2025 한양대학교

SW/ICT/AI 종합학술대회

[분야 : [L] Paper To Code Challenge]

제목: Extreme Parkour with Legged Robots 논문 구현

방동하(22학번), 이정현(22학번), 오승우(25학번)

2025. 11. 28

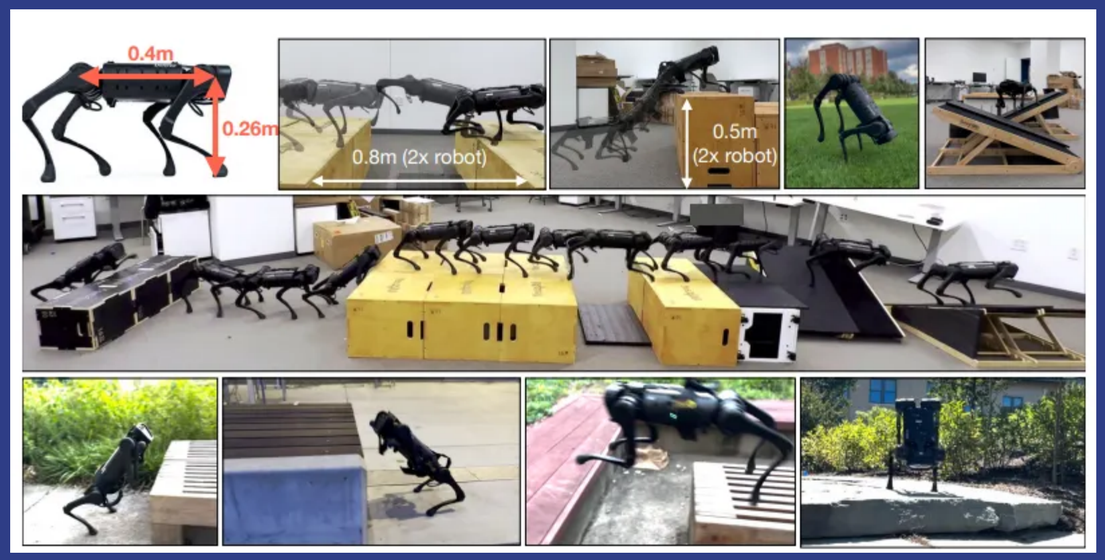
한양대학교 ERICA 소프트웨어융합대학 / 한양대학교 ERICA SW중심대학사업단

Extreme Parkour with Legged Rotos 논문구현

문제 정의 및 논문 요약

이 논문의 목표

" 저가·저정밀 센서/액추에이터를 가진 소형 로봇이,
깊이 카메라 한 개와 단일 신경망만으로 2배 높이·길이의 장애물을 넘는 파쿠르를 하게 만드는 것"



