中国科学院大学
University of Chinese Academy of Sciences

强化学习实验报告

机器人导航

2025 年 6 月 1 日

目录

摘要 2

1 任务和环境介绍 2

2 相关工作 3

2.1 点目标导航 3

2.2 导航中的强化学习方法 4

3 方法 5

3.1 整体框架 5

3.2 含辅助任务的 PPO 算法 5

3.2 基线模型 6

4 对比实验 7

4.1 位姿预测的影响 8

4.2 不同图像特征提取器 9

4.3 Actor 网络参数初始化 11

4.4 改进奖励函数 11

5 总结 13

5.1 实验总结 13

5.2 局限性与未来工作 14

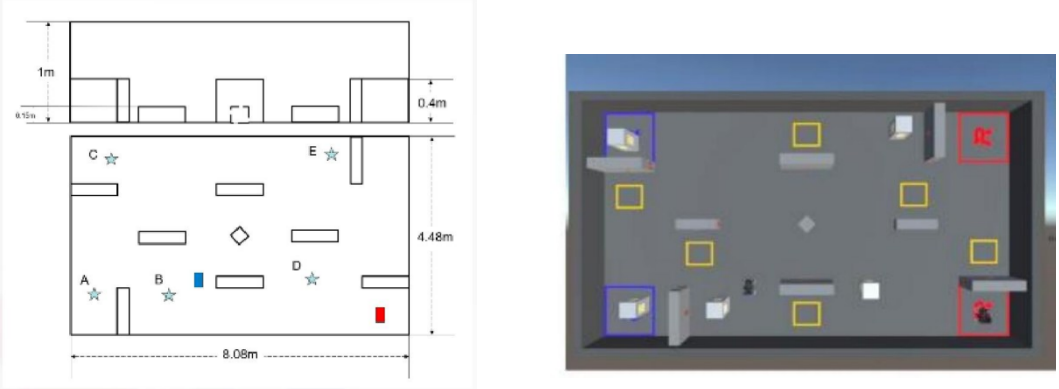
参考文献 16

摘要

本实验基于机器人对抗仿真场景，研究视觉引导下的点目标导航任务。实验采用 Unity3D 作为仿真平台，机器人通过第一视角 RGB 图像感知环境，在无先验地图信息的条件下，实现避障导航并完成矿石抓取。实验构建了基于 PPO 的强化学习模型，并引入模块化结构，通过辅助监督学习任务，预测目标点在机器人空间下的位置，提升训练效率与导航性能。通过对不同视觉特征提取器（如 EfficientNet、SqueezeNet、浅层 CNN）的对比分析，验证了轻量模型在低复杂度任务中的优越性；通过不同的网络架构以及设计技巧，验证了所提出模型的性能优势；在原始奖励函数的基础上设计了自适应朝向奖励机制，以强化机器人的有效对准行为，从而显著减少平均导航步数，提升策略稳定性。整体结果表明，融合强化学习、监督学习与合理网络结构设计的模块化导航系统在本实验的视觉导航任务中具备良好性能。代码见 https://github.com/Soappyooo/RL_Navigation。

1 任务和环境介绍

本实现的任务原始版本为 2022 CoG Robomaster Sim2Real 竞赛中的机器人对抗赛仿真场景，该场景开发了一个框架，具有速度快的敏捷物理机器人，并可用于训练机器人导航和对抗策略。仿真场景包含基于 Unity3D 的仿真平台，包括多种场景、机器人模型、控制器和各种传感器。本实验中的具体场景地图设置如图 1.1(a)所示，基于 Unity3D 的仿真平台俯视图如图 1.1(b)所示。



(a) (b)

图 1.1 实验仿真场景

任务描述：机器人在随机位置按照随机 SO(3)姿态初始化，机器人需要根据视觉信息（第一视角 RGB 图像）避开障碍物并寻找矿石，完成抓取矿石的任务。

动作空间：机器人x方向的速度v_x，y方向的速度v_y，z方向的角速度z_w，机械臂角度θ_{arm}，夹爪的状态$s_{gripper}$。其中$-10 \le v_x \le 10$、$-10 \le v_y \le 10$（单位 m/s），$-5 \le z_w \le 5$（单位 rad/s），$0 \le z_w \le \frac{\pi}{2}$（单位 rad/s），夹爪状态为 Bool 类型，True 代表关闭，False 代表开启。

观察空间：机器人在地图世界坐标系中的 SO(3)位姿和三维坐标（可选是否使用）、RGB 场景图像。

2 相关工作

2.1 点目标导航

本实验以 RGB 图像（或/和机器人状态）作为输入，导航机器人到环境中心点，该任务可以视为简化的点目标导航任务。点目标导航（Point Goal Navigation, PointNav）指给定一目标点的相对坐标，控制智能体在环境中移动并到达目标[1]。正式的定义为：给定环境集合$\mathcal{E}$，智能体的状态空间$\mathcal{S}$，观测空间$\mathcal{O}$，动作空间$\mathcal{A}$，目标空间$\mathcal{G}$，一个导航任务的示例为$\tau = (E, g, s_0)$，其中$E \in \mathcal{E}$为具体物理环境，$g = (\Delta x, \Delta y) \in \mathbb{R}^2$为目标空间的一个可行目标点，$s_0 \in \mathcal{S}$为初始状态。智能体应学习一个观测空间和目标空间到动作空间的映射函数，即策略$\pi: \mathcal{O} \times \mathcal{G} \to \mathcal{A}$，并最小化路径长度。观测空间通常包括以下列举的一种或几种：智能体感知的 RGB 图像，深度图像，GPS+罗盘信息，里程计等。

