DOI:10.19651/j.cnki.emt.2416502

基于 GRU-A3C 的四旋翼无人机视觉避障系统*

马澳华 邢关生
(青岛科技大学自动化与电子工程学院 青岛 266061)

摘 要:针对基于深度强化学习的四旋翼无人机视觉避障系统,模型训练速度慢、计算量大和响应不及时的问题,设计了一种轻量化且模型训练速度快的系统。该系统首先以深度图像和无人机自身状态信息作为输入,然后使用一种基于 GRU 结构的 A3C 算法(GRU-A3C),输出连续动作空间并结合课程学习的方法进行训练加速。最后,以 A3C 为基线进行消融实验。实验结果为:在训练 1 000 轮次时,利用课程学习方法训练的 GRU-A3C 算法成功率为 0.28,A3C 算法成功率为 0.2;在训练 5 000 轮次时,利用课程学习方法训练的 GRU-A3C 算法成功率为 0.72,A3C 算法成功率 0.62。数据表明,该系统可以有效加快模型收敛速度,缩短训练时间并提高训练效果。

关键词:深度强化学习;四旋翼无人机;A3C;课程学习;视觉避障
中图分类号:TP242;TN98 **文献标识码**:A **国家标准学科分类代码**:520.60

Visual obstacle avoidance system for quadrotor UAVbased on GRU-A3C

Ma Aohua Xing Guansheng
(School of Automation and Electronic Engineering, Qingdao University of Science and Technology, Qingdao 266061, China)

Abstract: Aiming at the problems of slow model training speed, large amount of computation and untimely response of quadrotor UAV vision obstacle avoidance system based on deep reinforcement learning, a lightweight and fast model training system is designed. The system first takes the depth image and the UAV′s own state information as input, and then uses a GRU structure-based A3C algorithm (GRU-A3C) to output continuous action space and combine the curriculum learning method for training acceleration. Finally, A3C was used as the baseline for ablation experiments. The experimental results are as follows: after 1 000 rounds of training, the success rate of GRU-A3C algorithm trained using curriculum learning method is 0.28, and the success rate of A3C algorithm is 0.2. After 5 000 rounds of training, the success rate of GRU-A3C algorithm trained using curriculum learning method was 0.72, and the success rate of A3C algorithm was 0.62. The data show that this system can effectively accelerate the model convergence speed, shorten the training time and improve the training effect.

Keywords: deep reinforcement learning; quadrotor UAV; A3C; curriculum learning; visual obstacle avoidance

0 引 言

近年来,无人机(unmanned aerial vehicle, UAV)广泛应用于货物运输、交通管理、航空摄影、环境污染检测等领域[1],自主避障成为无人机领域的迫切需求。由于机器学习的发展,目前已经出现很多基于深度学习和深度强化学习(deep reinforcement learning, DRL)的 UAV 避障方法。基于深度学习的方法虽然减少了为避障任务设计规则或约束的工作量,但这些方法严重依赖于人工制作庞大的数据集和高质量的标签。而基于 DRL 的方法可以让 UAV 在未知环境中自由探索获取学习数据来解决创建数据集问题[2]。因此,基于 DRL 的方法被广泛应用于 UAV 避障任务中。

Mnih 等[3]在 2013 年首次提出 DRL 的概念。研究提出的深度 Q 网络(deep Q network, DQN)可以仅根据图像输入指示,学习 Atari 2600 游戏的操作,其水平超过人类玩家。自此,研究人员提出许多使用 DRL 算法处理自主避障任务的方法,其具有无需地图、学习能力强、对传感器精度依赖性低等优点[4]。2016 年以来,DRL 应用于移动机器人避障的趋势有所增加[5]。

UAV 使用深度相机作为传感器,利用深度图像作为决策过程的状态输入,可以学习深度信息中复杂的空间特征关系。文献[6]所设计的系统以深度相机采集到的深度

收稿日期:2024-07-23
* 基金项目:国家自然科学基金(61503118,62006135)项目资助

图像作为 DRL 输入,分别测试了 DQN、PPO(proximal policy optimization)和 SAC(soft actor-critic)算法在离散和连续动作空间中 UAV 的自主避障能力。该类系统输入图像为深度图像,为 DRL 模型训练提供了丰富的数据,使模型在避障训练时达到较高的精度,但这往往会减慢模型训练速度。文献[7]设计了一种结合目标检测的 UAV 避障系统,该系统将深度相机采集到的深度图像作为目标检测模型的输入,根据检测到的障碍物来修改深度图像,UAV 在障碍物远离图像中心时获得奖励。该系统使用 D3QN (dueling double deep Q network)作为 DRL 算法,采取离散动作空间,实现了固定高度下的 UAV 自主避障。此类系统依赖于另一个模型来辅助 DRL 进行决策,虽较使用原始图像能更快进行收敛,但需处理的计算量较大,严重依赖检测模型的准确性,若遇到未训练过的障碍物可能无法识别,导致任务失败。

