

심층 강화 학습을 이용한 Luxo 캐릭터의 제어

이정민^o 이윤상^{*}

한양대학교 컴퓨터소프트웨어학과

{j0064423, yoonsanglee}@hanyang.ac.kr

Luxo character control using deep reinforcement learning

Jeongmin Lee^o Yoonsang Lee^{*}

Department of Computer Science, Hanyang University

요약

캐릭터로 하여금 시뮬레이션 내에서 사용자가 원하는 동작을 보이도록 물리 기반 제어기를 만들 수 있다면 주변 환경의 변화와 다른 캐릭터와의 상호작용에 대하여 자연스러운 반응을 보이는 캐릭터 애니메이션을 생성할 수 있다. 최근 심층 강화 학습을 이용해 물리 기반 제어기가 더 안정적이고 다양한 동작을 합성하도록 하는 연구가 다수 이루어져 왔다. 본 논문에서는 다리가 하나 달린 픽사 애니메이션 스튜디오의 마스코트 캐릭터 Luxo를 주어진 목적지까지 뛰어 도착하게 하는 심층 강화학습 모델을 제시한다. 효율적으로 뛰는 동작을 학습하도록 하기 위해서 Luxo의 각 관절의 각도값들을 선형 보간법으로 생성하여 참조 모션을 만들었으며, 캐릭터는 이를 모방하면서 균형을 유지하여 목표한 위치까지 도달하도록 하는 제어 정책(control policy)을 학습한다. 참조 동작을 사용하지 않고 Luxo 동작을 제어하도록 학습된 정책과 비교한 실험 결과, 제안된 방법을 사용하면 사용자가 지정한 위치로 Luxo가 점프하며 이동하는 정책을 더 효율적으로 학습할 수 있었다.

Abstract

Motion synthesis using physics-based controllers can generate a character animation that interacts naturally with the given environment and other characters. Recently, various methods using deep neural networks have improved the quality of motions generated by physics-based controllers. In this paper, we present a control policy learned by deep reinforcement learning (DRL) that enables Luxo, the mascot character of Pixar animation studio, to run towards a random goal location while imitating a reference motion and maintaining its balance. Instead of directly training our DRL network to make Luxo reach a goal location, we use a reference motion that is generated to keep Luxo animation's jumping style. The reference motion is generated by linearly interpolating predetermined poses, which are defined with Luxo character's each joint angle. By applying our method, we could confirm a better Luxo policy compared to the one without any reference motions.

키워드: 물리 기반 캐릭터 제어, 심층 강화 학습, Luxo

Keywords: Physics-based character control, Deep Reinforcement Learning, Luxo

1. 서론

게임과 같은 디지털 콘텐츠 내의 사실적인 가상 캐릭터 동작을 위해 가장 중요한 것들 중 하나는 캐릭터가 주변 환경의 변화에 반응하면서 사용자가 원하는 동작을 수행하도록 하는 것이다. 실 세계의 사람 및 동물, 그리고 주변 환경의 모든 움직임과 상호작용은 물리 법칙에 의해 기술되므로 사실적인 캐릭터 동작을 생

성하기 위해 물리 시뮬레이션을 통한 캐릭터 동작을 생성하려는 시도가 지속적으로 이루어져 왔다. 이러한 시도들에서 중요한 목표는 물리 시뮬레이션 되는 캐릭터가 균형을 유지하면서 원하는 동작을 수행하도록 하는 것이다. 이와 같이 캐릭터를 제어하는 방법에 대한 연구는 그동안 활발히 이루어져왔고, 그 방법론에는 유한 상태 기계 기반 제어기(SIMBICON) [1], 모션 캡처 데이터

*corresponding author: YoonsangLee/Hanyang University(yoonsanglee@hanyang.ac.kr)

를 간단한 규칙에 의해 변형하는 방식의 제어기 [2] 등 여러가지 접근방식이 있다.

최근 가장 주목 받는 방법은 심층 강화학습을 활용한 연구이다. 그동안 다양한 연구들을 통해 사람 [3], 사족보행 동물 [4], 비행 물체 [5] 등 다양한 캐릭터들을 심층 강화학습으로 제어하는 방법들이 제안되어 왔다. 주어진 모션 캡처 클립 없이 심층 강화학습으로 학습된 모델은 종종 자연스럽게 않은 결과를 보일 수 있기 때문에 [6] 최근의 심층 강화학습을 사용한 많은 연구에서는 모션 클립을 모방하는 모방 학습(imitation learning) 방식을 통해 자연스러운 동작을 만들어내고 있다 [1] [3] [7].

본 논문에서는 그동안 심층 강화학습 캐릭터 연구에서 주로 연구되어 왔던 사람, 사족보행 동물, 비행 물체가 아닌 하나의 다리를 가지는 가상 캐릭터 Luxo의 동작을 제어하는 것을 목표로 한다. 앞에서 기술한 모방학습 방식의 장점을 취하기 위해 Luxo가 동작하는 참조 모션을 만들어 학습에 사용하였으며, 랜덤하게 설정되는 목표 위치까지 균형을 유지하며 도달하도록 하였다. 이 논문에서 제안하는 제어기를 사용하면 쉽고 빠른 방법으로 Luxo 캐릭터의 동작을 학습 및 제어할 수 있으며, 이 방법은 키 프레임 동작들을 기반으로 한 모방 학습의 가능성을 보여주기도 한다.