点目标导航的方法根据是否显式建模环境，可以分为两大类，这里主要对非显式建模方法进行介绍。非显式建模方法通过神经网络提取环境的潜在特征，而非依赖于显式构建的环境几何特征。这类任务可进一步分为端到端的方法以及模块化的方法。端到端的方法通常采用强化学习[2]或模仿学习[3]，后者依赖于大量的专家路径。模块化方法在端到端策略的基础上，添加辅助的有监督学习任务，包括预测智能体状态[4]，感知信息[5]等。

端到端方法中，DD-PPO[2]采用去中心化分布式 PPO 进行训练，使用 ResNet 提取图像特征，并经过两层 LSTM 整合帧间信息，每 128 帧进行一次更新。ViNT[3]提出了一个视觉导航的基础模型，在 100 小时的实机轨迹上进行训练。其网络采用 EfficientNet 和 Transformer 解码器，从历史若干帧观测中预测行动序列，该方法还可以有效迁移到物体目标导航（ObjectNav）等其它任务上。

模块化方法中，Zhao 等人[4]在没有 GPS 和罗盘和有噪声的输入输出环境中，通过构建视觉里程计（Visual Odometry，VO）模型预测智能体的帧间位移，并替换 GPS 和罗盘输入，验证了纯视觉导航的有效性。SplitNet[5]在策略网络之外添加了辅助任务的学习，从输入 RGB 图像预测法线图，深度图，下时刻动作和特征，且隔断策略网络向共享的图像编码器的梯度传播。

本实验中，使用模块化方法，在强化学习策略之外还使用有监督学习预测智能体离目标的二维距离$(\Delta x, \Delta y)$，并验证了相对于其它方法的优势。

2.2 导航中的强化学习方法

强化学习中，如何高效快速地收集样本是训练的一大瓶颈。DD-PPO[2]提出去中心化分布式训练，每个工作进程都负责收集经验与优化模型，在通信阶段同步梯度，并设置同步阈值避免某些进程缓慢导致整体等待。它采用标准的 PPO 算法优化策略，设计了密集的奖励函数：包括一个终止奖励$r_T = 2.5\text{SPL} = 2.5S\frac{l}{\max(l,p)}$，其中SPL（Success weighted by Path Length）为导航任务常见的评价指标，S为成功标识符（1 表示成功到达目标，0 则失败），l为到目标的最短路程，p为实际路程；一个行为奖励$r_t = -\Delta_{\text{deo_dist}} - 0.01$，其中$\Delta_{\text{deo_dist}}$为当前时刻与上一时刻离目标点的距离差，该奖励鼓励智能体靠近目标点，并减小时间惩罚。许多后续工作沿用或改进了 DD-PPO 方法[6], [7], [8]。PIRLNav[8]采用先行为克隆预训练，再强化学习微调的方式。首先对人类演示进行模仿学习，为强化学习提供合适的起点。强化学习阶段，第一步固定其它参数，只训练 Critic 网络，第二步再训练整个网络。VER[6]在 DD-PPO 的基础上，允许每个环境收集不同长度的经验，通过动态填补处理变长序列，以支持 RNN 等时序模型的高效训练。

本实验中采用 PPO 方法在单个机器，多个环境中收集经验，总序列长度达到阈值后进行一轮训练。具体方法在第三章详细介绍。

3 方法

3.1 整体框架

图 3.1 方法整体框架

基于[4], [5]等工作对于辅助任务在视觉导航中有效性的验证，本实验也采用加入辅助任务的模块化方法。以观测空间作为输入，通过特征提取网络，如 CNN 和 MLP，分别获得图像和状态的特征表示。这些特征经过 Actor 网络和 Critic 网络进行强化学习参数更新，经过辅助任务网络进行辅助任务的监督学习。每轮训练后，再与环境进行若干回合交互收集回放经验。方法的框架如图 3.1 所示。其中，观测空间包含机器人本体感知的 RGB 图像以及状态信息，在训练时我们使用图像和状态，在测试时不使用状态，只使用 RGB 图像。

3.2 含辅助任务的 PPO 算法

本实验在 PPO 算法上添加了监督学习的辅助任务损失。首先简单回顾一下 PPO 算法。PPO (Proximal Policy Optimization)是一种策略梯度算法，通过限制策略更新幅度保证稳定性。核心特征包括：

- 截断（PPO-Clip）：限制策略更新幅度
- 优势估计（GAE）：使用价值函数基线减少方差
- 多轮 mini-batch 更新：提高样本利用率

其中，优势函数计算公式为：

$$A_t = \sum_{k=0}^{T-t} (\gamma\lambda)^k \delta_{t+k} \tag{1}$$

$$\delta_t = r_t + \gamma V_\psi(s_{t+1}) - V_\psi(s_t) \tag{2}$$

策略目标函数表示为：

$$\mathcal{L}^{\mathrm{CLIP}}(\phi)=\mathbb{E}_t\left[\min\left(\frac{\pi_\phi(a_t|s_t)}{\pi_{\phi_{\mathrm{old}}}(a_t|s_t)}A_t,\mathrm{clip}\left(\frac{\pi_\phi(a_t|s_t)}{\pi_{\phi_{\mathrm{old}}}(a_t|s_t)},1-\epsilon,1+\epsilon\right)A_t\right)\right] \tag{3}$$