문제 정의

- 파쿠르는 작은 타이밍 오차에도 바로 실패하는 고난도 과제
  → 정밀 지각 + 정밀 제어를 동시에 요구

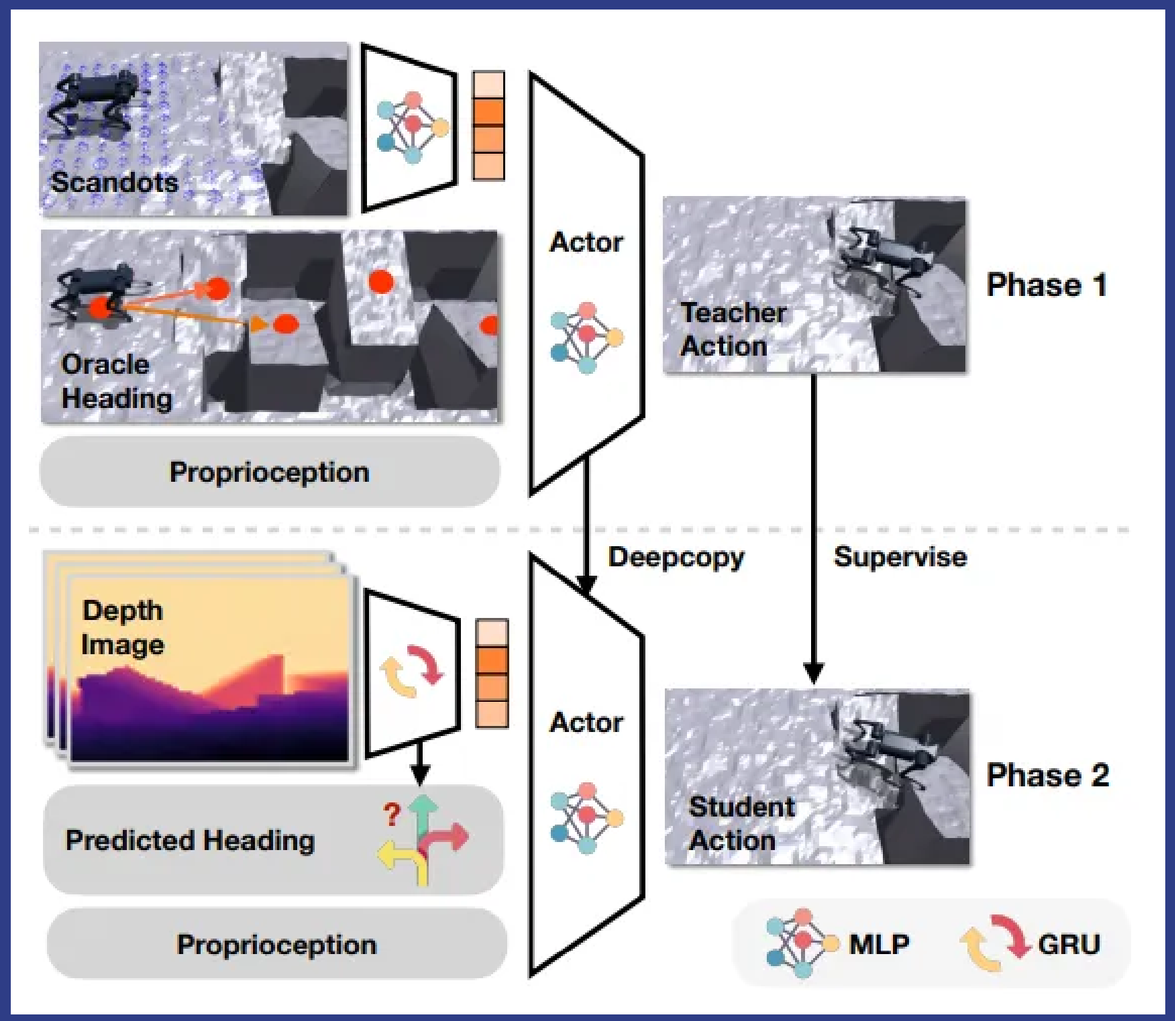
논문 요약

- 논문은 저가 4족 로봇(Unitree A1)과
  단일 전면 깊이 카메라만 사용
  (노이즈·지연 존재, 고정밀 센서/지도 없음)
- 입력: 깊이 이미지 + proprioception / 출력: 관절 제어 명령
- 명시적 맵·경로계획 없이 end-to-end RL policy 하나로

2× 높이 점프, 2× 길이 갭 점프, 경사로, 핸드스탠드까지 수행

Extreme Parkour with Legged Rotos 논문구현

방법론 핵심 구조



최종 policy

- 입력: 깊이 이미지 + proprioception
- 출력: 모터 명령 + heading(요 각도)

[Phase 1 (Teacher)]

- scandots(지형 높이), oracle heading 등 특권 정보를 사용해 RL(PPO)로 교사 정책 학습
- Unified Reward
- waypoint 방향과 속도 내적 등 단순한 통합 보상으로 점프·경사로·핸드스탠드 동작을 한 번에 학습

[Phase 2 (Student)]