2. 관련 연구

물리 시뮬레이션 내에서 캐릭터가 균형을 잃지 않고 자연스러운 동작을 할 수 있도록 여러 제어 알고리즘들이 제안되어 왔다. 그 예로는 사람의 직관을 통해 동작에 필요한 인자들을 설정하고, 동작을 분할해 유한 상태 기계(Finite State Machine, FSM)를 만들어 쓰는 방법 [1], 동작을 이차 계획법(quadratic programming) 문제로 공식화하여 FSM으로 정의된 목표 동작들을 따라 걷는 동작을 합성하는 방법 [8] 등이 있다. 그럼에도 역동적인 동작이나 장기적인 움직임, 접촉이 많은 환경에서의 자연스러운 동작이 어렵기 때문에 물리 시뮬레이션 환경에서 얻은 정보를 바탕으로 동작의 경로를 최적화하는 방법 [9], FSM에 기반한 제어기의 제어 인자를 최적화 기법을 통해 찾는 방법 [10], 미리 주어진 모션 캡처 데이터 셋 [2], 또는 영상 [11]을 이용하여 주어진 모델에게 동작의 자연스러움을 추구하도록 하는 방법 등이 제안되어 왔다.

최근에는 심층 강화학습(Deep Reinforcement Learning, DRL)을 적용한 연구가 많이 이루어지고 있다. 고차원의 연속적인 공간 상에서의 학습 정책을 학습하기 위해 REINFORCE [12], TRPO(Trust Region Policy Optimization) [13], PPO(Proximal Policy Optimization) [14]와 같은 방법들이 제안되어 왔다.

물리 시뮬레이션 환경에서 센서를 통해 얻을 수 있는 고차원 정보는 크기가 크고 캐릭터가 취하는 동작의 범위도 연속적이기 때문에 복잡하다. 심층 강화학습 정책은 이러한 고차원 환경으로부터 관찰한 값을 입력으로 캐릭터가 취할 수 있는 다음 동작을 효과적으로 출력하는데, 동작은 각 관절의 돌림힘, 캐릭터 근육의 활성화, 다음 시뮬레이션 단계에서의 캐릭터의 각 관절의 크기 또는 각속도처럼 다양하게 표현될 수 있으며 각 방법의 선택이

학습에 영향을 끼칠 수 있다 [15]. 심층 강화학습 정책은 캐릭터가 취한 동작에 대한 보상을 통해 주기적으로 갱신하면서 학습된다. 지나치게 단순한 보상을 주면 캐릭터는 자연스럽게 못된 동작을 보이기도 한다 [6]. 캐릭터가 대칭적인 행동을 보이도록 수정된 손실 함수(loss function) [16]를 포함해 여러 다른 방법들을 통해 캐릭터가 목표 동작을 수행하도록 학습시킬 수 있지만 정책이 주어진 데이터 셋을 모방하도록 학습시키는 모방 학습(imitation learning)을 통해서 더 사실적이고 역동적인 동작을 합성할 수 있다. 모방 학습은 목표하는 참조 동작과 시뮬레이션 된 동작의 차이를 계산하여 보상에 반영하고, 이렇게 학습시킨 모델은 물리 시뮬레이션 안에서 참조 동작을 따라하면서 균형을 잃지 않는 법을 학습한다. 2017년 발표된 연구 [17]에서는 두 단계의 네트워크를 이용해 각 동작들을 따라하도록 학습 시킨 후 목표를 달성하기 위해 필요한, 즉 가장 좋은 보상을 얻는 동작을 골라 실행하도록 정책을 학습시켰으며, 이어 2018년 후속 연구 [3]에서는 더 역동적인 동작들로 하위 목표를 달성하고, 동시에 다양한 캐릭터 및 환경에 적용시킬 수 있게 범용성을 넓혔다. 모방 학습을 통해 학습된 동작으로 사용자가 실시간으로 원하는 방향을 입력해 캐릭터가 나아가도록 학습시킬 수 있으며 [18], 이 밖에도 순환 신경망(Recurrent Neural Network, RNN)으로 생성시킨 동작을 이용해 심층 강화학습으로 위치, 속도 등 더 다양한 목표를 달성하고 여러 캐릭터들 간 상호작용할 수 있도록 학습시키는 방법 [7] 등 다양한 방법들이 제안되고 있다.