价值函数损失：

$$\mathcal{L}^{\mathrm{VF}}(\psi)=\mathbb{E}_t\left[\left(V_\psi(s_t)-R_t\right)^2\right] \tag{4}$$

总损失函数：

$$\mathcal{L}^{\mathrm{PPO}}=\mathcal{L}^{\mathrm{CLIP}}-c_1\mathcal{L}^{\mathrm{VF}}+c_2\mathcal{H}\left(\pi_\phi(\cdot|s_t)\right) \tag{5}$$

其中，ϵ为截断范围，c_1,c_2为损失缩放系数，$\mathcal{H}$为策略熵。价值函数损失由 Critic 网络优化，其余损失由 Actor 网络优化。

在 PPO 损失的基础上，我们引入辅助任务损失$\mathcal{L}^{\mathrm{AUX}}$，优化器的最终优化目标为：

$$\mathcal{L}^{\mathrm{TOTAL}}=c_3\mathcal{L}^{\mathrm{AUX}}+\mathcal{L}^{\mathrm{PPO}} \tag{6}$$

模型学习的辅助任务为：输入机器人本体感受的 RGB 图像，输出矿石坐标（世界坐标系原点）在机器人坐标系下的位置，即 GPS+罗盘信息。选择这样的状态信息作为预测目标，考虑因素包含两方面，其一是该目标（GPS+罗盘）与通常视觉导航任务的输入相符；其二是若采用机器人在世界坐标系下的位姿，由于场景对称，可能影响网络收敛。具体的，首先从 Unity3D 接口中获取机器人在世界坐标系下的位置和姿态，将姿态四元数转换为旋转矩阵，再将坐标表示从左手系变为右手系。之后，得到世界坐标系原点在机器人坐标系下的位置：

$$t_{\mathrm{world}}=-R_{\mathrm{robot}}^{T}\,t_{\mathrm{robot}} \tag{7}$$

其中t_{robot}为机器人在世界坐标系下的位置，R_{robot}为机器人在世界坐标系下的旋转矩阵，t_{world}为世界坐标系原点在机器人坐标系下的位置。由于实验是二维场地，我们只预测t_{world}的前两维，记作$(\Delta x,\Delta y)$，y轴正方向为机器人面向方向。辅助任务损失采用 MSE 损失。

3.3 基线模型

我们设计了一个基线模型来验证上述算法的有效性。该模型包含两个独立的网络，分别为辅助任务学习网络和策略网络。二者均采用三层 CNN 提取当前帧的 RGB 图像特征，辅

助任务学习网络在 CNN 后使用 MLP 输出预测的二维位置（后文称 pose）；策略网络在训练时使用真实二维位置，在测试时使用辅助任务学习网络输出的二维位置，与图像特征拼接后经过 MLP 搭建的 Actor 网络和 Critic 网络。结构如图 3.2 所示。

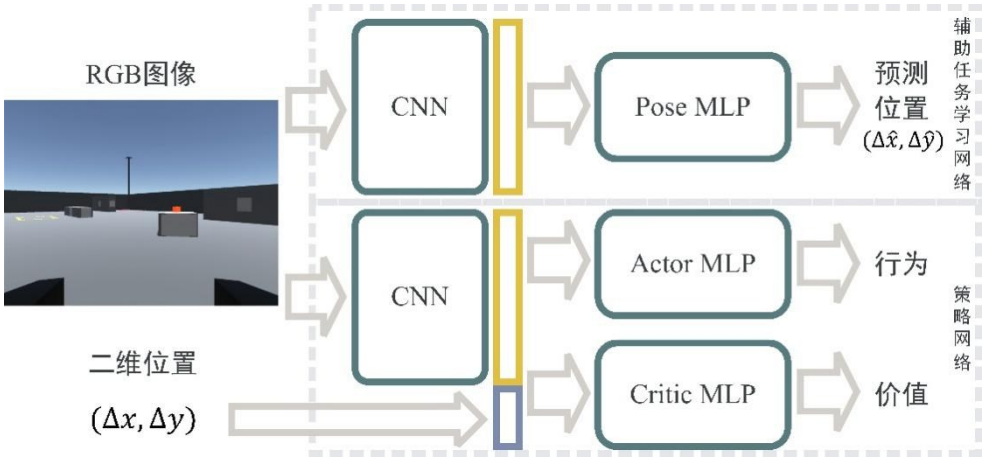


图 3.2 基线模型结构

使用这样的结构，可以让策略模型在训练时根据真实 GPS+罗盘信息学习合理的策略，在推理时不依赖于真实 GPS+罗盘信息，仅使用视觉输入导航。具体的模型细节上，CNN 采用三层卷积层，一层 LayerNorm，一层 AveragePooling，并展平映射，从宽高为 128 的图像提取 512 维特征。三个 MLP 均包含两个 256 维和 128 维的隐藏层，最终映射到特定的输出维度。两个独立模型的总参数量仅约 2.8M。