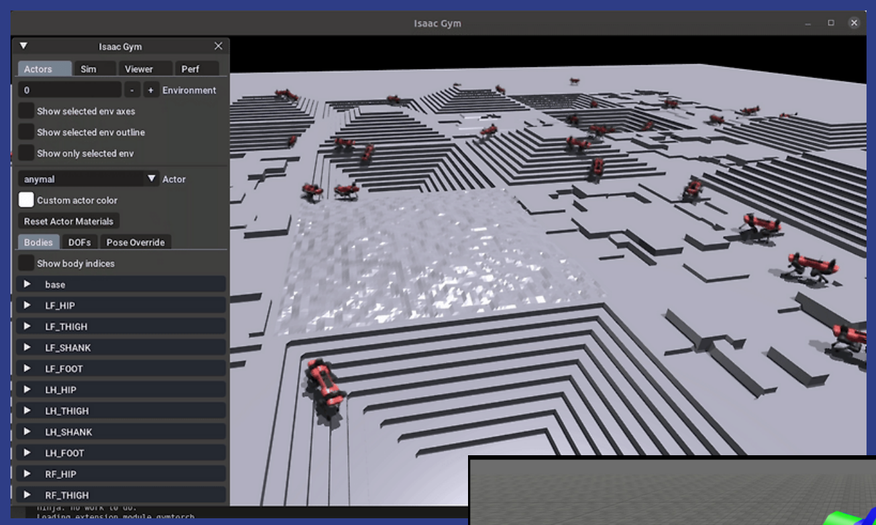
- 깊이 카메라 기반 CNN+RNN encoder로 teacher의 행동·heading을 distillation
- 실제 로봇에서는 depth + proprioception만으로 파쿠르 수행

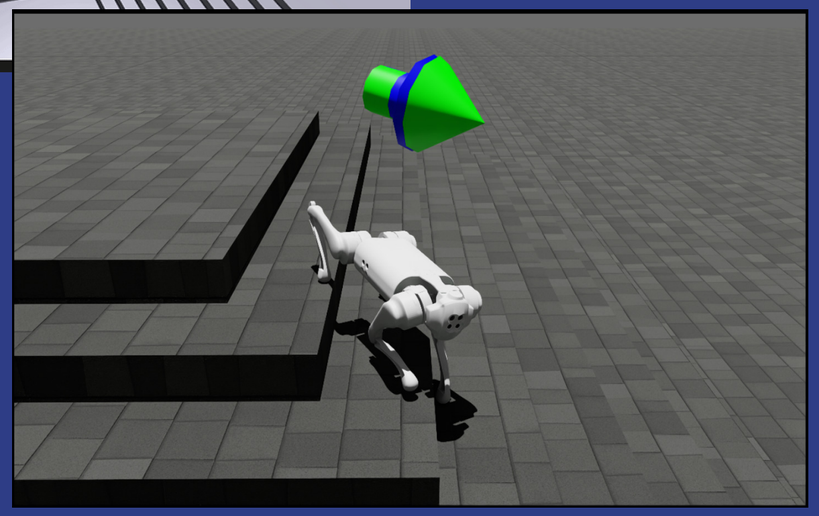
Extreme Parkour with Legged Rotos 논문구현

구현 목표 및 범위 정리

이번 과제의 목표

" **시뮬레이션 환경에서 학습 성공하기** "