文献[8]提出一种系统,该系统使用预训练的 Struct2depth 模型从单目相机图像中估计深度数据,再将深度数据转换为 2D 距离数据,并在仿真环境中使用 DDQN(double deep Q network)训练无人机。文献[9]使用 FCRN(fully connected residual network)网络从 RGB 图像输入估计深度信息,并将其作为 D3QN 模型输入,在 ROS 和 Gazebo 仿真环境中进行训练。为了验证所提方法的性能和鲁棒性,使用 Parrot Bebop2 无人机在各种复杂的室内环境中进行仿真和真实实验。该类系统仅使用单目摄像机即可实现局部环境下的避障,减少了 UAV 避障任务的硬件要求,适用于无法安装深度相机的小型 UAV,但其较使用原始深度图像的方法增添了多层网络,增加了计算需求和计算时间,一定程度上影响 UAV 及时做出决策。

以上基于视觉的 UAV 自主避障系统存在模型训练速度慢、计算量大和响应不及时的问题。针对此类系统的弊端,本文设计了一种轻量化且模型训练速度快的系统。该系统使用深度图像和无人机自身状态信息作为输入,为使 UAV 路径平滑,采用连续动作空间,使用一种融合 RNN 与 DRL 的算法——基于门控循环单元(gated recurrent unit, GRU)[10]结构的 A3C(asynchronous advantage actor-critic)[11]算法(GRU-A3C)。在模型训练时,结合课程学习(curriculum learning, CL)[12]的方法进行训练加速。经过消融实验验证,该系统可以有效加快模型收敛速度,缩短训练时间并提高训练效果。

1 基于 DRL 的 UAV 避障系统设计

1.1 避障控制系统结构

基于 DRL 的 UAV 避障系统主要包含 3 个部分:DRL、UAV 和环境,图 1 为避障控制系统结构图。UAV 的传感器与环境交互后,将当前状态(状态空间)作为 DRL 的输入,经过网络运算输出一组动作,随后将动作指令发布给 UAV,UAV 在环境中执行动作,执行完成后,环境会对其做出评价并得到一个奖励值,DRL 根据奖励值更新网络参数,不断迭代,直至学习到避障策略。

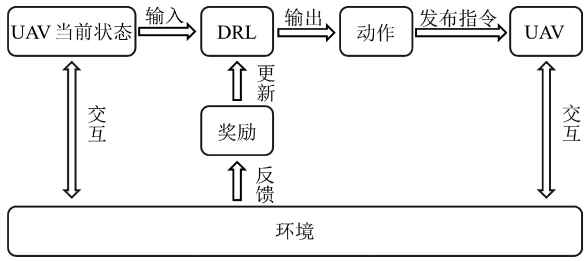


图 1 避障控制系统结构图

Fig. 1 Structure diagram of obstacle avoidance control system

1.2 状态空间设定

状态空间 S 是 UAV 感知环境的信息,是 DRL 网络的输入。本文在仿真环境中进行训练,为了实现避障功能,使用前置深度相机以及 IMU 作为环境感知器。本文的状态空间 $\boldsymbol{S}_t$ 分为两部分,一部分为深度相机采集到的连续四帧深度图像 $\boldsymbol{I}_D$;另一部分为 UAV 自身状态信息 $\boldsymbol{T}_s$,其包括 IMU 采集到的三轴线速度 $\boldsymbol{V}_s$、角速度 $\boldsymbol{\omega}_s$、线加速度 $\boldsymbol{a}_s$ 和角加速度 $\boldsymbol{\alpha}_s$、表示无人机当前姿态的四元数 $\boldsymbol{O}_s$ 以及上一时刻的动作 $\boldsymbol{a}_{t-1}$,具体表示为式(1)。

$$\begin{aligned}
&\boldsymbol{V}_s = [v_x, v_y, v_z]^{\mathrm{T}}\\
&\boldsymbol{\omega}_s = [\omega_x, \omega_y, \omega_z]^{\mathrm{T}}\\
&\boldsymbol{a}_s = [a_x, a_y, a_z]^{\mathrm{T}}\\
&\boldsymbol{\alpha}_s = [\alpha_x, \alpha_y, \alpha_z]^{\mathrm{T}}\\
&\boldsymbol{O}_s = [O_a, O_b, O_c, O_d]^{\mathrm{T}}\\
&\boldsymbol{a}_{t-1} = [a_{t-1}]^{\mathrm{T}}\\
&\boldsymbol{T}_s = [V_s^{\mathrm{T}}, \omega_s^{\mathrm{T}}, a_s^{\mathrm{T}}, \alpha_s^{\mathrm{T}}, O_s^{\mathrm{T}}, a_{t-1}^{\mathrm{T}}]^{\mathrm{T}}
\end{aligned} \tag{1}$$

