MoAI 26 Fall · 5회 humanoid

Humanoid

Whole-Body Control & Motion Tracking

휴머노이드는 몸 전체를 어떻게 제어하고,
사람의 움직임을 어떻게 따라 하는가

발표 윤서현 · 참고 논문 DeepMimic (Peng et al., SIGGRAPH 2018) 외



주황 점 = 따라가야 할 관절(마커)

Roadmap: Background + Three Questions

00

Background

로봇의 몸 · PD 제어 · 강화학습 · 시뮬레이션 · 모션 캡처

Q1

What Is a Humanoid?

떠 있는 몸 · 좁은 지지면 · 많은 관절

Q2

Whole-Body Control

팔·다리·몸통을 하나의 시스템으로 — 고전 최적화에서 학습 정책으로

Q3

Motion Tracking

사람 모션을 매 순간 따라가기 — DeepMimic에서 범용 트래커까지

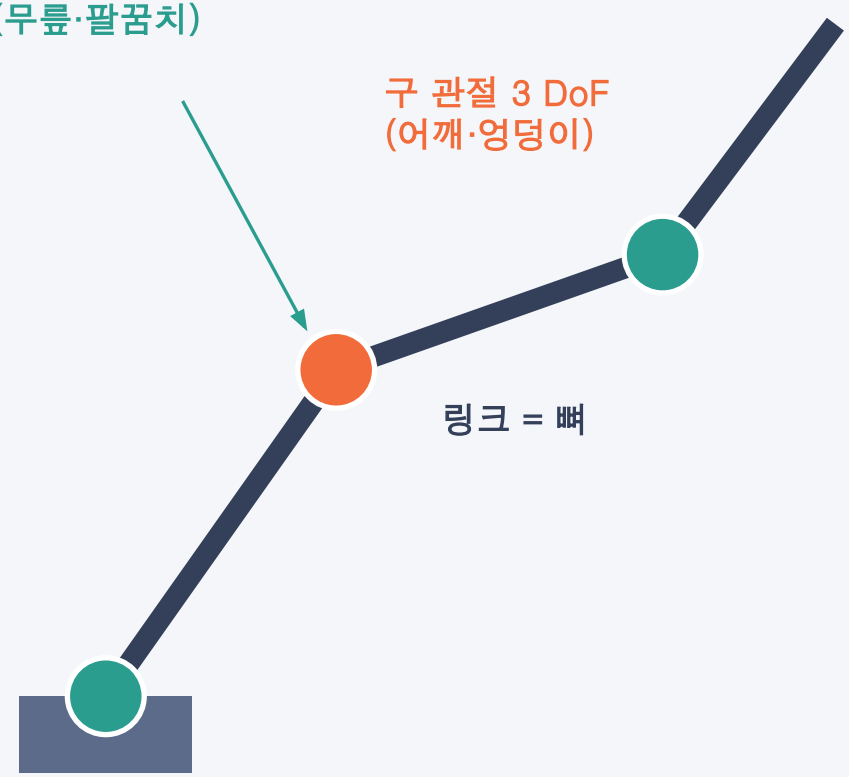
참고 논문: DeepMimic (2018) 중심 · 이후 흐름: PHC, ExBody, OmniH2O, HumanPlus, GMT, BeyondMimic, ASAP