원논문 : isaac gym에서 구현됨

BUT, isaac gym > isaac Lab으로 변경

isaac gym : 엔비디아에서 개발한 고성능 물리 시뮬레이션 환경

→ Isaac Lab 5.1.0 (최신버전)으로 구현함

Isaac Lab : NVIDIA 플랫폼을 기반으로 구축된 강화학습 기반 로봇 시뮬레이션 환경

원 논문은 sim2real (실제 로봇 까지 구현)로 구현했지만,
본 과제에서는 **시뮬레이션 환경에서 학습 성공하는 것까지 목표**

Extreme Parkour with Legged Rotos 논문구현

구현 상세 ① 환경/시뮬레이터 포팅

```
Isaaclab_Parkour/
├── LICENSE
├── README.md
├── setup.py
├── list_envs.py
├── parkour_isaaclab.egg-info/
│   ├── PKG-INFO
│   ├── SOURCES.txt
│   ├── dependency_links.txt
│   ├── top_level.txt
│   └── ...
│
├── parkour_isaaclab/              # 파쿠르용 IsaacLab 확장 패키지
│   ├── __init__.py
│   ├── parkour_viewport.py        # (예상) 뷰포트/카메라 컨트롤 관련
│   └── ...                        # 기타 유틸, 컨트롤러 등
│
├── parkour_tasks/                 # IsaacLab task 패키지 (pip -e . 하는 부
│   ├── setup.py / pyproject.toml  # editable 설치용 메타데이터
│   ├── parkour_tasks/
│   │   ├── __init__.py
│   │   ├── cfg/                   # task/agent 설정 YAML
│   │   │   ├── env/
│   │   │   │   ├── extreme_parkour_go2_env_cfg.yaml
│   │   │   │   ├── extreme_parkour_teacher_env_cfg.yaml
│   │   │   │   └── ...
│   │   │   └── agent/
│   │   │       ├── extreme_parkour_teacher_ppo_cfg.yaml
│   │   │       ├── extreme_parkour_student_ppo_cfg.yaml
│   │   │       └── ...
│   │   ├── terrains/              # 박스, 갭, 램프, 허들 지형 생성 코드
│   │   ├── envs/                  # Isaac-Extreme-Parkour-*-Unitree-Go
│   │   ├── utils/                 # 공통 유틸/헬퍼
│   │   └── ...
│   └── ...
│
├── parkour_test/                  # 모듈/뷰포트 테스트용 스크립트
│   ├── __init__.py
│   ├── test_viewport_camera.py    # ParkourViewportCameraController 테스
│   └── ...
│
└── scripts/
    └── rsl_rl/                    # rsl_rl 기반 학습/재생 래퍼
        ├── __init__.py
        ├── train.py               # Teacher / Student 학습 스크립트
        ├── play.py                # 학습된 policy 플레이
        ├── evaluation.py          # 학습된 policy 평가
        ├── demo.py                # IsaacLab 안에서 데모 실행 (sim 배포)
        └── modules/
            ├── __init__.py
            ├── on_policy_runner_with_extractor.py
            └── ...                # PPO 래퍼, runner 확장 등
```

목표

- Isaac Sim 4.5 기반 Parkour 코드를 Isaac Sim 5.1 + Isaac Lab 환경에서 재사용 가능하게 포팅

환경 구성

- OS : Ubuntu 22.04 사용
- conda로 Python 3.11 환경(isaaclab) 생성
- PyTorch CUDA 버전 설치 (RTX 4070 LapTop)
- isaacsim 5.1.0로 기본 시뮬 환경 구축

Parkour 레포 연동

- IsaacLab 루트에서 Isaaclab_Parkour clone + pip -e . 로 Isaac Lab에 통합
- Isaac Sim 5.1에서 Parkour 코스 로딩 및 실행까지 확인

포팅 과정 핵심 이슈

- typing_extensions, typeguard, isaacsim-kernel 버전 충돌 → 커널 요구 버전에 맞춰 정렬
- ROS2가 scripts 모듈을 가로채는 문제 → PYTHONPATH 정리로
- Parkour scripts 우선 사용