Luxo처럼 다리가 하나 달린 캐릭터를 움직이게 하기 위한 연구들 또한 다양하게 이루어져 왔다. 1988년에 이루어진 연구 [19]에서는 목표 동작에 해당하는 몇 가지 조건들을 입력으로써 뉴턴의 법칙을 만족하며 물리적으로 가능한 동작으로 최적화하였다. 이후 사람과, 사람과 구조가 전혀 다른 모델의 서로 다른 중요 동작(key pose)들을 정적으로 짝 지어 학습(GPLVM)시킨 것을 기반으로 사람과 전혀 다른 구조를 가진 Luxo의 동작을 사람의 모션 캡처 데이터를 이용해 합성하는 방법 [20], 페이즈(phase)를 사용하며 반복적으로 순환되는 동작의 구조가 정책 및 가치 네트워크에 바로 적용되도록 하여 다리가 한 개나 두 개인 캐릭터의 이동을 효율적으로 학습 시키는 방법 [21], CDM(Centroidal Dynamics Model)을 이용해 빠르고 유연하게 Luxo 등 다리를 여러 개 가진 캐릭터들의 동작을 최적화 하는 방법 [22] 등 다양한 방법들이 제안되었다.

이 논문에서는 모션 캡처 데이터를 구할 수 없는 가상의 캐릭터인 Luxo의 목표 동작을 선형 보간하여 생성한 후, Luxo 캐릭터가 심층 강화학습을 통해 생성된 동작을 모방하면서 효율적으로 학습된 자연스러운 동작으로 이동하는 정책을 학습한다.

3. 캐릭터 및 참조 모션 생성

3.1 캐릭터

Luxo의 몸체는 머리(4kg), 윗팔(8kg), 아랫팔(16kg), 발판(2kg) 총 4개의 부분(30kg)로 구성되어 있으며, 모델을 곧게 뿔 때의 높이는 1.5m이다. 각 부분을 잇는 세 개의 관절 중 처음과 마지막은 x축과 z축으로 회전이 가능하므로 Luxo는 총 다섯 개의 자유도를 가진다. 이 관절들을 이용하여 Luxo의 머리와 발판은 위, 아래, 양 옆을 향할 수 있다(Figure 1).

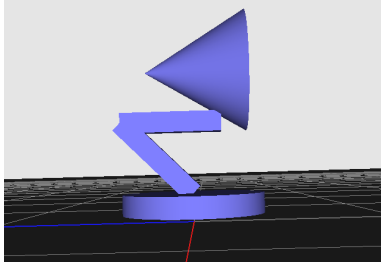


Figure 1: Luxo model

3.2 참조 모션 생성

시뮬레이션 공간의 수직축을 y축이라고 한다면, 다리가 하나 밖에 없는 Luxo는 점프 동작을 통해 xz평면 상에서 이동하게 된다. 이때 우리가 기대하는 것은 Luxo가 짧은 점프를 반복해 목표 위치에 도달하는 것이다. 이러한 점프 동작을 원활히 학습시키기 위해 참조 점프 모션을 만들어 이용하였다.

한 번의 점프 동작은 1초의 페이즈(phase) 동안 0.4초간 균형을 잡고(hold), 0.2초간 몸을 편 후(stretch), 0.4초간 다시 몸을 숙이는(fold) 동작 총 3구간으로 이루어져 있고, 각 동작들을 선행 보간하면 주어진 페이즈 값에 상응하는 내부 관절의 다섯 개의 자유도에 따른 값들을 알아낼 수 있다(Table 1, Figure 2).

Table 1: Angular values for each joint

joint(axis)	base(y)	base(x)	arm(x)	head(y)	head(x)
hold&fold	0	$\frac{\pi}{4}$	$-\frac{7}{8}\pi$	0	$\frac{\pi}{2}$
stretch	0	$\frac{\pi}{16}$	$-\frac{\pi}{2}$	0	0

4. 심층 강화 학습

강화학습의 문제는 마르코프 결정과정(Markov Decision Process, MDP)으로 표현된다. 환경(environment)과 에이전트가 놓여있을 때, 에이전트는 환경의 현재 상태(state)를 관찰한 값들을 얻어 행동(action)을 취하고, 환경으로부터 해당 행동이 얼마나 바람직하였는지를 보상(reward)의 형태로 피드백 받는다. 강화학습에서는

미리 정의된 입출력 쌍이 존재하지 않기 때문에 MDP로 공식화된 문제 공간을 탐색하며 한 에피소드 내에서 최적의 누적 보상을 받는 행동 정책 $\pi_{\theta^*}(a|s)$ 을 학습한다.

$$\theta^* = \operatorname{argmax}_{\theta} E_{s_{t+1} \sim p(s_{t+1}|s_t, a_t), a_t \sim \pi_{\theta}(a_t|s_t)} [\sum_t \gamma^t r_t]$$

물리 시뮬레이션이 이루어지고 있는 환경 안에서 s_t , a_t , r_t 은 각각 시뮬레이션 시간 t에서의 현재 상태, 행동 그리고 그에 따른 보상을 의미하고, $p(s_{t+1}|s_t, a_t)$ 는 주어진 상태와 행동에 따라 다음 상태 값을 결정하는 한 단계의 물리 시뮬레이션을, $\gamma \in [0, 1]$ 은 할인율(discount factor)을 의미한다.