实现技巧上，根据工作[9]的大量实验以及常规做法，我们将 RGB 图像输入使用 ImageNet 参数归一化，将二维位置输入使用 Min-max 归一化至$[-1,1]$，将 Actor 网络正态初始化并把最后一层的初始化标准差缩小 100 倍，Actor 网络使用 Tanh 激活函数而其它网络使用 SiLU 激活函数，将 Actor 网络输出重参数化采样后反归一化到动作空间等。

训练 1M 步后，该方法可以达到较好的性能。在测试中随机进行 320 回合导航，回合步长最大 512 步，在非确定性动作下平均奖励（±标准差）为11.7 ± 2.07，平均回合长度（±标准差）为44.18 ± 64.36，成功率为 0.99。

4 对比实验

在本节中，我们将探讨并回答以下问题：

- 为什么要添加辅助任务？不同的模型结构是否会影响辅助任务和策略的学习效率？

- 三层卷积能否有效完成任务？使用预训练或更精密设计的 CNN 能否提高性能？
- 各个实现技巧是否有效，例如对 Actor 网络参数的初始化设置？
- 能否改进奖励函数提高性能？

由于设备有限，部分实验只进行了一组，另一部分实验进行了三组并取均值方差。

4.1 位姿预测的影响

我们对比了是否加入 pose 对结果的影响，实验结果显示，通过引入位姿预测，模型能够用更短的步长去找到目标，性能表现更好。

纯 RL 只有每回合末端奖励，信号稀疏；预测目标在机器人坐标系下的位置（Δx, Δy）在每一步都能计算 L2/CE 损失，给视觉编码器持续、平稳的监督，提高样本效率。预测位姿迫使网络在特征层面建立从 视觉 → 几何运动 的映射，隐式学到“我看见什么 → 我在哪里 / 将去哪”，从而减少感知混淆，决策更可靠。

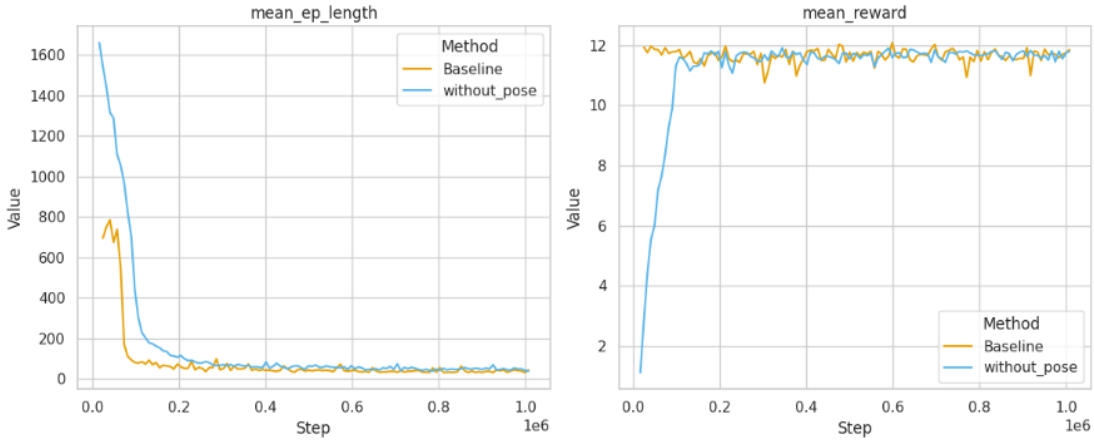


图 4.1 是否加入位姿预测对比

实验还探究了以三层卷积神经网络作为视觉特征提取器，policy 网络使用真实的 pose 和预测的 pose 对性能的影响，实验结果显示使用真实的 pose，模型能够有更好的表现。实验结果如下图。

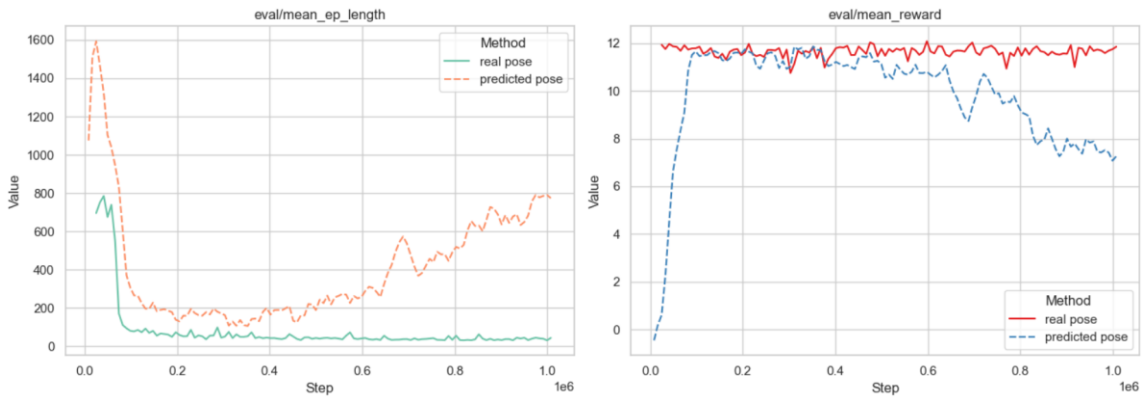


图 4.2 不同 pose 输入对性能的影响

4.2 不同图像特征提取器

在本次实验中我们使用了三种不同的视觉特征提取器（EfficientNetB0、SqueezeNet1_0 和三层卷积网络）探究不同图像特征提取器对机器人视觉导航任务中的影响。

