文章编号:1671-4598(2019)08-0217-05 DOI:10.16526/j.cnki.11—4762/tp.2019.08.045 中图分类号:TP3 文献标识码:A

基于深度强化学习的移动机器人导航策略研究

江其洲，曾 碧
（广东工业大学 计算机学院，广州 510006）

摘要：针对移动机器人在复杂动态变化的环境下导航的局限性，采用了一种将深度学习和强化学习结合起来的深度强化学习方法；研究以在OpenCV平台下搭建的仿真环境的图像作为输入数据，输入至TensorFlow创建的卷积神经网络模型中处理，提取其中的机器人的动作状态信息，结合强化学习的决策能力求出最佳导航策略；仿真实验结果表明：在经过深度强化学习的方法训练后，移动机器人在环境发生了部分场景变化时，依然能够实现随机起点到随机终点的高效准确的导航。

关键词：深度强化学习；移动机器人；导航；复杂环境

Research on Navigation Strategy of Mobile Robot Based on Deep Reinforcement Learning

Jiang Qizhou, Zeng Bi
(School of Computers, Guangdong University of Technology, Guangzhou 510006, China)

Abstract: Aiming at the limitation of mobile robot navigation in complex and dynamic environment, a deep reinforcement learning method combining deep learning and reinforcement learning is adopted. In this study, the image of the simulation environment built under the OpenCV platform was used as input data, which was input into the convolutional neural network model created by TensorFlow for processing, in which the robot's action state information was extracted, and the optimal navigation strategy was obtained by combining the decision-making ability with reinforcement learning. The simulation results show that after the training with the method of deep reinforcement learning, the mobile robot can still realize the efficient and accurate navigation from the random starting point to the random ending point when some scenes change in the environment.

Keywords: deep reinforcement learning method; mobile robot; navigation; complex environment

0 引言

移动机器人的研究起源月上世纪60年代末，最初是用来在恶劣、危险的条件下或者复杂环境中来代替人类完成工作。移动机器人技术处于当前科技研究的前沿，代表着当代高新技术的发展方向，是各国竞相研究发展的重点，是当前科学研究的热点之一。随着计算机技术、传感技术、网络技术和通信技术的飞速发展，移动机器人技术也得到了更加深入的而广泛的研究。

现在移动机器人的研究重点逐渐向智能化发展，如何让机器人体现人工智能是目前移动机器人的研究难点[1-2]。移动机器人的智能化即是实现其高度的自主性，能够使机器人在没有人的引导下，无需对环境进行特殊的限制和改变的情况下，能够有目的地、准确的完成任务，这需要机器人具备环境感知、行为决策、动作控制等能力。在移动机器人的智能化的研究中，导航技术的保障是其研究的核心，也是其实现智能化以及完全自主的关键技术和前提。

移动机器人的导航是指“基于移动机器人自身携带的传感器感知的周围的环境信息以及移动机器人的自身状态信息，在包含有限数量障碍物的环境中，安全地实现移动机器人面向目标的运动”。随着现在机器人应用越来越广泛，应用领域不断拓展，机器人需要完成的任务也越来越复杂，现阶段的大部分机器人在确定的、静态的、单一环境中执行导航任务，可以通过技术人员对机器人固定的导航任务人为的预先编程来实现，但这样的机器人往往不具备应变突发事件的能力。对于场景的动态变化、机器人的“绑架”等问题，设计人员难以对机器人遇到的问题作出合理的预测以及预设相应决策，都不能得到有效的解决。

不论是传统的机器人导航控制方法还是针对特定任务的预处理，要解决机器人应对突发事件的处理必须具备比较强的对周围环境信息感知和分析能力以及之后的动作执行能力。由此，基于强化学习（Reinforcement Learning，RL）的机器人导航成为国内外学者对于该领域的研究热点。基于强化学习的导航优势在于：模型简单、算法编程简易、鲁棒性强。但是传统的强化学习方法由于环境的多样性和复杂性，存在学习时间长、收敛速度慢、机器人状态信息提取困难等问题。近年来深度学习的研究进展能够有效的