本文中的深度相机采集到的原图像尺寸为 144×256,需进行预处理再使用。首先将其归一化至 0~255 之间,然后取反,按照式(2)降低强度,使暗像素点变得更暗,亮像素点变得略暗,以增强图像对比度,使障碍物信息更加清晰,效果如图 2 所示。之后将像素尺寸由 144×256 放缩到 55×84,截取中间 30×84 的区域并拉伸至 84×84 作为深度网络的图像输入,如图 3 所示。

$$newimage = 255 \times \left(\frac{image}{255}\right)^{50} \tag{2}$$

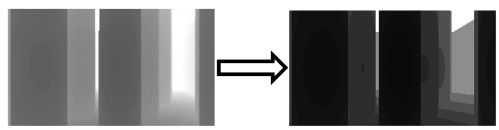


图 2 图像处理效果

Fig. 2 Image processing effect

三轴线速度、角速度、线加速度和角加速度均为(1,3)的张量,四元数为(1,4)的张量,上一时刻的动作为(1,2)的张量。不同属性的值数据范围差异较大,不利于网络训练

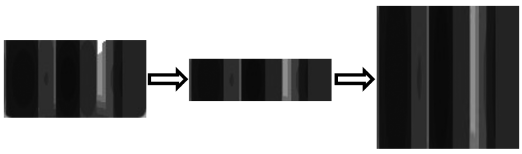


图 3　图像尺寸放缩

Fig. 3　Image size scaling

和收敛，所以将其全部标准化至[−1,1]区间内，再拼成(1,18)的张量作为神经网络的输入。

故本文的状态空间构成如图 4 所示。

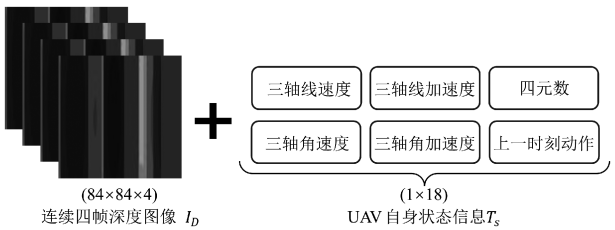


图 4　状态空间 $\boldsymbol{S}_t$

Fig. 4　State space $\boldsymbol{S}_t$

1.3　动作空间设定

UAV 的动作空间由俯仰角 θ 和偏航角 ψ 组成，当 θ 改变时，UAV 的前进速度会改变；当 ψ 改变时，UAV 的前进方向会改变。本文的深度相机位于 UAV 的正前方，故不允许后退。动作空间为连续的，θ 取值范围为[0.05,0.25](单位：rad)，ψ 取值范围为：[−π/4,π/4](单位：rad)。

1.4　奖励函数设计

奖励函数用于训练 UAV 完成任务。在经典避障导航任务中，只有当 UAV 到达目标或遇到障碍物时才会得到积极或消极的奖励，这意味着这种奖励是非常稀疏的。稀疏的奖励不利于 DRL 快速收敛，为了提高训练效率，本文塑造了集碰撞惩罚函数、目标引导函数、到达目标点奖励函数、摆动惩罚函数和时间步惩罚函数为一体的密集奖励函数。

1) 碰撞惩罚函数

当 UAV 与边界或环境中的障碍物发生碰撞时，会获得一个较大的惩罚值并返回起点重新开始，环境将会一直检测 UAV 是否发生碰撞且返回一个 bool 变量。为了引导 UAV 与障碍物保持安全距离，当未发生碰撞但过于靠近障碍物时也适当给予惩罚，所以本文的碰撞惩罚函数 r_c 可定义为：

$$r_c=\begin{cases}0, & collison=0,\ d_{ave}\geqslant 50\\ d_{ave}-50, & collison=0, d_{ave}<50\\ -200, & collison=1\end{cases}\tag{3}$$

其中，d_{ave} 为深度图像中央 40×40 像素点范围内的平均深度值，以 50 cm 为安全界限，当 $d_{ave}\geqslant 50$ 时认为 UAV 与障碍物之间的距离是安全的，当 $d_{ave}<50$ 时认为 UAV 与障碍物过近，并根据靠近程度给予惩罚。

2) 目标引导函数

为了使 UAV 不断地向目标点靠近，本文设计了以下目标引导函数 r_d：

$$d_{t-1}=\sqrt{(x_{t-1}-x_{dim})^2+(y_{t-1}-y_{dim})^2+(z_{t-1}-z_{dim})^2}\tag{4}$$

$$d_t=\sqrt{(x_t-x_{dim})^2+(y_t-y_{dim})^2+(z_t-z_{dim})^2}\tag{5}$$

$$r_d=\lambda(d_{t-1}-d_t)\tag{6}$$

如图 5 所示，$(x_{t-1},y_{t-1},z_{t-1})$ 表示 UAV 上一时刻的位置，(x_t,y_t,z_t) 表示 UAV 当前时刻的位置，$(x_{dim},y_{dim},z_{dim})$ 表示目标点位置，d_{t-1} 表示上一时刻 UAV 距目标点的欧氏距离，d_t 表示当前时刻 UAV 距目标点的欧氏距离，λ 为比例系数。当 $d_{t-1}<d_t$ 时，UAV 远离目标点，$r_d<0$ 表示惩罚；当 $d_{t-1}>d_t$ 时，UAV 趋近目标点，$r_d>0$ 表示奖励。