EfficientNetB0 参数量大约 5.3M，该网络使用复合系数统一缩放网络深度、宽度和分辨率，以平衡准确性和效率，能够以更小的计算成本提取丰富、分层的特征。

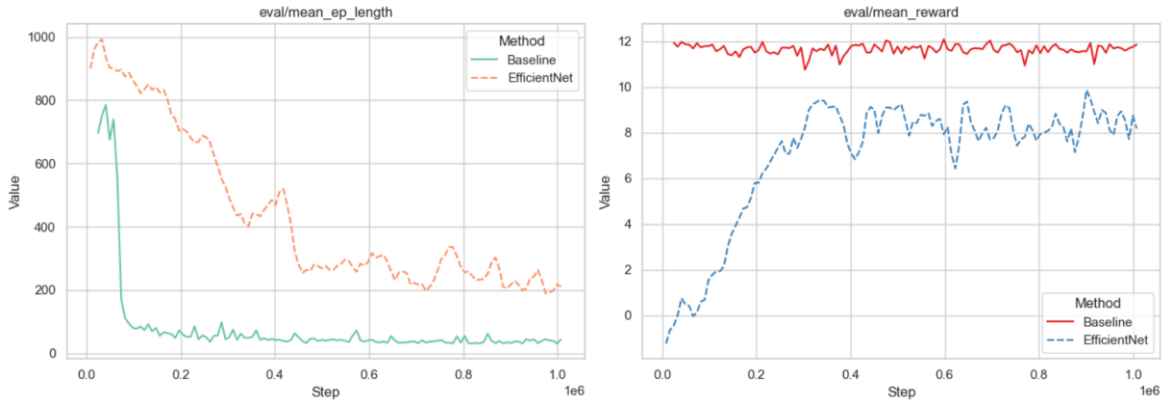


图 4.3 EfficientNet 和 Baseline 对比

通过实现发现，EfficientNet 作为视觉特征提取器，机器人到达目标位置的平均步数需要 277 步，而 baseline 使用三层卷积神经网络作为图像特征提取器到达目标平均位置只需要 42 步。虽然 EfficientNet 模型参数量更大，但是实验效果没有三层卷积神经网络作为图像特征提取器的实验效果好。猜测是实验场景比较简单，参数量大的模型训练难度更高。

SqueezeNet1_0 参数量大约 1.25M，SqueezeNet 的仅使用 AlexNet 1/50 的参数量，就达到了和 AlexNet 在 imagenet 上相同的性能。与 EfficientNet 相比，SqueezeNet 参数量更小，SqueezeNet 提取的特征在丰富性和细粒度上表现不如 EfficientNet，但是最终的实验效果却比 EfficientNet 好。使用 SqueezeNet 作为图像特征提取器，机器人到达目标位置的平均步数

需要 78 步。实验结果进一步说明，实验的场景较简单，使用简单的图像特征提取器便能达到较好的效果。

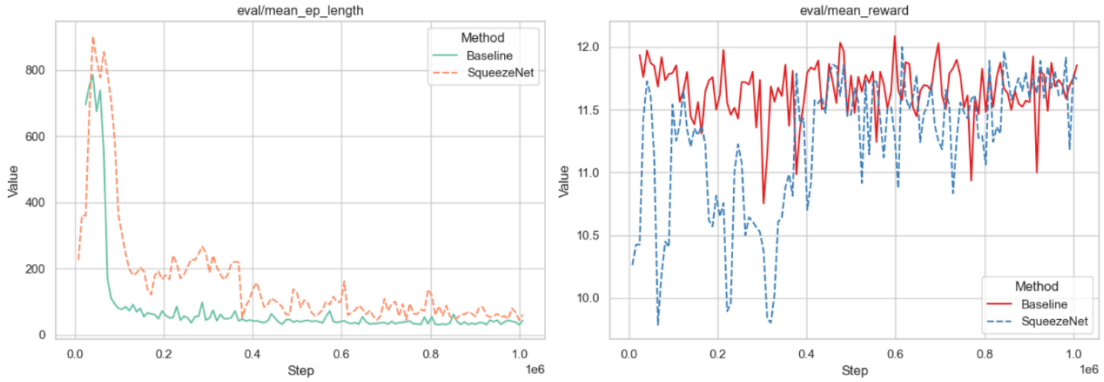


图 4.4 SqueezeNet 和 Baseline 对比

表 4.1 不同图像特征提取器性能

Model	Mean_length	Mean_reward
baseline	11.7±2.07	44.18±64.36
efficientnet	277.22±11.4	9.31±14.32
squeezenet	78.81±5.6	11.61±12.37

在实验过程中不仅考虑到图像特征提取器的类型对实验结果的影响，我们还比较了当 pose head 和 actor critic head 共用特征提取器对机器人导航性能的影响，下面左图是 baseline 的系统结构示意图，右图是 pose head 和 actor critic head 共用特征提取器的系统结构示意图。实验结果显示使用分开设计的 CNN 模块，模型能够有更好的表现。实验结果如下图。