00

Background

시작 전에: 로봇의 몸 · 모터 제어 · 강화학습 · 시뮬레이션 · 모션 캡처

Robot Body 101: Links, Joints, DoF



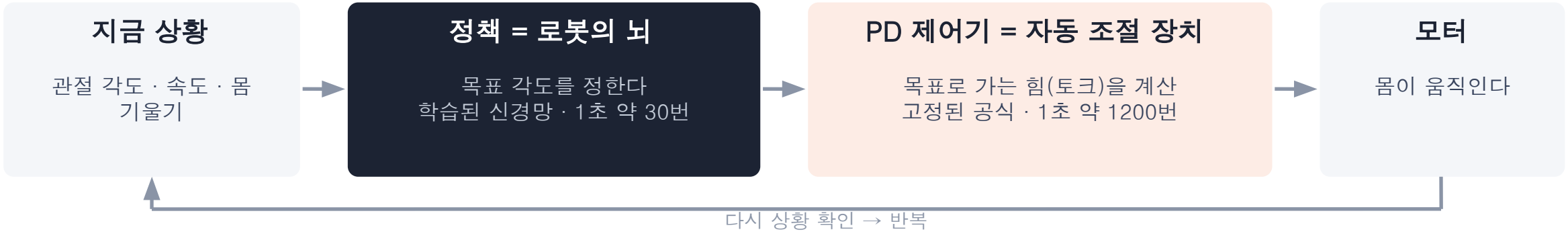
개념도 · 로봇 팔 하나

링크 (link)	로봇의 뼈. 단단해서 휘지 않는 부품
관절 (joint)	링크와 링크를 잇는 곳. 여기에 모터가 달린다
자유도 (DoF)	독립적으로 움직일 수 있는 방향의 수. 무릎은 1, 어깨는 3
관절 각도 q	로봇의 자세 = 모든 관절 각도를 모은 벡터

예: DeepMimic 휴머노이드 34 DoF = 관절 28 + 땅에 고정되지 않은 몸통 6 (위치 3 + 방향 3)

일반 로봇공학 개념 · 그림은 발표자 개념도

From Command to Motion: Policy & PD Control



PD 제어기가 쓰는 공식

$$\tau = kp \cdot (\hat{q} - q) - kd \cdot \dot{q}$$

τ	모터에 줄 힘 (토크)
$\hat{q} - q$	목표 각도 − 지금 각도 → 멀수록 세게 민다 (kp = 용수철 세기)
$\dot{q}$	지금 도는 속도 → 너무 빠르면 브레이크 (kd = 브레이크 세기)