Extreme Parkour with Legged Rotos 논문구현

구현 상세 ② 학습 파이프라인 & 정책

```
feet_pos_x = ((self.asset.data.body_state_w[:, self.asset_cfg.body_ids ,0] + self.rows_offset)
    /self.horizontal_scale).round().long()
feet_pos_y = ((self.asset.data.body_state_w[:, self.asset_cfg.body_ids ,1] + self.cols_offset)
    /self.horizontal_scale).round().long()
feet_pos_x = torch.clip(feet_pos_x, 0, self.x_edge_masks_tensor.shape[0]-1)
feet_pos_y = torch.clip(feet_pos_y, 0, self.x_edge_masks_tensor.shape[1]-1)
feet_at_edge = self.x_edge_masks_tensor[feet_pos_x, feet_pos_y]
contact_forces = self.contact_sensor.data.net_forces_w_history[:, 0, self.sensor_cfg.body_ids] #(N,
previous_contact_forces = self.contact_sensor.data.net_forces_w_history[:, -1, self.sensor_cfg.body
contact = torch.norm(contact_forces, dim=-1) > 2.
last_contacts = torch.norm(previous_contact_forces, dim=-1) > 2.
contact_filt = torch.logical_or(contact, last_contacts)
self.feet_at_edge = contact_filt & feet_at_edge
rew = (self.parkour_event.terrain.terrain_levels > 3) * torch.sum(self.feet_at_edge, dim=-1)
## This is for debugging to matching index and x_edge_mask
# origin = self.x_edge_masks_tensor.detach().cpu().numpy().astype(np.uint8) * 255
# cv2.imshow('origin',origin)
# origin[feet_pos_x.detach().cpu().numpy(), feet_pos_y.detach().cpu().numpy()] -= 100
# cv2.imshow('feet_edge',origin)
# cv2.waitKey(1)
return rew
```

```
def reward_tracking_goal_vel(
    env: ParkourManagerBasedRLEnv,
    parkour_name : str,
    asset_cfg: SceneEntityCfg = SceneEntityCfg("robot"),
    ) -> torch.Tensor:
    asset: Articulation = env.scene[asset_cfg.name]
    parkour_event: ParkourEvent = env.parkour_manager.get_term(parkour_name)
    target_pos_rel = parkour_event.target_pos_rel
    target_vel = target_pos_rel / (torch.norm(target_pos_rel, dim=-1, keepdim=True) + 1e-5)
    cur_vel = asset.data.root_vel_w[:, :2]
    proj_vel = torch.sum(target_vel * cur_vel, dim=-1)
    command_vel = env.command_manager.get_command('base_velocity')[:, 0]
    rew_move = torch.minimum(proj_vel, command_vel) / (command_vel + 1e-5)
    return rew_move
```

전체 학습 구조
- PPO 기반 on-policy RL (rsl_rl)
- Teacher : scandots·oracle heading이 포함된 특권 관측으로 교사 정책(PPO) 학습
- Student : depth + proprioception만으로 교사 정책을 distillation해 학생 정책 학습

전진 속도 보상
- reward_tracking_goal_vel : 웨이포인트 방향 벡터와 로봇 속도의 내적을 보상으로 사용
→ "가야 할 방향으로 얼마나 잘 가는지" 측정

엣지 패널티
- reward_feet_edge : 발 위치 + terrain edge mask + 접촉센서를 이용해,
장애물 엣지(위험 구간)를 밟으면 큰 패널티 부여

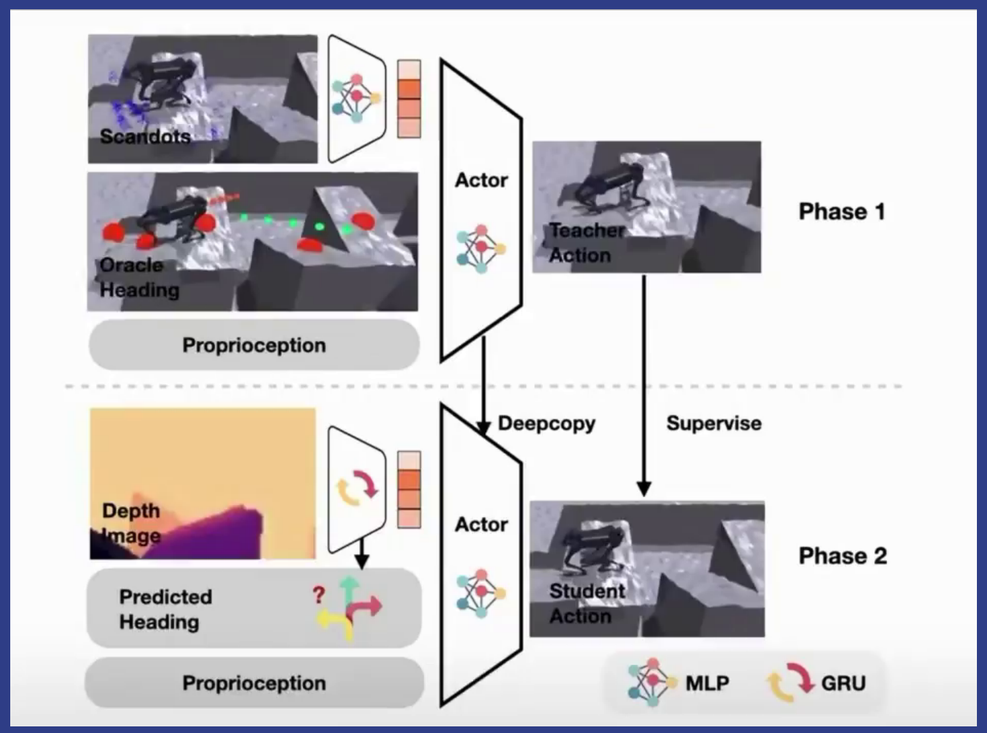
안정성 / 에너지 항
- reward_lin_vel_z, reward_orientation, reward_ang_vel_xy, reward_torques
등으로 상하 튐, 몸 기울어짐, 급격한 회전, 과도한 토크를 억제