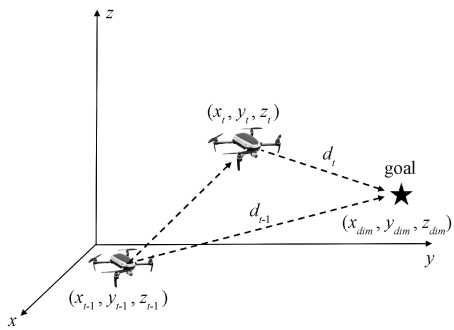


图 5　UAV 坐标系

Fig. 5　UAV coordinate system

3) 到达目标点奖励函数

到达目标点周围 10 cm 时给予 UAV 较大的成功奖励，其余时刻为 0。

$$r_e=\begin{cases}0, & d>10\\ 200, & d\leqslant 10\end{cases}\tag{7}$$

4) 摆动惩罚函数

由于 UAV 的移动方向由神经网络控制，网络学习初期难免会出现频繁左右摆动且摆动幅度过大的现象，为了引导 UAV 平稳运动，设计了如下摆动惩罚函数：

$$r_s=\begin{cases}0, & |\theta_t-\theta_{t-1}|\leqslant\dfrac{\pi}{4}\\ -|\theta_t-\theta_{t-1}|\div 10, & |\theta_t-\theta_{t-1}|>\dfrac{\pi}{4}\end{cases}\tag{8}$$

其中，θ_t/θ_{t-1} 表示当前时刻的转向角与上一时刻的转向角。

5) 时间步惩罚函数

时间步惩罚主要用于引导 UAV 在较短的时间段内完成任务，或是在执行任务过程中出现陷入局部最优等异常行为时结束任务并返回起点。本文的时间步惩罚函数为：

$$r_t = \begin{cases} -1, & step \leqslant max\_step \\ -200, & step > max\_step \end{cases} \tag{9}$$

其中,max_step 为设定的最大步数,本文将其设置为 200,在限定步数内,每走一步给予一定的惩罚。超过限定步数时,认为发生异常行为,给予一个与任务失败等价的惩罚并返回起点。

综上所述,本文的奖励函数为:

$$R = r_c + r_d + r_e + r_s + r_t \tag{10}$$

1.5 避障算法设计

算法 1 为本文的避障算法伪代码,其中设 ξ 为 DRL 网络权重参数,γ 为衰减系数,$batch\_size$ 为批大小,$done$ 为标志位,当达到目标点或发生意外时,$done = True$,无人机复位到起点。正常情况下,通过 $get\_states()$ 函数获取当前状态 s_t,将 s_t 作为 $DRL()$ 算法输入,网络会输出一组动作 a_t,将其作为 $step()$ 的输入,$step()$ 函数会控制 UAV 在环境中执行动作 a_t,并会返回当前奖励 r_t 和标志位 $done$,这一批次完成后,会根据奖励计算公式计算总奖励 R,并更新网络参数 ξ。

算法 1:避障算法

Input: $\xi, \gamma, batch\_size$
while $True$
  $step \leftarrow 0, R \leftarrow 0, done \leftarrow false$
  while $step < batch\_size$ *or* $done$ **do**
    $s_t \leftarrow get\_states()$
    $a_t \leftarrow DRL(s_t)$
    $r_t, done \leftarrow step(a_t)$
    $step \leftarrow step + 1$
  for $i := 0 \rightarrow step$ **do**
    $R \leftarrow accumulate\_reward(r_i, \gamma, R)$
    $\xi \leftarrow update\_params(a_i, s_i, R, \xi)$