收稿日期:2019-03-05； **修回日期**:2019-03-26。
基金项目:国家自然科学基金(61701122);广东省产学研重大专项项目(2016B010108004);广州市重点科技项目(201604020016);广东省产学研专项(2014B090904080)。
作者简介:江其洲(1994-),男,江西宜春人,硕士,主要从事深度强化学习方向的研究。
曾 碧(1963-),女,广东广州人,博士,教授,主要从事智能信息处理,智能机器人方向的研究。

弥补强化学习的劣势，谷歌的人工智能研究团队 DeepMind 创新地将具有感知能力的深度学习（Deep Learning，DL）相结合，开创了一个新的研究热点，即深度强化学习（Deep Reinforcement Learning，DRL）[3]，因此本文使用基于 DRL 的研究策略实现机器人在复杂环境下的导航。基于 DRL 的导航策略研究采用端对端的学习方式，利用经验回放机制，将包含机器人感知到的周围环境信息、当前所处的状态以及动作产生反馈的图像信息存储到经验回放池中，再定期每一个时间步从经验回放池随机提取一组参数作为输入传递到卷积神经网络中来不断的迭代更新网络参数，最终求取网络参数的最大值，即为一次导航的最优策略。

1 相关研究

1.1 卷积神经网络

卷积神经网络（convolutional neural network，CNN[4]）从本质上来说是一个前向的反馈神经网络，来源于生物视觉神经结构启发，是以最简化预操作为目的的多层感知器的变形。CNN 提供了一种端对端的学习模型，通过把图像作为参数输入到模型中，使用传统的梯度下降的方法对其进行训练，经过训练后的 CNN 网络能够学习图像中的特征，最终完成对图像特征的提取，所提取到的特征具有平移，旋转不变性[5]等特性。近年来，CNN 被很好的应用在了强化学习的任务上，如 Atari 游戏，机器操纵和模仿学习等方面。

卷积神经网络主要包括 4 个方面的技术：1）局部感知域，当需要训练的参数过多时，全连接网络训练难度极大，极难收敛。因此 CNN 与人类视觉类似采用局部感知信息，低层次神经元感知局部信息，高层次神经元整合低层次神经元感知的局部信息得到全局信息，由此大大降低了训练参数的量级；2）参数共享，利用对图像顺序的进行卷积的方式提取图像的某种特征，将多个具有相同统计特征的参数统一，进而进一步降低训练参数的量级；3）多卷积核，对图像进行的一个卷积便是一种提取方式，通常在对一幅图像来说，单个卷积核提取的特征是远远不够的，因此使用多重卷积核才能提取多种不同的特征；4）池化，解决使用特征图训练分类器时可能产生的特征维度过多计算复杂、过拟合等问题。近年来卷积神经网络已经成功应用于人脸识别、字符识别、行为检测和目标检测等方面。

1.2 强化学习

强化学习[6]的基本原理是利用自身与周围环境的即时交互产生的反馈信号来对所采取的行动进行评价，如果反馈信号越强，代表环境对这个动作的正奖励，则这个动作的趋势便加强；反之，这个动作的产生趋势就减弱。强化学习的本质上就是个不断试错来逐步改进策略的过程，目的就是学习一个行为策略来获得环境最大的奖励。

强化学习的基本模型如图 1 所示，智能体 agent 采取一个动作 a 作用到环境中，环境接收到这个动作后，产生一个奖励 r 反馈给 agent，agent 再根据反馈回来的奖励 r 和当前的环境状态信息 e 来选择下一个动作，如此循环往复，不断改进策略。