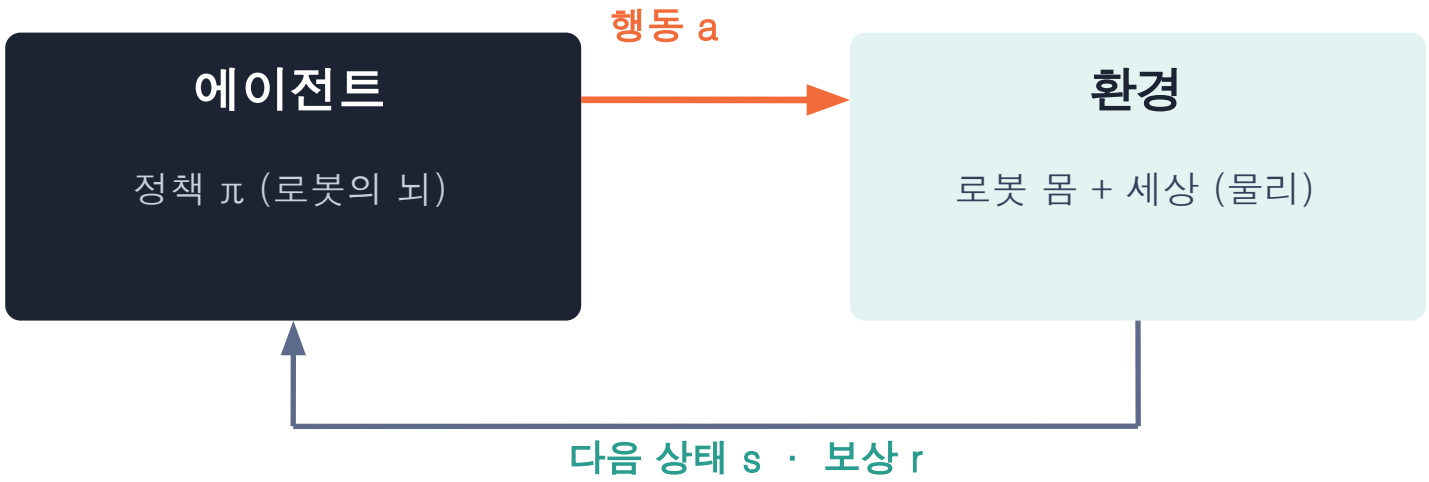
비유: 에어컨

‘26도로 맞춰’를 정하는 것 = 정책
지금 온도를 보고 바람 세기를 조절하는 것 = PD 제어기

정책은 ‘어디로’만, 힘 조절은 PD가
→ 신경망이 배우기 쉬워진다

일반 제어 개념 · 30Hz / 1.2kHz는 DeepMimic(Peng et al. 2018) 설정

Reinforcement Learning in One Slide



상태 s	지금 상황 (관절 각도, 속도, 몸 기울기…)
행동 a	정책이 내리는 명령 (목표 관절 각도)
보상 r	방금 행동이 얼마나 좋았나를 숫자로
에피소드	시작부터 끝(넘어짐·시간 종료)까지 한 판

목표

한 판 동안 받는 보상의 합을 최대로 만드는 정책 π를 찾는다

비유: 강아지 훈련

정답을 알려주지 않는다. 해 보고, 잘하면 간식(보상). 수없이 반복하며 무엇이 간식으로 이어지는지 스스로 알아낸다

일반 강화학습 개념 (Sutton & Barto)

Policy = Neural Network, Trained by PPO

정책 π = 작은 신경망 (MLP)



층 크기는 예시 · 논문마다 다르다

1 해 본다

시뮬레이션에서 정책대로 수천 번 움직여 데이터를 모은다

2 기대보다 좋았나?

가치 함수 V(s)가 '이 상태라면 보통 이 정도 보상'을 예측 → 그보다 좋았던 행동은 확률↑, 나빴던 행동은 확률↓

3 조금씩만 바꾼다

PPO는 한 번에 정책을 너무 많이 바꾸지 않게 제한 → 학습이 안정적. 로봇 RL의 사실상 표준

Schulman et al. 2017 (PPO) · 개념 설명

Why Simulation? And the Sim-to-Real Gap

넘어져도 괜찮다

실제 로봇은 넘어지면 망가진다.
시뮬레이터에선 수백만 번 넘어져도 된다

수천 대를 동시에

GPU 위에서 로봇 수천 대를 병렬로
→ 수억 스텝을 하루 안에

정답을 안다

마찰·지형·정확한 위치 같은 '특권 정보'를 시뮬레이터는 알고 있다

문제: 시뮬레이터는 현실과 다르다 (마찰, 모터 특성, 센서 노이즈, 지연…) → Sim-to-Real Gap

1 해법 ① 도메인 무작위화

마찰·무게·모터 세기를 매번 무작위로 바꿔 학습 → 현실이 그중 하나이길 바란다

2 해법 ② 교사 → 학생

특권 정보를 쓰는 교사 정책을 먼저 만들고, 실제 센서만 쓰는 학생이 교사를 따라 하게 학습

3회 locomotion 발표(RMA · Perceptive Locomotion)에서 다룬 내용 요약

Motion Capture: Where Reference Motions Come From

사람이 움직인다

몸에 마커를 붙이고 여러 카메라로 촬영
(또는 일반 영상에서 자세 추정)

→

자세의 시계열

프레임마다 관절 위치·각도
예: 1초에 30~120 프레임

→

로봇 몸에 맞게 변환

사람과 로봇은 키·관절이 다르다
→ 리타게팅 (다음 발표)

→

참조 모션

로봇이 '따라가야 할 정답 동작'

시간 (s)	엉덩이	무릎	발목	…
0.000	0.12	0.35	−0.08	…
0.033	0.15	0.41	−0.10	…
0.067	0.19	0.48	−0.11	…

참조 모션의 모양 (관절 각도, rad · 가상 예시)

왜 중요한가

'자연스럽게 걸어라'를 보상 식으로 쓰긴 어렵지만,
사람의 실제 동작은 데이터로 얼마든지 있다 →
이걸 정답지로 쓰자 = tracking

일반 개념 · 공개 데이터 예: CMU Mocap, AMASS · 표는 발표자 가상 예시

01

What Is a Humanoid?