2 GRU-A3C 算法及模型训练加速

本节对避障系统结构中的 DRL 进行具体设计,分为 GRU-A3C 算法和基于课程学习方法的模型训练加速两部分。

2.1 GRU-A3C 算法

Mnih 等提出的 A3C 算法是一种 AC 结构的方法,可以在多核 CPU 上运行,其计算成本低于 DQN 等方法[13]。在 AC(actor-critic)[14] 结构的基础上,A3C 进行以下改进[15]:A3C 结构如图 6 所示,算法创建多个并行环境,使得多个具有次级结构的智能体能够在这些并行环境中同时更新主体结构的参数;A3C 采用 n 步回归的方法,此轮奖励 R 按照式(11)计算,Actor 网络参数 δ 按照式(12)更新,Critic 网络参数 ϕ 按照式(13)更新。

$$R = \sum_{t=0}^{n} \gamma^t r_t \tag{11}$$

$$d\delta = d\delta + \nabla_\delta \log \pi_\delta(a_t \mid s_t)(R - V_\phi(s_t)) \tag{12}$$

$$d\phi = d\phi + 2(R - V_\phi(s_t))\nabla_\phi(R - V_\phi(s_t)) \tag{13}$$

其中,r_t 为当前步的奖励,γ 为衰减系数,策略 $\pi_\delta(a_t \mid s_t)$ 表示在状态 s_t 情况下采取动作 a_t 的概率,$V_\phi(s_t)$ 为价值函数。

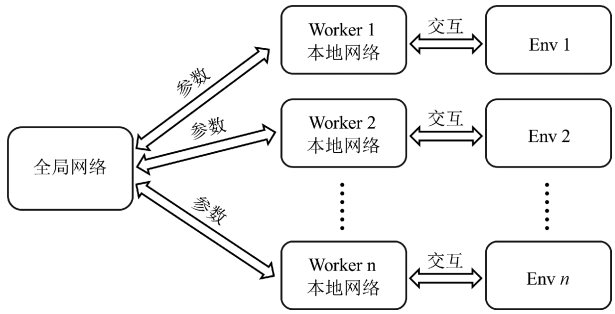


图 6 A3C 结构图

Fig. 6 A3C structural diagram

基于视觉的 A3C 网络通常使用连续 4 帧的图像信息作为输入,但其只能使 UAV 在学习过程中理解当前图像之间的时间关系,无法记忆历史信息。而空间中的障碍物不仅只存在于当前时刻,若当前图像中检测到障碍物,那么前后时刻图像也会存在该障碍物。此外,UAV 上一时刻的动作会导致其所处位置变动进而影响当前时刻的动作选择。可见,时空信息在视觉避障任务中非常重要。本文为了使 UAV 能够学习到时空关系,在 A3C 算法的 CNN 架构后面添加一层基于 RNN 的时序网络层,增加历史信息以加强对障碍物信息的分析。

传统 RNN 网络在序列较长的情况下会导致梯度消失和梯度爆炸[16],主流 RNN 网络变体是 LSTM(long short-term memory)网络,能缓解梯度的计算异常,但其内部结构复杂,在同等算力情况下,网络训练效率较传统的 RNN 网络更低。GRU 网络和 LSTM 作用相同,效果均优于传统 RNN 网络,但 GRU 网络的模型复杂度较 LSTM 网络小[17],训练模型的收敛速度快。GRU 有两个门,即复位门和更新门。复位门控制输入信息和记忆信息的融合,更新门控制记忆信息保存到当前时间步长的数据量。其原理是利用门控机制控制输入、记忆等信息,使信息可以选择性地影响 RNN 中当前时间的状态。GRU 结构图如图 7 所示,式(14)为其计算过程。

$$\begin{aligned} z_t &= \sigma(W_z \times [h_{t-1}, x_t]) \\ r_t &= \sigma(W_r \times [h_{t-1}, x_t]) \\ \tilde{h}_t &= \tanh(W \times [r_t \times h_{t-1}, x_t]) \\ h_t &= (1 - z_t) \times h_{t-1} + z_t \times \tilde{h}_t \end{aligned} \tag{14}$$

其中,x_t 为当前时间步的输入值,h_t 为当前时间步的隐状态输出,h_{t-1} 为上一时间步的隐状态输出,r_t 为重置门,z_t 为更新门,$\tilde{h}_t$ 为候选隐状态,σ 和 tanh 为激活函数。

本文选择 GRU 网络作为 RNN 网络核心加入 A3C 的

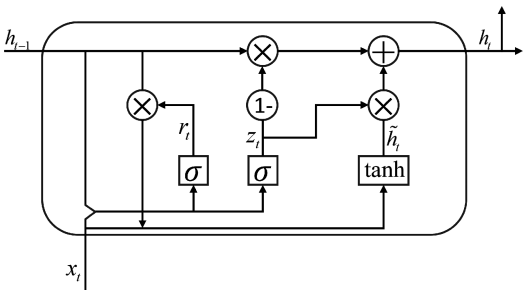


图 7　GRU 结构图

Fig. 7　GRU structural diagram

网络构建中，能够将 CNN 网络所忽视的图像序列之间的时间关系建立起来，使 UAV 在避障任务中能够结合短期记忆信息，选择更优的策略。模型网络层较少，且无需其他模型辅助，相较于文献[7-9]的系统更为轻量化。GRU-A3C 网络结构如图 8 所示。