최종 보상
- 위 항들을 가중합해서 하나의 통합 보상으로 사용
→ high jump / gap / ramp 같은 다양한 파쿠르 동작을 한 policy로 학습하도록 설계

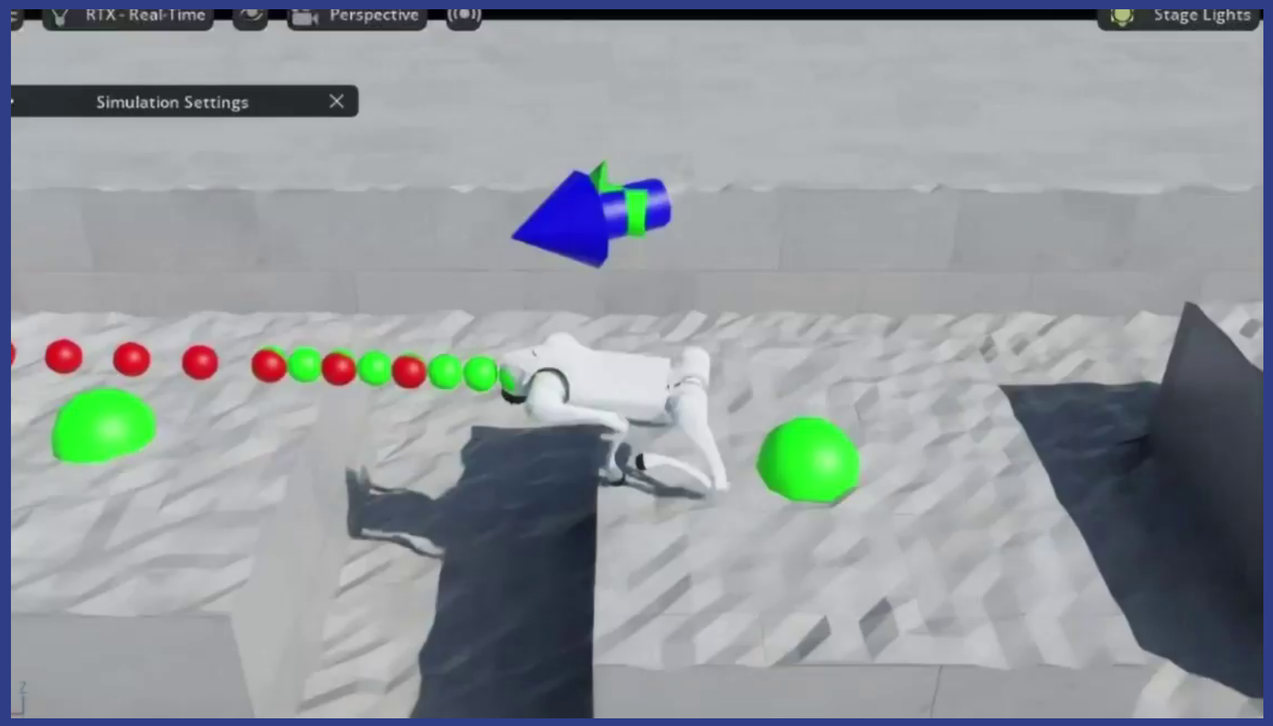
Extreme Parkour with Legged Rotos 논문구현

실험 결과 및 재현 수준

논문



구현 결과



정성적 재현 결과

1) 고난도 파쿠르 동작 재현

- 2× 높이 박스 점프, 2× 길이 갭 점프, 경사로 코스 주행 등은 원 논문 영상과 유사한 형태로
  성공적으로 수행되는 장면을 시뮬레이터에서 확인