4족 로봇과 무엇이 다른가, 그런데도 왜 사람 모양인가

Four Reasons Humanoid Control Is Hard

1 떠 있는 몸 (floating base)

몸통은 어디에도 고정되어 있지 않다. 몸통을 직접 움직이는 모터는 없고, 발이 땅을 미는 힘으로만 간접 제어
→ underactuated

2 좁은 지지면

두 발, 한 발로 서는 순간이 계속 온다. 무게중심이 높고 지지 다각형이 작아 조금만 어긋나도 넘어진다

3 많은 관절

다리·팔·허리·목까지 수십 개 자유도. 팔을 뻗으면 무게중심이 바뀌어 다리가 보정해야 한다

4 접촉이 계속 바뀐다

발이 닿고 떨어질 때마다 동역학이 바뀐다. 손이 물체를 잡으면 또 바뀐다

일반적인 배경 설명 · 1·4번은 3회 locomotion 발표(RMA 배경)와 같은 문제의식

Quadruped vs. Humanoid

	4족 (지난 locomotion 회차)	휴머노이드 (오늘)
땅에 닿는 다리	보통 2~3개, 지지면이 넓다	1~2개, 지지면이 좁다
무게중심	낮고 안정적	높고 넘어지기 쉽다
팔	없음 (보행만)	있음 → 보행 + 조작을 동시에
주로 내리는 명령	몸통 속도 (vx, vy, ω)	속도 + 손 위치 + 전신 자세
좋은 참고 데이터	동물 모션 (구하기 어렵다)	사람 모션 (공개 데이터 다수)