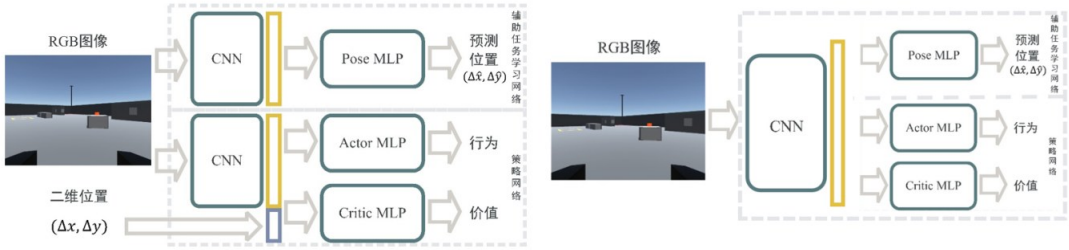


图 4.5 系统结构示意图

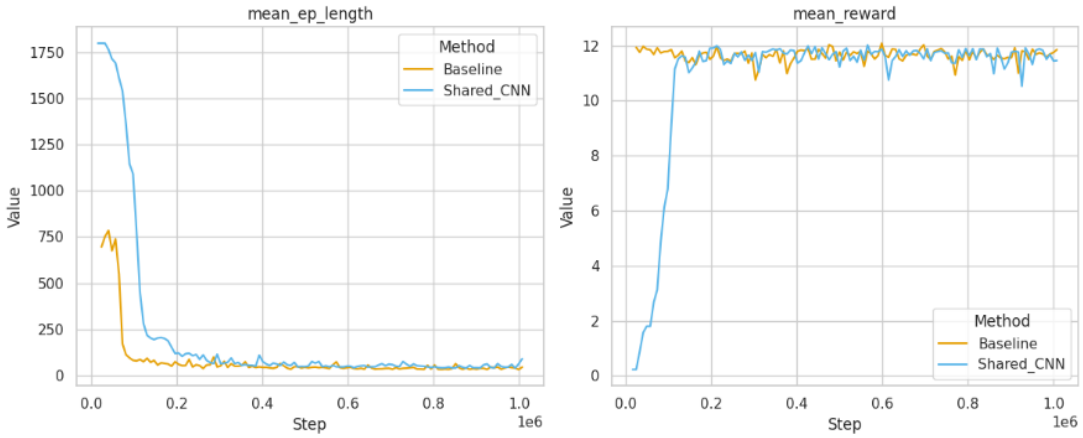


图 4.6 共用特征提取器对性能的影响

4.3 Actor 网络参数初始化

实验还探究了 actor 网络参数初始化对模型性能的影响。实验中尝试了不对 actor head 参数做处理和对 actor head 的参数使用正态初始化，并将最后一层 std 缩小 100 进行对比。实验结果表明，对 actor head 进行初始化，模型初始性能更好。

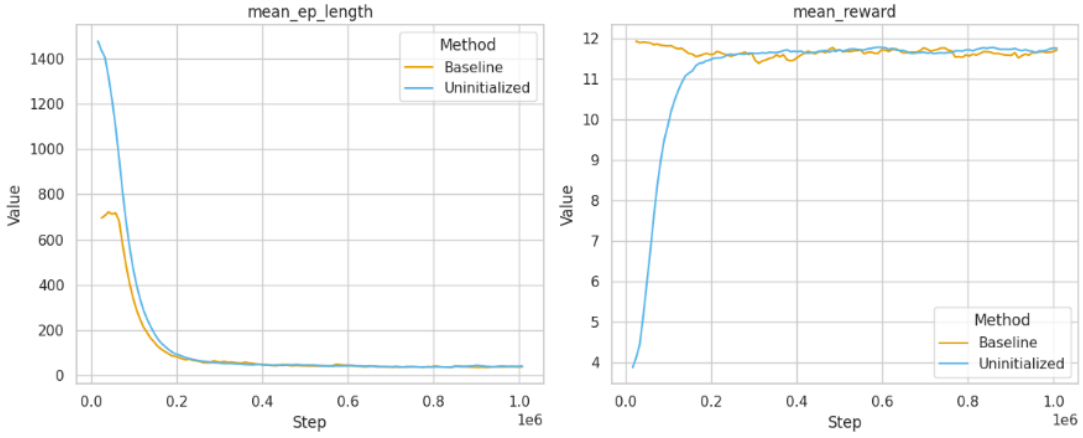


图 4.7 是否对 actor head 初始化输入对性能的影响

4.4 改进奖励函数

原始版本中环境中的奖励函数主要有以下两个部分：任务奖励 Task Reward 和距离奖励 Distance Reward。

Task Reward(R_{task})为 Unity3D 环境反馈的原始奖励，具体体现为机器人撞到环境中的障碍物或者发生倾倒时的奖励为-10，当机器人到达矿物附近并通过机械臂和夹爪成功抓取到矿物的奖励为 10。