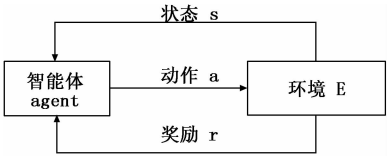

图 1 强化学习基本模型

1.2.1 马尔科夫模型与贝尔曼方程

强化学习的目的可以转化为求解马尔科夫决策过程（markov decision process，MDP）的最优策略，MDP 的本质是：下一状态的概率和奖励值由且仅由当前状态和动作决定，与其他任何历史状态和历史动作无关。

其公式表述为式（1）所示，r 表示奖励，γ_* 表示折扣因子：

$$\sum_{i=0}^{\infty}\gamma^i r_{t+i}0\leqslant\gamma\leqslant 1 \tag{1}$$

用价值函数 v 表示 MDP 求解的值，价值函数模型如式（2）、（3）所示：

$$v^{\pi}(s)=E_{\pi}\Big[\sum_{t=0}^{\infty}\gamma^i r_i\,|\,s_0=s\Big] \tag{2}$$

$$\begin{aligned}v^{\pi}(s)&=E_{\pi}[r_0+\gamma r_1+\gamma^2 r_2+\gamma^3 r_3+\cdots\,|\,s_0=s]=\\&E_{\pi}[r_0+\gamma E[\gamma r_1+\gamma^2 r_2+\gamma^3 r_3+\cdots]\,|\,s_0=s]=\\&E_{\pi}[r(s'\,|\,s,a)+\gamma V^{\pi}(s')\,|\,s_0=s]\end{aligned} \tag{3}$$

其中：γ 表示折扣系数，代表后续动作对当前值的影响程度。其取值范围是［0，1］，0 表示只考虑当前动作，不考虑后续动作的影响，而 1 表示当前动作和后续每步动作都有均等的影响。通常为了避免使问题陷入局部最优，随着步数的增加，折扣系数应当减小，影响变小。使用贝尔曼方程来求解价值函数。求解过程如式（4）、（5）所示：

$$\begin{aligned}v^{\pi}(s)&=\sum_{s'\in S}p(s'\,|\,s,\pi(s))[r(s'\,|\,s,\pi(s))+\gamma V^{\pi}(s')]=\\&E_{\pi}[r(s'\,|\,s,a)+\gamma V^{\pi}(s')\,|\,s_0=s]\end{aligned} \tag{4}$$

$$\begin{aligned}Q^{\pi}(s,a)&=\sum_{s'\in S}p(s'\,|\,s,a)[r(s'\,|\,s,a)+\gamma V^{\pi}(s')]=\\&E_{\pi}[r(s'\,|\,s,a)+\gamma V^{\pi}(s')\,|\,s_0=s,a_0=a]\end{aligned} \tag{5}$$

在式（4）中，π 表示当前的策略，$Q^{\pi}(s,a)$ 是针对实际问题在 $v^{\pi}(s)$ 基础上引入的动作值 a，$Q^{\pi}(s,a)$ 表示动作值函数，式（5）表示动作值函数模型。对贝尔曼方程求解最优解得到贝尔曼最优方程（6）、（7）为：

$$V^*(s)=\max_a E[r(s'\,|\,s,a)+\gamma V^*(s')\,|\,s_0=s] \tag{6}$$

$$\begin{aligned}Q^*(s)&=E[r(s'\,|\,s,a)+\gamma\max_{a'}Q^*(s',a')\,|\,s_0=s,a_0=a]=\\&\sum p(s'\,|\,s,\pi(s))[r(s'\,|\,s,\pi(s))+\gamma\max_{a\in A(s)}Q^*(s',a')]\end{aligned} \tag{7}$$