2.2　基于课程学习方法的 A3C 模型训练加速

传统的基于 DRL 的 UAV 避障系统在模型训练时学习效率较低、收敛速度较慢，其原因主要是环境中障碍物较多、位置杂乱无序且起止点距离较远，训练初期很长一段时间 UAV 采取的动作随机性较大，持续碰撞，无法靠近或者到达终点，未学习到关键知识。

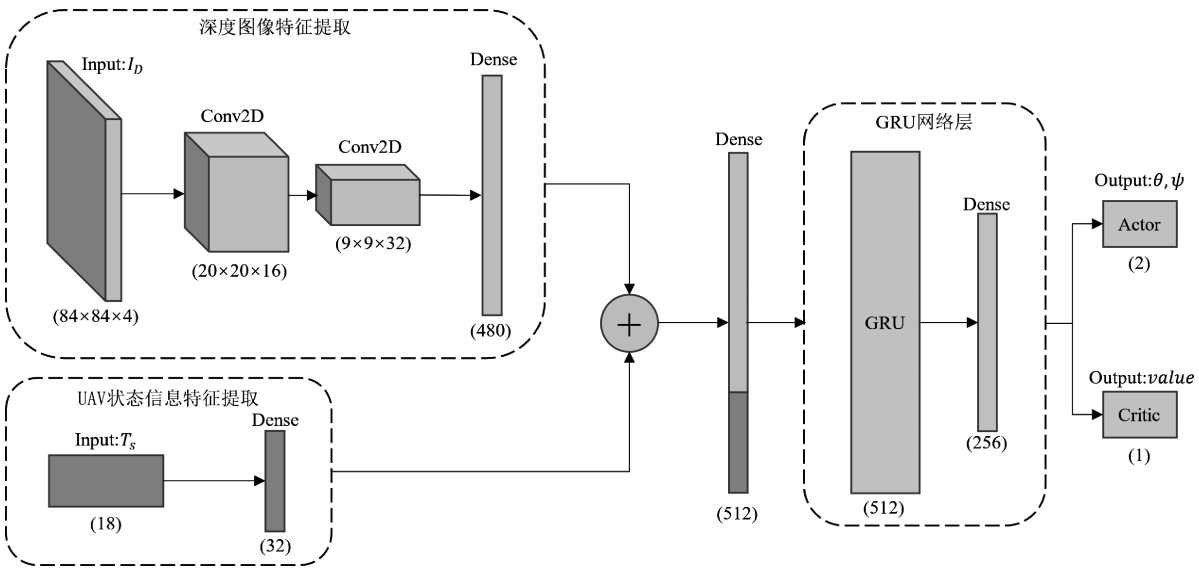


图 8　GRU-A3C 网络图

Fig. 8　GRU-A3C network diagram

基于 CL 的模型训练方法类似学生上课，需要由易到难、循序渐进地学习，利用这种方法，面对复杂任务可以有效提高模型收敛速度、减少训练时间。A3C 算法本身就是多个并行智能体在相同的环境中训练并共享权值，本文受到 CL 方法的启发，使 A3C 创建的多个智能体位于不同复杂度的环境中同时进行训练，如图 9 所示。障碍物数量与间隔是环境复杂程度的重要体现。环境复杂程度由易到难，智能体在简单环境较容易达到终点，学习到关键知识并获得正向奖励，并更新至全局网络，A3C 算法多个智能体共享全局权重，在较难环境下的智能体能通过下载全局网络参数来引导自身向目标点靠近，按照上述方式，能够加快智能体在最复杂环境中的模型训练速度，有效缩短训练时间。

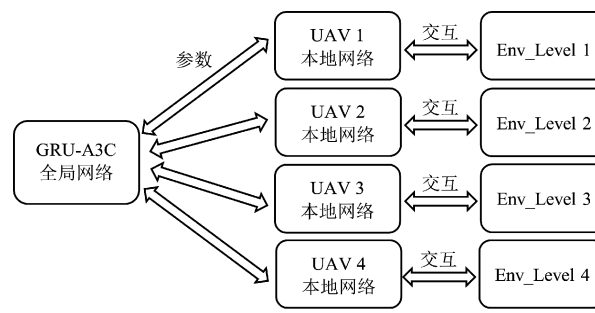


图 9　课程学习

Fig. 9　Curriculum learning

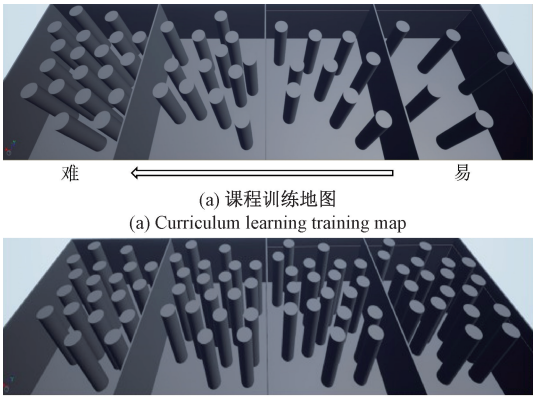
3　系统仿真实验

本研究采用 AirSim + UE4 仿真平台进行实验。AirSim+UE4 是一个微软开源具有逼真视觉的高还原仿真平台，其包含许多模块用来模拟真实环境，如天气条件、阴影、反射和重力等。