Distance Reward(R_{dist})为由机器人当前的平面坐标和矿物平面坐标（原点）计算得到的欧氏距离，用于将机器人引导到矿物附近，从而加速任务的完成。其具体计算为

$$R_{dist} = \sqrt{x^2 + y^2} \tag{8}$$

原始环境的总的奖励函数为

$$R = R_{task} + R_{dist} \tag{9}$$

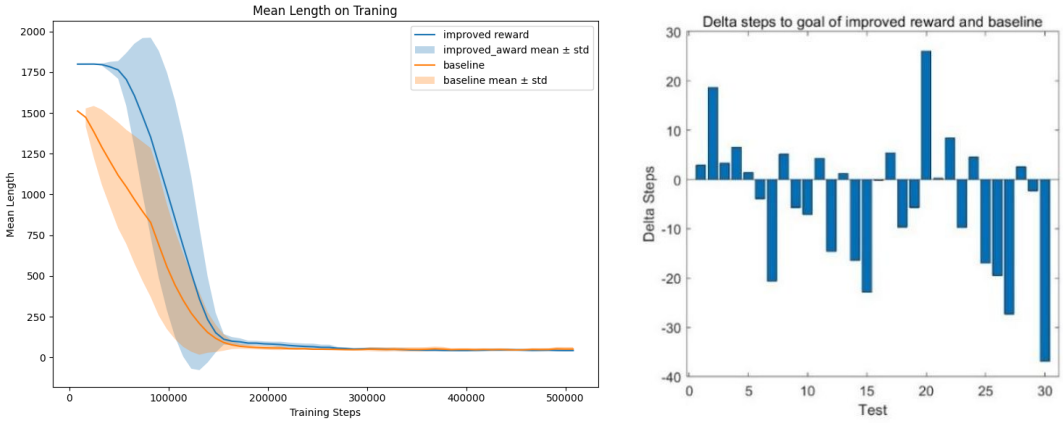
为了加速模型的收敛并提升模型性能，我们提出了一个改进的奖励函数，具体而言是在原始奖励的基础上增加了一个奖励项，用于奖励机器人始终朝向矿物的位置，从而方便机器人第一视角相机快速寻找到矿物。朝向奖励 Orientation Reward(R_{orient})，计算式为

$$R_{orient} = -0.1\alpha^2 \tag{10}$$

其中，α 为机器人朝向与机器人和矿物连线的夹角，$-\pi \leq \alpha \leq \pi$。表达式虽简单，但计算夹角的过程中需要涉及四元数欧拉角变换、左手坐标系到右手系的变换、夹角值域的变换等。改进后的总奖励函数为

$$R = R_{task} + R_{dist} + R_{orient} \tag{11}$$

我们使用改进奖励函数进行了模型的训练，训练结果对比如图 4.8(a)所示，同时我们验证了改进平均步数和 baseline 平均步数的差值（图 4.8(b)），发现模型性能没有明显的提升。



(a) (b)

图 4.8 改进奖励函数训练和对比结果

我们发现改进 reward 相较于 baseline 没有比较大的提升主要原因初始位置位于地图两侧两个较大障碍墙后时，由于奖励函数鼓励机器人朝向矿物，因此机器人会撞墙后卡住，导

致该次任务步长过长，计算平均步长时整体偏大。

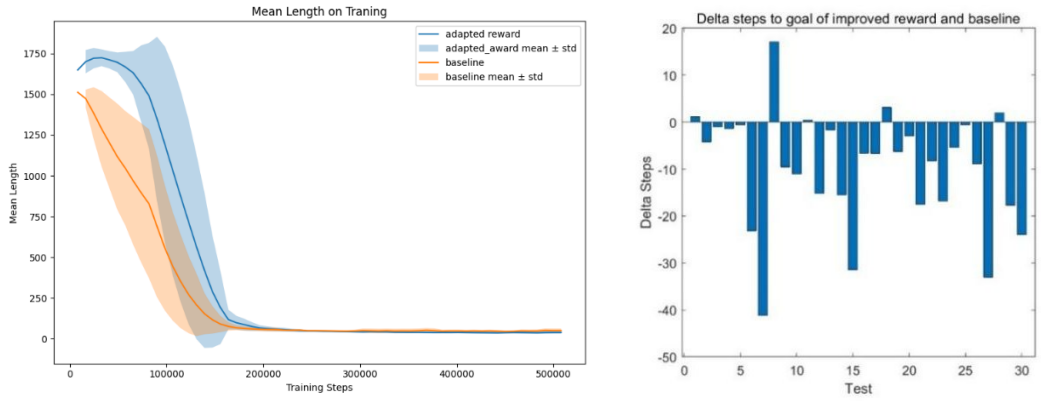
为了解决上述问题，我们希望距离矿物较远时，机器人朝向的奖励函数起到的作用较小，机器人靠近矿物时，朝向奖励起到的作用较大。我们提出了自适应朝向奖励 Adapted Reward(R_{adapt})，其计算式为

$$R_{adapt} = \frac{-0.1\alpha^2}{\sqrt{x^2 + y^2}} \tag{12}$$

改进后的总奖励为

$$R = R_{task} + R_{dist} + R_{adapt} \tag{13}$$

进行模型训练和对比验证，训练结果对比如图 4.9(a)所示，同时我们验证了改进平均步数和 baseline 平均步数的差值（图 4.9(b)），发现模型性有明显的提升，平均步长有明显的下降，从而证明了自适应奖励函数的有效性。