이 논문에서 쓰인 PPO(Proximal Policy Optimization) [14] 알고리즘은 MDP 문제를 푸는 방법들 중 하나로, 환경과 에이전트가 환경으로부터 받아낸 데이터를 저장해 샘플링하여 정책 네트워크(policy network)와 가치 네트워크(value network)를 지속적으로 업데이트 한다. 네트워크는 TD(γ) 함수 [23]가 반환하는 값들을 통해 업데이트 되고, 일반화된 어드밴티지 추정, 즉 GAE(λ) 방식을 통해 보상의 손실과 분산을 줄여 안정적으로 에이전트를 학습시킨다 [24].

정책 네트워크의 입력과 출력은 각각 상태와 행동이다. 먼저 입력인 관찰값 s 는 발판의 xz 회전(2), 땅에서 발판까지의 수직 높이(1), 내부 관절의 각도(5), 속도(11), 목표 위치(2), 페이즈(2)까지 총 23개의 값으로 이루어져 있다. xz 회전(2)은 발판이 x축과 z축을 기준으로 얼마나 기울어져 있는지를 나타내며 이는 Luxo가 균형을 잡고 서 있는지에 대한 간접적인 지표이기도 하다. xz 회전과 발판의 속도(6), 목표 위치는 모두 현재 Luxo의 위치에서 수평 발판 좌표계를 기준으로 구한 값들이며, 목표 위치와 페이즈를 제외한 모든 관찰값들은 현재 시뮬레이션되고 있는 Luxo의 상태로부터 가져온다. 수평 발판 좌표계는 모델의 발판이 xz평면과 수평하고 y축 높이가 0이라고 가정하고 구한 Luxo의 발판의 지역 좌표계이다.

참조용 모션이 주어지는 모방 학습에서는 주어진 페이즈(phase)에 대해 해당하는 참조 포즈를 모방하는 적절한 행동을 선택하도록 학습한다. 페이즈 θ 는 한 번의 참조 점프 동작 사이클(1초) 동안 0에서 시작하여 2π 까지의 값을 가지는데, 페이즈의 연속적인 변화가 연속적인 상태 입력이 되도록 $\sin(\theta)$ 과 $\cos(\theta)$, 두 개의 주기 함수로 계산된 값을 관찰값으로 하였다. 이렇게 구해진 23개의 관찰값들을 y축에 대한 발판의 수직 높이를 제외하고 모두 $[-1, 1]$ 사이의 값으로 정규화하였다.

네트워크의 출력 a 는 Luxo가 특정 관찰값이 주어졌을 때 확률적으로 취하게 되는 행동으로, PD 컨트롤(Proportional-Derivative control)의 타겟 포즈에 해당하는 Luxo 내부 관절의 각도(5)이다. 해당 값(radian)은 범위에 제한이 없어 어떤 값이든 가질 수 있지만 처음에 출력되는 값들이 1보다 약간 작거나 큰 값들이기 때문에 학습이 진행되면서 적절한 크기로 조정된다.

Luxo의 강화학습 모델은 3가지 조건 아래 양수의 보상을 주는

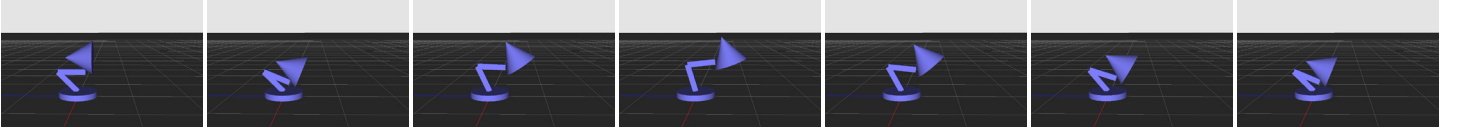


Figure 2: The reference motion of Luxo

방식으로 되어있고, 학습의 효율성을 위해 캐릭터가 넘어진다면 더이상 에피소드를 진행하지 않고 종료하는 기법(early termination)을 사용하였다 [3].

첫번째 보상 r_{imi} 은 시뮬레이션 동작이 참조 모션과 비슷하고 균형을 잘 잡고 있을수록 높은 보상(최대 1) 값을 가진다. 우선 선형 보간법으로 생성된 참조용 모션과 시뮬레이션 된 모션의 각 자유도의 인덱스 i 에 대한 각도 차이 $a_{\text{f}}^i - a^i$ 를 계산하고, 발판의 경우 항상 바닥과 평행, 즉 발판의 x, z 회전값 x_0, z_0 값이 0을 유지하는 것이 이상적이므로 둘을 더하여 그 합이 작을 수록 높은 보상을 준다.

$$r_{\text{imi}} = \exp\left(-\left(\sum_{i=1}^5 (a_{\text{f}}^i - a^i)^2 + (x_0)^2 + (z_0)^2\right)\right)$$

두번째 보상 r_{upright} 은 수평 발판 좌표계에 대한 시뮬레이션 동작의 xz 평면 상의 머리 위치 h 가 수평 발판 좌표계에 대한 참조 모션의 xz 평면 상의 머리 위치 h_{r} 와 같을수록 높은 보상(최대 1)을 준다.