마지막 줄이 핵심: 사람 모양이라서 사람 모션 데이터를 그대로 '정답 동작'으로 쓸 수 있다 → tracking

일반적인 비교 · 3회 RMA·Perceptive Locomotion 발표자료와 연결

02

Whole-Body Control

온몸을 하나의 시스템으로 — 고전 최적화 vs 학습 정책

Whole-Body Control: Many Tasks, One Body

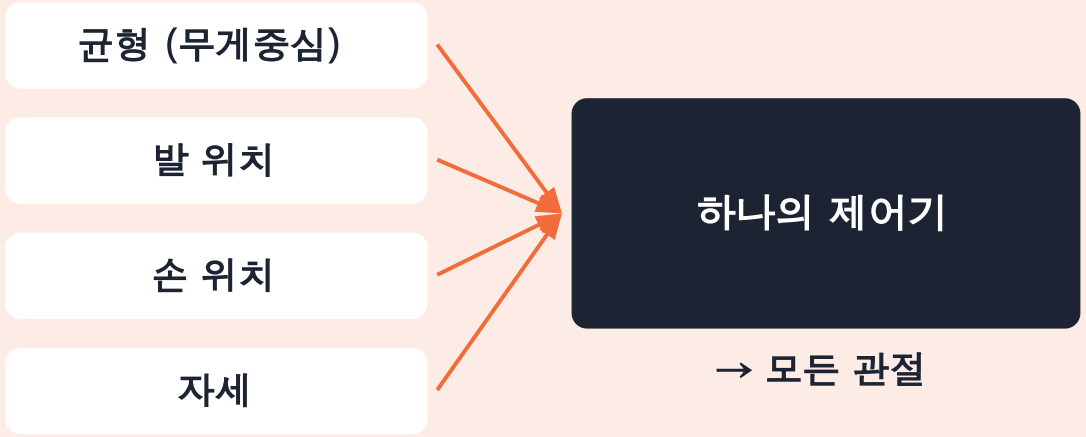
따로 제어

팔 제어기 → 손을 목표로

다리 제어기 → 걷기·균형

팔을 뻗으면 무게중심이 앞으로 → 다리 제어기는 이유를 모른 채 넘어진다

Whole-Body Control



과제끼리 부딪히면 우선순위로 조정: 넘어지지 않기 > 발 위치 > 손 위치 > 자세

핵심: '손을 저기에' 같은 과제를 관절 하나하나가 아니라 몸 전체가 같이 푼다

개념도 · 우선순위 예시는 발표자 구성 (Sentis & Khatib 2005의 계층적 제어 아이디어)

The Equation of Motion

$$M(q)\,\ddot{q} + h(q, \dot{q}) = S^{T}\,\tau + J_c{}^{T}\,f$$

$M(q)\,\ddot{q}$	관성 × 가속도. q는 몸통 위치·자세(6) + 관절 각도
$h(q, \dot{q})$	중력, 원심력·코리올리 힘
$S^{T}\,\tau$	모터 토크. S가 관절만 골라낸다 → 몸통 6자유도에는 토크가 없다
$J_c{}^{T}\,f$	발·손이 받는 접촉력 (지면 반력)

읽는 법

몸통을 움직이려면 모터로 '직접'은 안 되고, 발로 땅을 밀어 생긴 f를 거쳐야 한다

게다가 f는 땅을 당길 수 없고(밀기만), 미끄러지지 않을 만큼만 옆으로 낼 수 있다 → 마찰 원뿔 제약

부유 기저(floating-base) 로봇의 표준 동역학 표기 · 로봇공학 교재 일반

Classical WBC: Optimize at Every Instant

목표: 과제별 오차의 가중합을 최소로

$$\min \ \Sigma_i \, w_i \, \| \text{(과제 i의 실제 가속도)} - \text{(원하는 가속도)} \ \|^2$$

변수: 관절 가속도 $\ddot{q}$, 토크 τ, 접촉력 f

지켜야 할 제약

운동 방정식

앞 장의 식을 반드시 만족

마찰 원뿔

미끄러지지 않는 접촉력만

토크·관절 한계

모터가 낼 수 있는 범위 안

딛은 발은 고정

접촉점의 가속도 = 0

보통의 계층 구조

계획 (MPC 등)

무게중심·발 디딜 곳을 앞을 내다보며

↓

WBC (QP)

그 계획을 지금 순간의 관절 명령으로

↓

관절 토크

모터로

예: DARPA Robotics Challenge 시기 Atlas (Kuindersma et al., 2016)

Sentis & Khatib 2005 (계층적 WBC) · Kuindersma et al. 2016, Autonomous Robots (Atlas 최적화 기반 제어)

Classical WBC vs. Learned Whole-Body Policy

	고전 WBC (모델 + 최적화)	학습 기반 (시뮬레이션 RL)
필요한 것	정확한 동역학 모델, 접촉 시점 가정	시뮬레이터 + 보상 (또는 참고 모션)
제약 처리	식으로 명시 → 보장이 분명	보상·종료 조건으로 간접 유도
외란·넘어짐	모델 밖 상황에 약하다	무작위화로 다양하게 경험 → 튼튼
새 동작 추가	과제·가중치를 사람이 다시 설계	데이터(모션)를 바꾸고 다시 학습
실행 시 계산	매 순간 QP 풀기	신경망 한 번 통과 (빠르다)
해석 가능성	높다	낮다

요즘 휴머노이드 학습의 주류는 오른쪽: 정책이 전신 관절 목표를 내고 PD가 토크로 바꾼다 (RMA와 같은 구조)