求解上述贝尔曼最优方程（6）、（7）有两种方法：策略迭代和价值迭代。

1.2.2 策略迭代

策略迭代共有两个步骤：策略评估和策略改进，首先

对已有的策略进行评估，获得状态值函数，然后根据评估结果，如果新策略更好则取代之前策略，否则，保持原有策略。具体算法流程如下所示：

1）策略评估

Input π（输入策略 π）

Initialize an array $v(s)=0$，for all $s\in\delta^{+}$

Repeat

$\Delta\leftarrow 0$

For each $S\in\delta$：

$temp\leftarrow v(s)$

$v(s)\leftarrow\sum_{a}\pi(a\mid s)\sum_{s'}p(s'\mid s,a)[r(s,a,s')+\gamma v(s')]$

$\Delta\leftarrow\max(\Delta,\mid temp-v(s)\mid)$

Until $\Delta<\theta$（a smalll positive number）

Output $v\approx v_{\pi}$

2）策略迭代

policy－stable ← true

For each $s\in\delta$：

$temp\leftarrow\pi(s)$

$\pi(s)\leftarrow\operatorname{argmax}_{a}\sum_{s'}p(s'\mid s,a)[r(s,a,s')+\gamma v(s')]$

If $temp\neq\pi(s)$，then policy－stable ← false

If policy－stable，then stop and return v and π

Else go to evalue policy

1.2.3　值迭代

值迭代使用贝尔曼最优方程来更新 value，经过反复迭代使得最终的 value 收敛于 V^{π}，即在当前状态下最优值为 value 时，该最优值 value 对应的策略即为最优策略。其算法流程如下：

Initialize array v arbitrarily（e.g.，$v(s)=0$ for all $s\in\delta^{+}$）

Repeat

$\Delta\leftarrow 0$

For each $s\in\delta$

temp ← $v(s)$

$v(s)\leftarrow\max_{a}\sum_{s'}p(s'\mid s,a)[r(s,a,s')+\gamma v(s')]$

$\Delta\leftarrow\max(\Delta,\mid temp-v(s)\mid)$

Until $\Delta<\theta$（a small positive number）

Output a determinisitc policy π，such like

$\pi(s)=\operatorname{argmax}_{a}\sum_{s'}p(s'\mid s,a)[r(s,a,s')+\gamma v(s')]$

1.3　深度强化学习

在高级人工智能领域，智能体感知和决策能力是衡量智能体智能化的关键性指标。强化学习虽然具有优秀的决策能力，但是其应用大部分均依赖于人工提取特征，难以处理高维度状态空间下的问题。而深度学习具有优秀的感知能力，能够从高维原始数据提取特征。这两者优势互补、结合成深度强化学习。目前 DRL 技术在游戏[7-8]，机器人控制[9-10]，参数优化[11]和机器视觉[12]等领域均有广泛的应用。

1.3.1　基于值函数

基于值函数的深度强化学习最典型的的代表就是 Mnih[7]等人将 CNN 与 Q 学习算法[14-15]结合提出的深度 Q 网络（Deep Q－network，DQN）模型。其基本原理就是将 Q 学习神经网络化，利用深度卷积神经网络不断迭代更新值函数的优化目标，即目标 Q 值，从而得到最优的学习策略。

1.3.2　基于策略梯度

基于值函数的深度强化学习主要用于解决在离散动作空间下的任务，对于连续动作空间的任务采用基于策略梯度的深度强化学习算法可以或得更好的决策效果。策略梯度通过不断计算策略的总奖励期望值关于策略参数的梯度来更新参数，得到最优策略[13]。其优势在于：直接优化策略的总奖励期望，以端对端的方式直接在策略空间里搜索最优策略，比基于 DQN 的模型适用范围更广泛，优化效果也更好。

2　基于 DQN 的移动机器人导航策略研究

本文将 DQN 网络、经验回放机制、搜索与利用平衡策略[16]以及随机梯度下降法等方法结合应用到机器人导航研究中，提出一种基于深度强化学习的移动机器人导航策略的研究方法。通过 OpenCV 仿真平台的检验，验证本文提出的算法能够高效准确的完成导航任务。