$$r_{\text{upright}} = \exp(-(\text{abs}(h_{\text{r}} - h)))$$

마지막 보상 r_{obj} 은 목표 달성량을 나타낸다. Luxo의 목표는 환경 안에서 정해진 xz 평면 위의 목표 지점을 향해 뛰어가는 것이다. 그러므로 xz 평면 위에서의 목표 위치 $goal_{x,z}$ 와 현재 발판 위치의 값 $pos_{x,z}$ 의 차이를 구하여 작을 수록 높은 보상(최대 1)을 준다.

$$r_{\text{obj}} = \exp(-(pos_{x,z} - goal_{x,z})^2 * (pos_{x,z} - goal_{x,z})^2)$$

최종적으로 모델이 한 번의 스텝에서 받을 수 있는 보상 값 r 은 다음과 같다.

$$r = w_{\text{imi}} * r_{\text{imi}} + w_{\text{upright}} * r_{\text{upright}} + w_{\text{obj}} * r_{\text{obj}}$$

보상값들의 총합에서 $w_{\text{imi}}, w_{\text{upright}}, w_{\text{obj}}$ 는 각 항의 비중을 나타내며 각각 20, 1, 40으로 설정해주었을 때 가장 자연스러운 결과를 보였다(Figure 5).

모든 보상 계산식에서 보상 값의 크기를 차이 값이 작을수록 큰 값을 갖는 0보다 크고 1보다 작은 양수로 조정하기 위해 기존의 연구에서 보였듯이 [3] 지수 함수를 사용하였다.

5. 실험 결과

학습은 stable baselines 라이브러리 [25]의 PPO2 모듈을 이용하였다. PPO1과 달리 여러 개의 환경을 생성해 동시에 학습을 시킬 수 있으며 네트워크 구성은 다음과 같다(Figure 3).

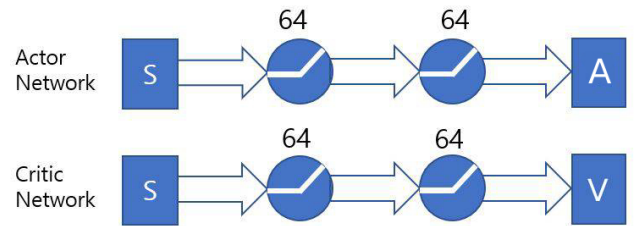


Figure 3: PPO2 network structure of stable baselines library. Given state S as an input, the actor network produces action A and the critic network produces expected value V.

학습 시의 $\gamma=0.99$, learning rate=0.00025, clip range=0.2로 모두 stable baselines 라이브러리의 기본 값을 사용하였으며, 네트워크가 업데이트 될 때마다 샘플링 되는 minibatch의 갯수는 128개로 증명하였다. 모든 실험은 16개의 CPU로 16개의 Luxo 환경들을 만들어 계산하였으며 시뮬레이션은 0.001초마다 진행되고 0.02초마다 Luxo가 목표하는 다음 관절 값이 출력을 통해 결정된다. Luxo의 에피소드는 다음과 같이 진행된다. 현재 위치를 기준으로 반경 5m인 반원을 그려 그 경계의 랜덤한 위치 하나를 고른 후 만일 Luxo 발판의 중심이 목표 위치의 0.5m 근처까지 도달한다면 그 위치를 기준으로 반경 5m인 반원을 그려 새로운 목표 위치를 설정하고, 그 전에 넘어진다면 곧바로 에피소드를 종료하는 기법을 사용한다. 목표 위치는 사용자가 알아보기 쉽게 하기 위해 붉은 큐브로 표기하였다(Figure 4).

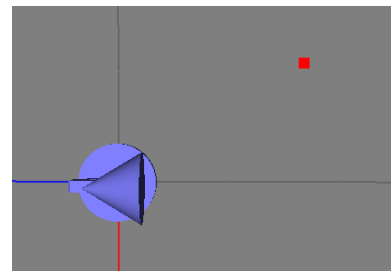


Figure 4: Luxo on xz plane

이 논문에서는 총 3가지의 실험을 진행하였다. 첫번째 실험은 에피소드가 시작할 때 목표를 랜덤하게 정하고 일정 거리(0.5m)에 다다를 때마다 다시 랜덤하게 목표를 업데이트 해주는 실험으로 이 논문에서 제안하는 방법의 성능을 보여준다(Figure 5). 모델의 학습에는 다섯 시간에서 여섯 시간이 걸렸다. 실험에서 Luxo는 목표 위치가 Luxo가 바라보는 방향, 즉 앞을 향해 설정된다면 목표 위치에 쉽게 도달하였다. 그렇지만 목표 위치가 오른쪽 또는 왼쪽으로 크게 치우쳐져 있을 때는 급격히 방향을 전환하고자 몸체가 크게 돌아 종종 실패하는 모습을 보이기도 한다. 이는 제공된 참조 동작이 앞으로 향하는 것 밖에 없기 때문에, 학습된 정책이 Luxo가 그 외의 방향을 향할 때 상대적으로 자연스럽게 못한 동작을 출력하는 것으로 생각된다. 또한 커리큘럼 학습을 통해 모방 학습을 충분히 시킨 후 해당 동작을 기반으로 목표 위치를 찾도록 학습시켰다면 조금 더 효율적인 학습이 가능했을 것이다.