3.1　环境搭建

根据课程学习思想，搭建 4 个不同难度的子地图，在地图中放置多个圆柱墙体作为障碍物，每个子地图的尺寸和每个障碍物的尺寸一致，但障碍物数量和最小间距不同，课程训练地图如图 10(a)所示。同时搭建 4 个均为最高难度的子地图作为正常训练地图，如图 10(b)所示。在

每个子地图起点为固定一点,终点为 x 坐标随机,y 和 z 坐标固定的一点。



(a) 课程训练地图
(a) Curriculum learning training map

(b) 正常训练地图
(b) Normal training map

图 10 训练地图

Fig. 10 Training map

3.2 网络训练

网络超参数配置为:初始学习率为 0.000 1,折扣因子 γ 为 0.99,batch size 设置为 5,无人机个数为 4,分别在 4 个不同的地图中同时开始训练,4 个无人机均完成一个 batch 为一轮,每训练 1 000 轮次作为 1 次测试模型,采用逐轮递减的方式更新学习率以增强模型的训练效果。实验软硬件环境如表 1 所示。

表 1 软硬件环境配置

Table 1 Software and hardware environment configuration

CPU	Intel(R) Core(TM) i9-11900K 3.50 GHz
GPU	NVIDIA RTX3070Ti
RAM	32 GB
Operating System	Windows 10
Program Language	Python 3.7
ML Library	Keras 2.0.0
CV Library	OpenCV 4.8.1
Simulator	Airsim 1.8.1
Game Engine	4.27.2

3.3 实验数据及分析

本文以 A3C 作为基线进行消融实验验证算法的有效性(利用课程学习的方法训练的 GRU-A3C 记为 GRU-A3C-CL)。分别训练了 A3C、GRU-A3C 和 GRU-A3C-CL,其中 GRU-A3C-CL 训练地图为图 10(a),A3C 和 GRU-A3C 训练地图为图 10(b)。图 11 展示了上述 3 个模型在训练过程的平均奖励曲线,表 2 统计了训练模型测试 100 次的成功率。

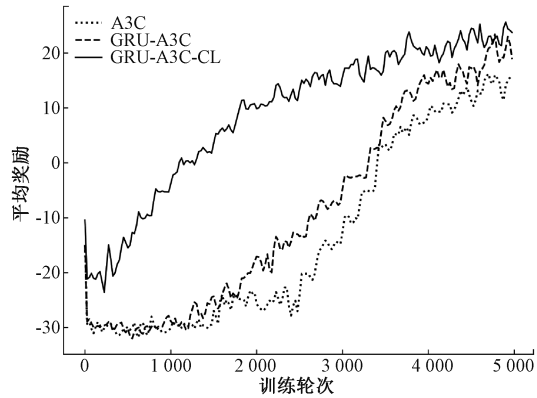


图 11 训练过程平均奖励图

Fig. 11 Average reward chart during training process

表 2 训练模型成功率

Table 2 Success rate of training model

训练轮次	A3C 成功率	GRU-A3C 成功率	GRU-A3C-CL 成功率
0	0	0	0
1 000	0.02	0.02	**0.28**
2 000	0.06	0.11	**0.53**
3 000	0.17	0.24	**0.60**
4 000	0.44	0.56	**0.66**
5 000	0.62	0.69	**0.72**

曲线图和表格数据显示,利用课程学习的方法训练的 GRU-A3C 在训练前期收敛速度较在正常地图训练的 A3C 和 GRU-A3C 显著提高,模型成功率大幅领先。3 000 轮次后 A3C 和 GRU-A3C 收敛速度开始提升,GRU-A3C 较 A3C 收敛速度无明显提升,但在训练相同轮次的情况下,GRU-A3C 的成功率提高,说明加入 GRU 网络后,对障碍物的分析能力增强,避障策略得到优化。综上所述,通过数据对比可知,用课程学习的方法训练的 GRU-A3C,既能大幅加快模型收敛速度,又能提高模型的效果。

4 结 论

针对基于深度强化学习四旋翼无人机视觉避障任务,本文设计了基于 DRL 的 UAV 避障系统,该系统以深度图像和 UAV 自身状态信息作为 DRL 输入,输出连续动作空间,使用一种基于 GRU 结构的 A3C 算法(GRU-A3C),并利用课程学习方法加速模型训练。该方法利用 GRU 网络记忆历史信息,融合时空特征,加强对障碍物信息的分析程度,提高了模型效果;使用课程学习的方法训练模型,有效提升了模型收敛速度。但训练后期,其在课程地图中的训练效率低于正常地图。这是由于在训练后期,课程地图中低难度的子地图对训练模型的影响较小,相当于仅有一个有效子地图。因此,后续的工作应当考虑在训练至合适

的程度时，将课程地图切换为正常地图继续训练，以保证训练效率。

参考文献

[1] 刘宜成，杨迦凌，梁斌，等. 基于强化学习的多段连续体机器人轨迹规划[J]. 电子测量技术，2024，47(5)：61-69.
LIU Y CH，YANG J L，LIANG B，et al. Trajectory planning of multi stage continuum robot based on reinforcement learning[J]. Electronic Measurement Technology，2024，47(5)：61-69.