2.1　DQN 网络参数预处理

OpenCV 仿真平台生成的地图原始图像是 RGB 图像，有 3 个通道。直接将其输入网络计算量较大。因此本文采用了基本的图像预处理来降低输入维度，通过将图像等比例缩放至大小为 80×80，然后利用二值法将其转换为只有两个通道的灰度图像，这样可以降低输入参数一个维度和数据量，有利于之后网络的特征提取和处理。

2.2　模型结构与图像处理过程

本文采用的网络模型是 2015 年，由 DeepMind 提出的深度 Q 网络（deep Q network，DQN）[3]，DQN 的输入是经过预处理后当前时刻连续的 4 幅图像。经过 3 个卷积层和两个全连接层的处理后，最终输出动作的 Q 值。图 2 表示 DQN 的模型结构。

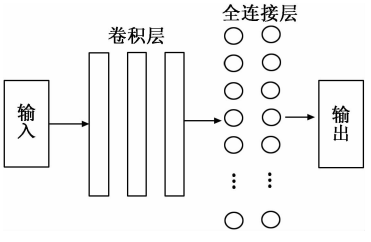

图 2　DQN 网络模型结构

图 3 描述了本文采用的 DQN 网络模型对图像进行处理的具体过程。

1）将经过预处理后的连续四幅图像 80×80×4（4 表示

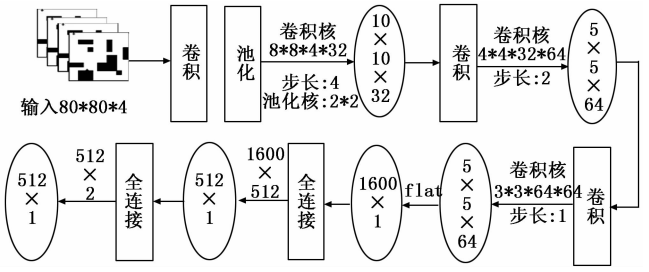

图 3 图像处理过程

4 个通道，四张图即是 4 个通道）经过卷积核为 8×8×4×32，步长为 4 的卷积，得到 32 张大小为 20×20 的特征图，即 20×20×32。将其进行池化核为 2×2 的池化得到 10×10 的图像，即此时为 10×10×32；

2）将上一步所得图像进行卷积核为 4×4×32×64，步长为 2 的卷积得到 64 张 5×5 的图像，即 5×5×64；

3）再进行一次卷积核为 3×3×64×64，步长为 1 的卷积，此时依旧得到 5×5×64 的图像，但此时经过了再一轮卷积的图像，其图像信息更加抽象，更具全局性；

4）对第二次卷积后 5×5×64 的图像进行 1 600×512 的全连接，得到一个 512 维的特征向量，即 512×1；

5）再次进行全连接，最终输出二位向量［0，1］和［1，0］，表示仿真实验中的正反馈和负反馈。

2.3 DQN 算法训练流程

DQN 算法是在传统 q 学习算法的基础上将其神经网络化实现的。传统 q 学习是最早的在线学习算法，是基于值迭代的具有代表性的强化学习算法。图 4 描述了 DQN 算法的训练流程。