두번째 실험에서는 Luxo가 넘어지면 곧바로 에피소드를 종료하는 기법을 적용하지 않고 모델을 학습시켰다(Figure 6). 모델의 첫번째 실험보다 짧은 두 시간 정도의 학습 후 훨씬 적은 누적 보상값에서 수렴하였으며 목표 위치에 전혀 다다르지 못하고 시작과 동시에 넘어져 단순히 참조 동작을 모방하는데 그쳤다. 이 결과는 정책이 최적의 보상값에 수렴하기 위해서 Luxo가 수직으로 선 상태에서 점프 모션을 모방해야 하는데, 학습되는 초기에 Luxo가 쉽게 넘어지면서 넘어진 이후의 행동들이 누적된 학습 정보의 대부분을 차지하기 때문인 것으로 생각된다.

세번째 실험에서는 참조용 모션을 생성하지 않고 모델을 학습시켰다(Figure 7). 모델의 누적 보상값은 선행된 실험들만큼의 시간이 지나도 특정 값에 금방 수렴하지 못하고 크게 진동하였는데, 이는 학습의 가이드가 되는 참조 모션이 존재하지 않기 때문에 학습이 잘 되지 않은 것으로 추측할 수 있다. 세번째 실험의 시뮬레이션에서 Luxo는 다른 두 실험들에 비해 높이 뛴 후 그대로 뒤로 넘어지는 모습을 보이며, 충분히 학습된 결과를 보이지 않았다.

6. 결론

이 논문에서는 한 개의 다리를 가진 Luxo 캐릭터가 자연스럽게 이동하도록 하는 심층 강화학습 모델을 제안한다. 애니메이션의 점프 동작처럼 뛰게 하기 위해서 선형 보간법으로 생성한 참조 모션을 사용하였으며, Luxo는 PD제어법과 심층 강화학습을 통해 이를 모방하며 임의로 지정된 위치로 이동하는 정책을 학습하였다. 이렇게 학습된 심층 강화학습 모델을 통해 Luxo가 자연스러운 동작으로 뛰어다니는 애니메이션을 생성할 수 있다.

향후 연구에서는 Luxo가 더 역동적으로 뛰는 모습을 학습시키고자 한다. 참조 동작을 생성할 때에 있어서 선형 보간법 외에 베지에 보간법처럼 더 부드러운 보간법을 사용한다면 더 부드럽고 자연스러운 참조 모션을 만들 수 있을 것이다. 또한 앞서 실험 결과(단락 5)에서 언급한 바와 같이, 앞으로 전진하는 참조

동작 외에 다른 모션들을 추가한다면 정방향으로 향하는 동작만 주어진 현재보다 방향을 전환하는 동작이 더 자연스럽게 학습될 것으로 기대된다. 네트워크 학습에 있어서는 페이즈(phase)를 네트워크 출력값으로 받아 유동적으로 조정하여 더 효율적인 학습을 시키거나 커리큘럼 학습을 통해 학습의 진척도에 따라 보상 체계를 조금씩 조정하는 방법을 시도해 볼 수 있을 것이다. 처음에는 모방 보상(r_{imi})만으로 학습을 시키다가, 점차 전체 보상의 크기에서 목적 보상(r_{obj})의 비율을 늘리면 원하는 동작을 모방하면서 조금 더 효율적으로 목표 위치에 도달하도록 학습시킬 수 있다.

감사의 글

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 SW중심대학지원사업의 연구결과로 수행되었으며 (2016-0-00023), 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구이며 (No. 2019R1C1C1006778, NRF-2019R1A4A1029800), 과학기술정보통신부 및 정보통신산업진흥원의 ‘고성능 컴퓨팅 지원’ 사업으로부터 지원받아 수행하였음.

References

- [1] K. Yin, K. Loken, and M. van de Panne, “Simbicon: Simple biped locomotion control,” *ACM Trans. Graph.*, vol. 26, no. 3, p. Article 105, 2007.
- [2] Y. Lee, S. Kim, and J. Lee, “Data-driven biped control,” in *ACM SIGGRAPH 2010 Papers*, ser. SIGGRAPH '10. New York, NY, USA: Association for Computing Machinery, 2010. [Online]. Available: <https://doi.org/10.1145/1833349.1781155>
- [3] X. B. Peng, P. Abbeel, S. Levine, and M. van de Panne, “Deepmimic: Example-guided deep reinforcement learning of physics-based character skills,” *ACM Trans. Graph.*, vol. 37, no. 4, pp. 143:1–143:14, July 2018. [Online]. Available: <http://doi.acm.org/10.1145/3197517.3201311>
- [4] J. Z. Kolter, P. Abbeel, and A. Y. Ng, “Hierarchical apprenticeship learning, with application to quadruped locomotion,” in *Proceedings of the 20th International Conference on Neural Information Processing Systems*, ser. NIPS'07. Red Hook, NY, USA: Curran Associates Inc., 2007, p. 769–776.
- [5] P. Abbeel, A. Coates, and A. Ng, “Autonomous helicopter aerobatics through apprenticeship learning,” *I. J. Robotic Res.*, vol. 29, pp. 1608–1639, 11 2010.