[2] 邓修朋，崔建明，李敏，等. 深度强化学习在机器人路径规划中的应用[J]. 电子测量技术，2023，46(6)：1-8.
DENG X P，CUI J M，LI M，et al. Application of deep reinforcement learning in robot path planning[J]. Electronic Measurement Technology，2023，46(6)：1-8.

[3] MNIH V，KAVUKCUOGLU K，SILVER D，et al. Playing Atari with deep reinforcement learning[J]. ArXiv preprint arXiv：1312.5602，2013.

[4] 桓琦，谢小权，郭敏，等. 针对深度强化学习导航的物理对抗攻击方法[J]. 信息安全研究，2022，8(3)：212-222.
HUAN Q，XIE X Q，GUO M，et al. Physical adversarial attacks against navigation based on deep reinforcement learning[J]. Journal of Information Security Research，2022，8(3)：212-222.

[5] BANINO A，BARRY C，URIA B，et al. Vector-based navigation using grid-like representations in artificial agents[J]. Nature，2018，557(7705)：429-433.

[6] KALIDAS A P，JOSHUA C J，MD A Q，et al. Deep reinforcement learning for vision-based navigation of UAVs in avoiding stationary and mobile obstacles[J]. Drones，2023，7(4)：245.

[7] ROGHAIR J，KO K，ASLI A E N，et al. A vision based deep reinforcement learning algorithm for UAV obstacle avoidance[J]. ArXiv preprint arXiv：2103.06403，2021.

[8] YOKOYAMA K，MORIOKA K. Autonomous mobile robot with simple navigation system based on deep reinforcement learning and a monocular camera[C]. 2020 IEEE/SICE International Symposium on System Integration，2020：525-530.

[9] KIM M，KIM J，JUNG M，et al. Towards monocular vision-based autonomous flight through deep reinforcement learning[J]. Expert Systems with Applications，2022，198：116742.

[10] SALEM F M. Gated RNN：The gated recurrent unit (GRU) RNN[J]. Recurrent Neural Networks，2022：85-100. DOI：10.1007/978-3-030-89929-5_5.

[11] MNIH V，BADIA A P，MIRZA M，et al. Asynchronous methods for deep reinforcement learning[J]. ArXiv preprint arXiv：1602.01783，2016.

[12] ELMAN J L. Learning and development in neural networks：The importance of starting small[J]. Cognition，1993，48(1)：71-99.

[13] LEE D，KIM J，CHO K，et al. Advanced double layered multi-agent systems based on A3C in real-time path planning[J]. Electronics，2021，10(22)：2762.

[14] 蔡军，苟文耀，刘颜. 基于 actor-critic 框架的在线积分强化学习算法研究[J]. 电子测量与仪器学报，2023，37(3)：194-201.
CAI J，GOU W Y，LIU Y. Research on online integral reinforcement learning algorithm based on actor-critic framework[J]. Journal of Electronic Measurement and Instrumentation，2023，37(3)：194-201.

[15] ZHU K，ZHANG T. Deep reinforcement learning based mobile robot navigation：a review[J]. Tsinghua Science and Technology，2021，26(5)：674-691.

[16] 石庆研，张泽中，韩萍. 基于时空特征融合的 Encoder-Decoder 多步 4D 短期航迹预测[J]. 信号处理，2023，39(11)：2037-2048.
SHI Q Y，ZHANG Z ZH，HAN P. Multi-step 4D short-term trajectory prediction using Encoder-Decoder with spatio-temporal features fusion[J]. Journal of Signal Processing，2023，39(11)：2037-2048.

[17] 杨丽，吴雨茜，王俊丽，等. 循环神经网络研究综述[J]. 计算机应用，2018，38(S2)：1-6，26.
YANG L，WU Y Q，WANG J L，et al. Research on recurrent neural network[J]. Journal of Computer Applications，2018，38(S2)：1-6，26.

作者简介

马澳华，硕士研究生，主要研究方向为深度强化学习和无人机视觉避障。

E-mail：4022040031@mails.qust.edu.cn

邢关生（通信作者），博士，副教授，硕士生导师，主要研究方向为智能机器人系统、多机器人协调和机器人视觉。

E-mail：xinggs@qust.edu.cn