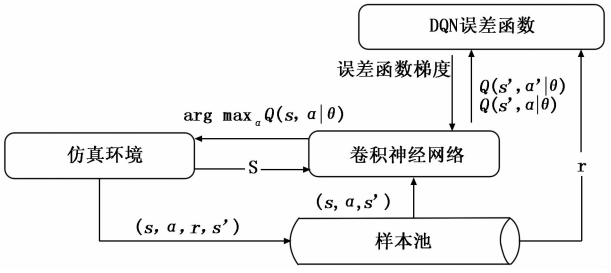

图 4 DQN 训练流程图

具体算法流程为：

1）初始化样本池 D，容量为 N；

2）将卷积神经网络进行随机权重初始化，得到初始 Q 函数；

3）进入循环 A，$i=1$，…，M；

4）选择初始状态，对仿真环境图像进行预处理；

5）进入循环 B，$t=1$，…，T；

6）采用随机策略 ε 选择一个动作 a_t；

7）执行动作 a_t，得到奖励 r_t 和下一时刻仿真环境图像 x_{t+1}；

8）令 $S_{t+1}=S_t,a_t,x_{t+1}$，对状态进行预处理 $\varphi_{t+1}=\varphi(s_{t+1})$；

9）将 $(\varphi_t,a_t,r_t,\varphi_{t+1})$ 存储到样本池 D 中；

10）从样本池 D 中随机采集 m 个训练样本 $(\varphi_j,a_j,r_j,\varphi_{j+1})$；

11）令样本标签值为：

$$Y_j=\begin{cases}r_j,\varphi_{j+1}\text{ 为终止状态}\\ r_j,r\max_{a'}\varphi(\varphi_{j+1}a';\theta),\varphi_{j+1}\text{ 不为终止状态}\end{cases}$$

12）用梯度下降法更新网络参数，计算损失函数；

13）退出循环 B；

14）退出循环 A。

DQN 算法在传统的 Q 学习算法上进行了改进，采用经验回放机制和固定目标网络两个关键技术来提升算法的稳定性。

经验回放机制：经验回放最初是由 Linux 在其博士论文中提出[17]，其原理是将训练过程中的样本依次存储在样本池中，训练时再从中随机抽取一定量的样本，使用随机梯度下降法（SGD）更新网络参数。经验回放机制的使用，对历史数据也能进行重复采样，提高了数据的使用效率，同时也打破了样本间的关联，使样本间相互独立，提升的算法的稳定性。

固定目标网络：将 q 网络迭代优化的目标 Q 值采用时序差分法由另一个单独的较慢的目标网络产生，这样提高了算法的收敛性。

DQN 算法的主要特点有 3 个：

1）是一种端到端的训练方法，以原始图像和奖励函数作为的输入和每个动作和对应 Q 值的输出相映射；

2）使用经验回放机制和固定目标网络提升整个训练过程的稳定性和收敛性；

3）可以再不同的仿真平台中采用大致相同的网络结构和训练算法，仅需根据训练情况调整相应的奖励函数。

2.4 搜索与利用平衡策略

在 2.3 节中的 DQN 算法中步骤（6）使用了一个策略来生成移动机器人的下一步动作，这个策略并不是求解优化过程得到的策略，是单独用来生成机器人动作的策略。因此，本文所用的 Q－learning 算法属于 off－policy，整个 DQN 的算法流程也是无模型的，只考虑当前的环境信息和奖励函数 reward 的反馈，即 model－free 的方法。一般来说，使用策略生成机器人动作主要有两种策略：

1）greedy policy，即贪心策略，让机器人尽可能朝奖励函数大的方向行进，当机器人执行一个动作如果得到的奖励是正的，积极的则下一次继续朝该方向行进，反之，则朝其他方向前进。

2）randomized policy，即随机策略，不考虑机器人执行动作后得到的反馈，每次都均等的随机选取一个动作执行。

考虑到 greedy policy 容易导致过拟合的现象，使得机器人导航的策略陷入局部最优，只能执行单一或少数情况

下的导航，不具备良好的泛化能力。因此，本文采用 randomized policy 来随机生成机器人的动作，相应的也增加了一定的训练时间来保证良好实验的效果。

3　实验仿真与分析

3.1　实验平台描述

本文仿真实验使用的平台如表 1 所示。

表 1　仿真使用平台

名称	版本型号
CPU	Conroe i7
RAM	8G
GPU	NVIDIA 1070
系统	Ubuntu 16.04
OpenCV	3.3.1
TensorFlow－gpu	1.0.0