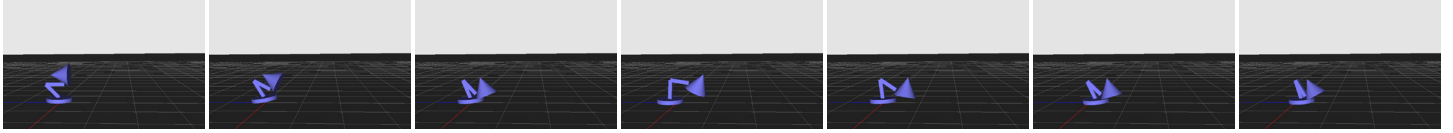


Figure 5: Simulated Luxo motion based on the learned policy

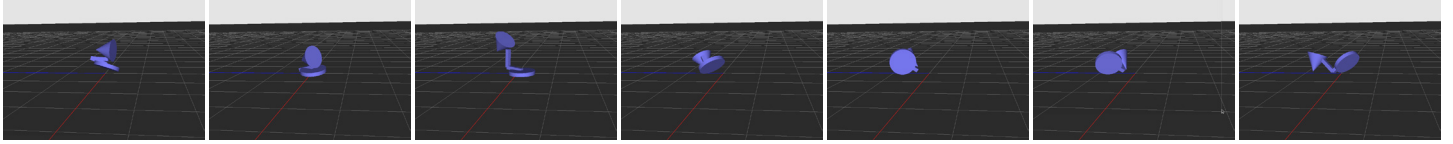


Figure 6: Simulated Luxo motion based on the policy learned without early termination

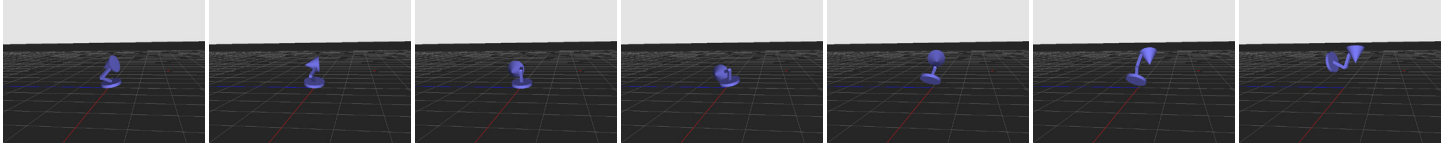


Figure 7: Simulated Luxo motion based on the policy learned without reference motion

- [6] N. M. O. Heess, T. Dhruva, S. Sriram, J. Lemmon, J. Merel, G. Wayne, Y. Tassa, T. Erez, Z. Wang, S. M. A. Eslami, M. A. Riedmiller, and D. Silver, “Emergence of locomotion behaviours in rich environments,” *ArXiv*, vol. abs/1707.02286, 2017.
- [7] S. Park, H. Ryu, S. Lee, S. Lee, and J. Lee, “Learning predict-and-simulate policies from unorganized human motion data,” *ACM Trans. Graph.*, vol. 38, no. 6, 2019.
- [8] M. de Lasa, I. Mordatch, and A. Hertzmann, “Feature-based locomotion controllers,” *ACM Trans. Graph.*, vol. 29, no. 4, July 2010. [Online]. Available: <https://doi.org/10.1145/1778765.1781157>
- [9] S. Agrawal and M. van de Panne, “Task-based locomotion,” *ACM Transactions on Graphics (Proc. SIGGRAPH 2016)*, vol. 35, no. 4, 2016.
- [10] J. M. Wang, D. J. Fleet, and A. Hertzmann, “Optimizing walking controllers for uncertain inputs and environments,” in *ACM SIGGRAPH 2010 Papers*, ser. SIGGRAPH ’10. New York, NY, USA: Association for Computing Machinery, 2010. [Online]. Available: <https://doi.org/10.1145/1833349.1778810>
- [11] K. Wampler, Z. Popoviundefined, and J. Popoviundefined, “Generalizing locomotion style to new animals with inverse optimal regression,” *ACM Trans. Graph.*, vol. 33, no. 4, July 2014. [Online]. Available: <https://doi.org/10.1145/2601097.2601192>
- [12] R. J. Williams, “Simple statistical gradient-following algorithms for connectionist reinforcement learning,” *Mach. Learn.*, vol. 8, no. 3–4, p. 229–256, May 1992. [Online]. Available: <https://doi.org/10.1007/BF00992696>
- [13] J. Schulman, S. Levine, P. Moritz, M. Jordan, and P. Abbeel, “Trust region policy optimization,” in *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ser. ICML’15. JMLR.org, 2015, p. 1889–1897.
- [14] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” 2017.
- [15] X. B. Peng and M. van de Panne, “Learning locomotion skills using deeprl: Does the choice of action space matter?” in *Proceedings of the ACM SIGGRAPH / Eurographics Symposium on Computer Animation*, ser. SCA ’17. New York, NY, USA: Association for Computing Machinery, 2017. [Online]. Available: <https://doi.org/10.1145/3099564.3099567>
- [16] W. Yu, G. Turk, and C. K. Liu, “Learning symmetry and low-energy locomotion,” *CoRR*, vol. abs/1801.08093, 2018. [Online]. Available: <http://arxiv.org/abs/1801.08093>
- [17] X. B. Peng, G. Berseth, K. Yin, and M. Van De Panne, “Deeploco: Dynamic locomotion skills using hierarchical deep reinforcement learning,” *ACM Trans. Graph.*, vol. 36, no. 4, July 2017. [Online]. Available: <https://doi.org/10.1145/3072959.3073602>
- [18] J. Won, J. Park, and J. Lee, “Aerobatics control of flying creatures via self-regulated learning,” *ACM Trans. Graph.*, vol. 37, no. 6, Dec. 2018. [Online]. Available: <https://doi.org/10.1145/3272127.3275023>

