位性能优于 SVM 和 KNN 算法.

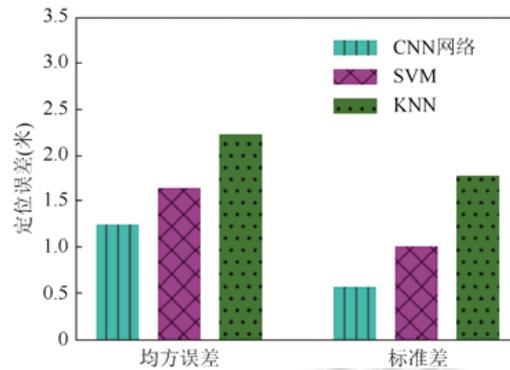


Fig.8 Comparison of localization algorithms based on CNN, SVM and KNN

图8 CNN、SVM 和 KNN 定位算法的比较

#### 4 总 结

随着通信技术和移动设备的普及,物联网和 LBS 在人们的生活中扮演着越来越重要角色,从而推动着室内定位进入了飞速发展期.本文首先介绍了室内定位的几种常见的方法,然后介绍了基于 WiFi 和地磁场的室内定位方法,并针对这两种方法中 RSS 和磁场强度各自存在的缺陷和 CSI 的优点提出了一种改进算法.本文使用深度学习神经网络即 CNN 网络来处理采样数据,通过将多层 CNN 网络的参数值存储为指纹,提高了对原始采样数据利用率的同时,也极大的抑制了原始数据中的噪声信号.最后我们通过贝叶斯概率法得出目标的估计位置.实验结果表明,我们提出的 CNN 指纹定位算法,在定位精度和稳定性上都优于单层的神经网络算法和无学习的算法.

#### References:

- [1] Li JL. BDS/GPS multi-frequency real-time kinematic positioning theory and algorithms. *Acta Geodaetica et Cartographica Sinica*, 2015,44(11):1297 (in Chinese).
- [2] Li T, Sun JQ, Wang ZL. Design of a locatable intelligent car based on Android. *Journal of Yancheng Institute of Technology (Natural Science Edition)*, 2017,30(4):41-46 (in Chinese with English abstract).
- [3] Yang GJ. Research on RFID localization algorithm [MS. Thesis]. Luoyang: He'nan University of Science and Technology, 2011 (in Chinese with English abstract).
- [4] Wang YJ. Research and implementation on key technologies of bluetooth indoor positioning [MS. Thesis]. Nanjing: Southeast University, 2015 (in Chinese with English abstract).
- [5] Jarle B. ZigBee positioning solution. *Electronic Design & Application World—Nikkei Electronics China*, 2008,(1):85-87 (in Chinese).
- [6] Gao ZJ. Design of Kalman filter for inertial positioning system. *Journal of Chinese Inertial Technology*, 2000,8(4):5-9 (in Chinese with English abstract).
- [7] Moreno M V, Skarmeta A F. An indoor localization system based on 3D magnetic fingerprints for smart buildings. In: *Proc. of the IEEE RIVF Int'l Conf. on Computing & Communication Technologies—Research, Innovation, and Vision for the Future*. IEEE, 2015. 186-191.
- [8] Wang Q, Luo H, Zhao F, *et al.* An indoor self-localization algorithm using the calibration of the online magnetic fingerprints and indoor landmarks. In: *Proc. of the Int'l Conf. on Indoor Positioning and Indoor Navigation*. IEEE, 2016. 1-8.

- [9] Zhang M, Shen W, Zhu J. WIFI and magnetic fingerprint positioning algorithm based on KDA-KNN. In: Proc. of the Chinese Control and Decision Conf. IEEE, 2016. 5409–5415.
- [10] Ezani MFM, Abdullah MA, Haseeb S. A region-to-point indoor localization approach via RSS-magnetic fingerprinting. In: Proc. of the Int'l Conf. on Information and Communication Technology for the Muslim World. IEEE, 2015. 253–258.
- [11] Ban R, Kaji K, Hiroi K, *et al.* Indoor positioning method integrating pedestrian dead reckoning with magnetic field and WiFi fingerprints. In: Proc. of the 8th Int'l Conf. on Mobile Computing and Ubiquitous Networking. IEEE, 2015. 167–172.
- [12] Wu K, Xiao J, Yi Y, *et al.* CSI-Based indoor localization. IEEE Trans. on Parallel & Distributed Systems, 2013,24(7):1300–1309.
- [13] Wu K, Xiao J, Yi Y, *et al.* FILA: Fine-Grained indoor localization. Proc. of the IEEE INFOCOM, 2012,131(5):2210–2218.
- [14] Chapre Y, Ignjatovic A, Seneviratne A, *et al.* CSI-MIMO: Indoor Wi-Fi fingerprinting system. In: Proc. of the 2014 IEEE 39th Conf. on Local Computer Networks (LCN). IEEE, 2014. 202–209.
- [15] Li YD, Hao ZB, Lei H. A review of convolutional neural network research. Journal of Computer Applications, 2016,36(9): 2508–2515 (in Chinese with English abstract).
- [16] Wang X, Gao L, Mao S. CSI phase fingerprinting for indoor localization with a deep learning approach. IEEE Internet of Things Journal, 2017,3(6):1113–1123.
- [17] Yang Z, Zhou Z, Liu Y. From RSSI to CSI: Indoor localization via channel response. ACM Computing Surveys, 2013,46(2):1–32.
- [18] Jiang Q, Li K, Zhou M, *et al.* Competitive agglomeration based KNN in indoor WLAN localization environment. In: Proc. of the Int'l Conf. on Communications and Networking in China. IEEE, 2016. 338–342.
- [19] Zhou R, Lu X, Zhao P, *et al.* Device-Free presence detection and localization with SVM and CSI fingerprinting. IEEE Sensors Journal, 2017,17(23):7990–7999.

#### 附中文参考文献:

- [1] 李金龙.北斗/GPS 多频实时精密定位理论与算法.测绘学报,2015,44(11):1297.
- [2] 田莉,孙金秋,王志凌.基于安卓系统的定位智能小车设计.盐城工学院学报(自然科学版),2017,30(4):41–46.
- [3] 杨公建.RFID 定位算法研究[硕士学位论文].洛阳:河南科技大学,2011.
- [4] 王益健.蓝牙室内定位关键技术的研究与实现[硕士学位论文].南京:东南大学,2015.
- [5] Boe J.ZigBee 定位解决方案.电子设计应用,2008,(1):85–87.
- [6] 高钟毓.惯性定位系统的卡尔曼滤波器设计.中国惯性技术学报,2000,8(4):5–9.
- [15] 李彦冬,郝宗波,雷航.卷积神经网络研究综述.计算机应用,2016,36(9):2508–2515.



王英(1980—),女,四川简阳人,讲师,CCF 专业会员,主要研究领域为室内定位,边缘计算,语义本体.



黄旭东(1992—),男,硕士,主要研究领域为室内定位.



郭松涛(1975—),男,博士,教授,博士生导师,CCF 专业会员,主要研究领域为室内定位,无线传感器网络,移动边缘计算,数据中心网络.