2) 동작 안정성/자세

- 장애물 직전 감속 및 이륙 타이밍, 착지 후 회복 동작 등에서 논문에서 보고된
  "속도 유지 + 정확한 타이밍" 특성이 전반적으로 관찰됨

재현 수준 평가 (논문 vs 구현)

1) 환경 측면

- ✔ 시뮬레이터 버전 업그레이드(4.5 → 5.1) + Parkour 코스 구성 재현

2) 동작 측면

- ✔ pretrained policy 기준으로 주요 파쿠르 동작(점프/경사로)의 시각적/질적 재현
- policy를 처음부터 재학습한 결과는 아직 없음 (정량 지표 비교 불가)

3) sim2real 측면

- 실제 하드웨어(Unitree 계열)에서의 검증 단계는 향후 계획으로 남음

Extreme Parkour with Legged Rotos 논문구현

성과 및 한계점



1) 이번 구현에서 얻은 성과
- IsaacLab_Parkour(Isaac Sim 4.5 기반)를 Isaac Sim 5.1 + 최신 Isaac Lab 환경으로 이전
- 공개된 pretrained Extreme Parkour policy를 5.1 환경에서 실행하여
- 2× 높이/길이 점프, 경사로 등 논문 수준의 파쿠르 동작 일부를 시뮬레이터에서 재현

2) 한계 및 아쉬운 점
- 실로봇(Unitree 계열)에서의 실험은 진행하지 못하고 시뮬레이션 기반 재현 수준에서만 머무른 점

Extreme Parkour with Legged Rotos 논문구현

부록 (참고문헌)

1) Legged Locomotion in Challenging Terrains using Egocentric Vision

- Ananye Agarwal, Ashish Kumar, Jitendra Malik, Deepak Pathak

2) RMA: Rapid Motor Adaptation for Legged Robots

- Ashish Kumar, Zipeng Fu, Deepak Pathak, Jitendra Malik

3) A Reduction of Imitation Learning and Structured Prediction to No-Regret Online Learning

- Stephane Ross, Geoffrey J. Gordon, J. Andrew Bagnell

Extreme Parkour with Legged Rotos 논문구현

부록 (참고문헌)

4) Learning to Jump from Pixels

- Gabriel B. Margolis , Tao Chen , Kartik Paigwar , Xiang Fu , Donghyun Kim, Sangbae Kim, Pulkit Agrawal

5) ANYmal Parkour: Learning Agile Navigation for Quadrupedal Robots

- David Hoeller, Nikita Rudin, Dhionis Sako, Marco Hutter

6) Robot Parkour Learning

- Ziwen Zhuang, Zipeng Fu, Jianren Wang, Christopher Atkeson, Soeren Schwertfeger, Chelsea Finn, Hang Zhao

Extreme Parkour with Legged Rotos 논문구현

부록 (개발 이슈들)

- 오류 1 : typeguard 4.4.4 requires typing_extensions>=4.14.0,
- 오류 2 : isaacsim-kernel 5.1.0.0 requires typing_extensions==4.12.2
- 오류 3 : ModuleNotFoundError: No module named 'toml'
- 오류 4 : ModuleNotFoundError: No module named 'scripts.rsl_rl'
- 오류 5 : ImportError: cannot import name 'EmpiricalNormalization' from 'rsl_rl.modules'
- 오류5.1 : 위 해결방안으로 해결이 안될경우
- 오류 6 : ImportError: cannot import name 'dump_pickle' from 'isaaclab.utils.io'
- 오류 7 : TypeError: PPO.init_storage() takes 6 positional arguments but 7 were given
- 오류 8 : SyntaxError: invalid syntax
- 오류 9 : AttributeError: 'list' object has no attribute 'items'