- [19] A. Witkin and M. Kass, “Spacetime constraints,” *SIGGRAPH Comput. Graph.*, vol. 22, no. 4, p. 159–168, June 1988. [Online]. Available: <https://doi.org/10.1145/378456.378507>
- [20] K. Yamane, Y. Ariki, and J. Hodgins, “Animating non-humanoid characters with human motion data,” in *Proceedings of the 2010 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, ser. SCA ’10. Goslar, DEU: Eurographics Association, 2010, p. 169–178.
- [21] A. Sharma and K. M. Kitani, “Phase-parametric policies for reinforcement learning in cyclic environments,” in *AAAI*, 2018.
- [22] T. Kwon, Y. Lee, and M. van de Panne, “Fast and flexible multilegged locomotion using learned centroidal dynamics,” *ACM Trans. Graph.*, 2020. [Online]. Available: <http://calab.hanyang.ac.kr/papers/flexLoco.html>
- [23] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*, ser. Adaptive Computation and Machine Learning series. MIT Press, 1998. [Online]. Available: <https://books.google.co.kr/books?id=6DKPtQEACAAJ>
- [24] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, “High-dimensional continuous control using generalized advantage estimation,” 2015.
- [25] “stable baselines,” <https://github.com/hill-a/stable-baselines>, accessed: 2020-03-10.
- [26] S. Coros, P. Beaudoin, and M. van de Panne, “Generalized biped walking control,” *ACM Transactions on Graphics*, vol. 29, no. 4, p. Article 130, 2010.
- [27] I. Mordatch, E. Todorov, and Z. Popoviundefined, “Discovery of complex behaviors through contact-invariant optimization,” *ACM Trans. Graph.*, vol. 31, no. 4, July 2012. [Online]. Available: <https://doi.org/10.1145/2185520.2185539>
- [28] J. Tan, K. Liu, and G. Turk, “Stable proportional-derivative controllers,” *IEEE Comput. Graph. Appl.*, vol. 31, no. 4, p. 34–44, July 2011. [Online]. Available: <https://doi.org/10.1109/MCG.2011.30>
- [29] A. Rajeswaran, V. Kumar, A. Gupta, J. Schulman, E. Todorov, and S. Levine, “Learning complex dexterous manipulation with deep reinforcement learning and demonstrations,” *CoRR*, vol. abs/1709.10087, 2017. [Online]. Available: <http://arxiv.org/abs/1709.10087>
- [30] Y. Lee, M. S. Park, T. Kwon, and J. Lee, “Locomotion control for many-muscle humanoids,” *ACM Trans. Graph.*, vol. 33, no. 6, Nov. 2014. [Online]. Available: <https://doi.org/10.1145/2661229.2661233>
- [31] D. Sharon and M. van de Panne, “Synthesis of controllers for stylized planar bipedal walking,” in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, 2005, pp. 2387–2392.
- [32] K. Bergamin, S. Clavet, D. Holden, and J. R. Forbes, “Drecon: Data-driven responsive control of physics-based characters,” *ACM Trans. Graph.*, vol. 38, no. 6, Nov. 2019. [Online]. Available: <https://doi.org/10.1145/3355089.3356536>
- [33] K. Lee, S. Lee, and J. Lee, “Interactive character animation by learning multi-objective control,” *ACM Trans. Graph.*, vol. 37, no. 6, Dec. 2018. [Online]. Available: <https://doi.org/10.1145/3272127.3275071>

〈 저 자 소 개 〉



이 정 민

- 2016-2020 한양대학교 컴퓨터공학부 학사
- 2020-현재 한양대학교 컴퓨터소프트웨어학부 석사
- 관심분야: Physically-Based Character Control
- <https://orcid.org/0000-0002-7774-1427>



이 윤 상

- 1999-2007 서울대학교 기계항공공학부 학사
- 2007-2014 서울대학교 컴퓨터공학부 박사
- 2014-2016 삼성전자 소프트웨어센터 책임연구원
- 2016-2018 광운대학교 소프트웨어학부 조교수
- 2018-현재 한양대학교 컴퓨터소프트웨어학부 조교수
- 관심분야: Physically-Based Character Control, Robot Control Algorithm, Computational Design
- <https://orcid.org/0000-0002-0579-5987>