使用 OpenCV 构建的移动机器人仿真实验环境如图 5 所示，仿真环境是由一个 800 * 800 像素大小的图像构成，其中黑色边框代表围墙，黑色矩形代表障碍物，圆点表示出发点，方块表示导航的目的地，起点和终点均是随机出现在地图中非障碍物的地方。

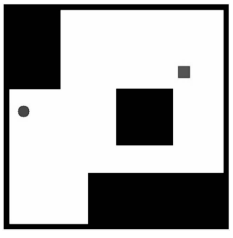

图 5　移动机器人仿真环境

3.2　实验结果

3.2.1　初始地图下的导航

图 6（a）和图 6（b）分别展示地图环境未发生变化时，两次机器人从随机起点到随机终点的顺利导航。如图中所示，机器人有上下左右 4 个方向维度的动作，每次 5 个像素点移动一次。

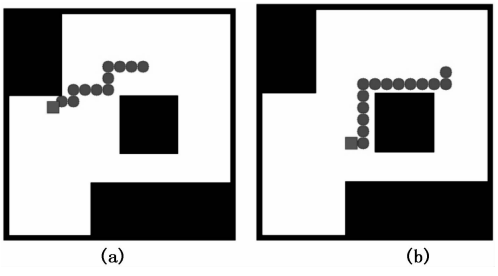

图 6　初始地图下的导航

3.2.2　增量环境下的导航

图 7（a）和图 7（b）展示了增量环境下，即地图中障碍物增加的情况下，移动机器人也能够顺利完成导航任务，并且在此情况下，依旧采用的是之前训练好的模型，相同的网络结构，相同的参数。

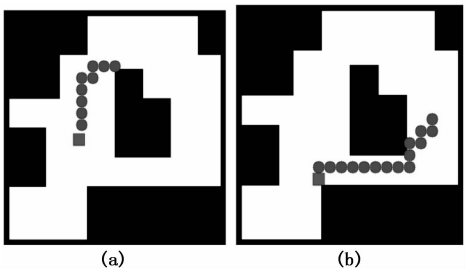

图 7　增量环境下的导航

4　结束语

本文针对复杂动态变化的室内环境下采用了区别于以往 A^* 算法等的只能解决固定两点位置间的路径规划的算法，使用当下人工智能最新的研究领域深度强化学习，将其运用在机器人导航策略的研究上，有效的解决了在室内环境中，场景的发生改变的情况下也能完成移动机器人从任意一个位置到任意另一个位置的导航。不过该研究方法也有一些难点：

1）样本利用率低，需要大量实验迭代次数才能达到较好的结果，因此导致训练所需时间较长；

2）奖励函数较难设置，需要根据使用的实际平台训练时的实验结果进行细微调整；

3）过拟合严重，场景发生较大改变时实验结果不太理想，需要重新训练；

4）导航成功率需待提高，当环境信息较为复杂是，移动机器人的导航难以保持比较高的准确性。

虽然深度强化学习应用于机器人导航策略研究有以上难点，但是随着硬件性能的逐步提升大量的实验训练次数的需求将不是问题，样本利用率低的问题也能得到有效的解决，此外越来越多的学者对于奖励函数的设置和更优的训练模型展开了研究，因此利用深度强化学习进行机器人导航策略的研究一定会是今后一个研究的热点，能成为满足人们对移动机器人智能化的要求的有利手段。

参考文献：

[1] Ohnishi N，Imiya A Appearance－based navigation and homing for autonomous mobile robot [J]. Image and Vision Computing. 2013，31 (6/7)：511－532.

[2] Colle E，Galerne S. Mobile robot localization by multiangulation using set inversion [J]. Robotics and Autonomous Systems，2013，61 (1)：39－48.

[3] Mnih V，Kavukcuoglu K，Silver D，et al. Playing atari with deep reinforcement learning. [A] Proceedings of Workshops at the 26th Neural Information Processing Systems 2013 [C]. Lake Tahoe，USA，2013：201－220.

（下转第 226 页）