(a) (b)

图 4.9 自适应奖励函数训练和对比结果

5 总结

5.1 实验总结

本实验以 2022 CoG Robomaster Sim2Real 竞赛场景为背景，围绕视觉引导的机器人自主导航任务，设计并实现了基于强化学习的端到端导航模型。实验过程中，综合引入了最新的导航任务研究成果，结合点目标导航和强化学习领域的发展趋势，对模型结构和训练机制进行了系统设计和多项实验评估。

首先，模型设计方面，本实验构建了包含视觉特征提取、辅助监督训练任务和 Actor-

Critic 策略网络的端到端结构。特征提取部分探索了不同视觉编码器对导航性能的影响，结果表明：相比高参数模型（如 EfficientNet），结构简单的三层卷积神经网络在本实验场景中更具优势，体现出轻量模型在低复杂度任务中的有效性。

其次，在训练机制方面，采用 PPO 算法在多环境中并行采样训练，融合了监督学习目标（如目标距离预测）以增强模型学习效率。实验进一步表明，在策略网络中加入辅助监督位姿信息能够有效缩短导航路径，提高任务完成效率。

在奖励函数设计上，实验从原始任务奖励出发，提出了改进奖励函数，并在此基础上引入自适应朝向奖励机制。实验证明该机制能够在特定场景下显著提升导航性能，有效减少任务平均步数，从而提高整体任务完成效率与稳定性。

总体来看，实验验证了在机器人视觉导航任务中，适当结合强化学习与监督学习策略、多种视觉编码方式、合理的网络结构与奖励机制设计，能够有效提升导航性能与训练效率。后续研究可进一步探索更加复杂场景下的迁移能力、样本效率优化方法，以及融合多模态感知信息以提升泛化能力。

5.2 局限性与未来工作

时序特征建模缺失：当前模型仅使用单帧图像，没有使用历史观察图像。在早期的实验中，尝试过使用历史若干帧以及间隔的历史若干帧，采用 LSTM 或 Transformer 对时序特征进行建模，但在本实验简单的任务上效果不佳，收敛缓慢，因此没有展开进一步实验。后续可以进一步探索时序特征的建模方法，例如尝试 BPTT 等梯度传播策略。

复杂场景扩展：在障碍物分布更密集或目标点动态变化的场景中，进一步验证模块化方法与自适应奖励函数的泛化能力；

迁移学习应用：参考 ViNT 等基础模型的训练策略，通过预训练提升模型在低样本场景下的收敛速度。

Sim2Real 拓展：实验尚未在真实机器人验证，后续工作可以结合竞赛仿真框架（Robomaster Sim2Real），推进实物机器人部署验证。

综上，本实验通过理论方法对比与多组控制变量实验，明确了导航模型设计中任务解耦、轻量化特征提取与动态奖励机制的关键作用，为后续机器人自主导航研究提供了实验依据与优化思路。

6 贡献与分工

姓名	贡献(%)	分工
fdx	25	编写基线环境、模型、算法代码； 撰写摘要、第二章、第三章；排版与修改部分内容
wk	25	编写代码和实验验证改进奖励函数对模型性能的提高效果 撰写第一章、第四章 4.4 改进奖励函数
cq	25	进行位姿预测、共用特征提取、部分参数初始化的对比实验，撰写第四章部分内容和第五章
zb	25	进行不同类型视觉编码器的对比实验和部分参数初始化测试， 撰写第四章部分内容

参考文献

[1] I.-T. Ieong and H. Tang, “Multimodal Perception for Goal-oriented Navigation: A Survey,” Apr. 22, 2025, *arXiv*: arXiv:2504.15643.

[2] E. Wijmans *et al.*, “DD-PPO: Learning Near-Perfect PointGoal Navigators from 2.5 Billion Frames,” in *Proceedings of the International Conference on Learning Representations (ICLR)*, Sep. 2019.

[3] D. Shah *et al.*, “ViNT: A Foundation Model for Visual Navigation,” in *Proceedings of the 7th Annual Conference on Robot Learning (CoRL)*, Aug. 2023.

[4] X. Zhao, H. Agrawal, D. Batra, and A. G. Schwing, “The surprising effectiveness of visual odometry techniques for embodied pointgoal navigation,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 16127–16136.

[5] D. Gordon, A. Kadian, D. Parikh, J. Hoffman, and D. Batra, “SplitNet: Sim2Sim and Task2Task Transfer for Embodied Visual Navigation,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2019, pp. 1022–1031.

[6] E. Wijmans, I. Essa, and D. Batra, “Ver: Scaling on-policy rl leads to the emergence of navigation in embodied rearrangement,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 7727–7740, 2022.

[7] R. Partsey, E. Wijmans, N. Yokoyama, O. Dobosevych, D. Batra, and O. Maksymets, “Is mapping necessary for realistic pointgoal navigation?,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 17232–17241.

[8] R. Ramrakhya, D. Batra, E. Wijmans, and A. Das, “Pirlnav: Pretraining with imitation and rl finetuning for objectnav,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 17896–17906.

[9] M. Andrychowicz *et al.*, “What matters for on-policy deep actor-critic methods? a large-scale study,” in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2021.