일반적인 비교 · 발표자 정리

What Do We Ask the Policy? A Spectrum of Commands

속도 명령

앞·옆·회전 속도

조이스틱 보행 (4족 locomotion)

일부 키포인트

머리·양손 위치

VR 텔레오퍼레이션 (OmniH2O)

상체 모션 + 하체 속도

팔은 따라 하고 다리는 걷기만

ExBody

전신 참조 모션

모든 관절의 궤적

DeepMimic · GMT · BeyondMimic

명령이 담는 정보

더 풍부한 표현 · 더 어려운 학습

오른쪽 끝, '전신 참조 모션을 따라가라'가 바로 motion tracking

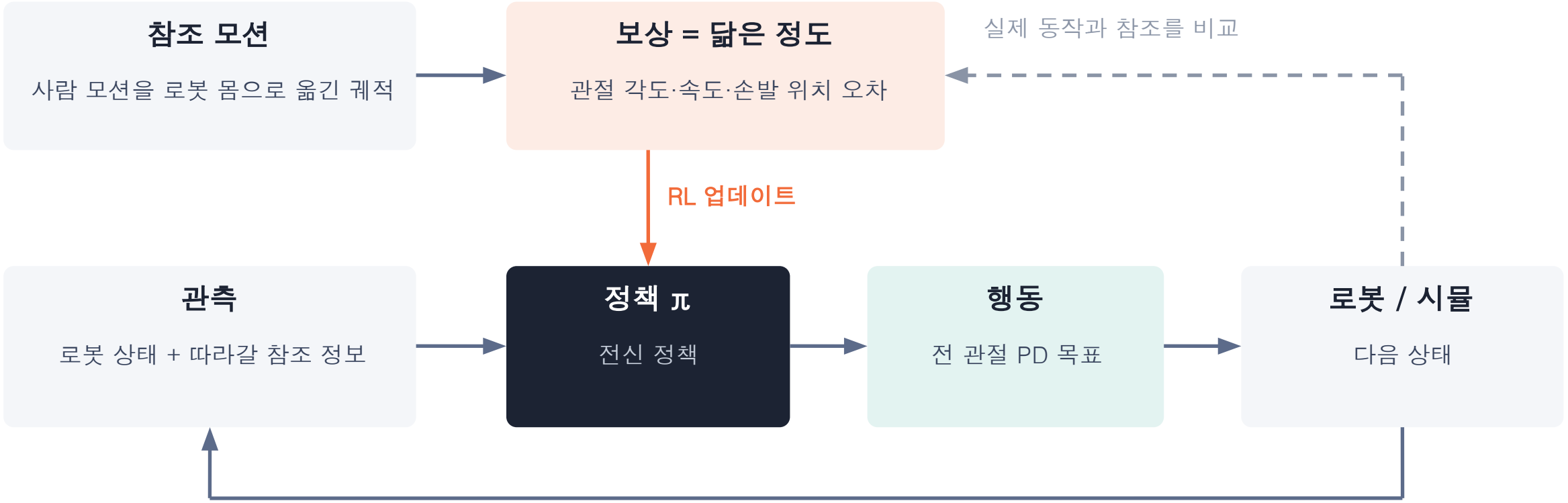
He et al. 2024 (OmniH2O, arXiv:2406.08858) · Cheng et al. 2024 (ExBody, arXiv:2402.16796) · Chen et al. 2025 (GMT) · Liao et al. 2025 (BeyondMimic)

03

Motion Tracking

DeepMimic이 세운 틀, 그리고 범용 트래커와 텔레오퍼레이션으로

Motion Tracking: Follow the Reference



장점: 동작마다 보상을 새로 짤 필요 없이 '이 모션을 닮아라' 하나로 걷기·춤·옆돌기를 모두 배운다

구조는 Peng et al. 2018 (DeepMimic) §3를 일반화한 발표자 개념도

DeepMimic (2018): Reward = Similarity

자세 pose 0.65	속도 0.10	손발 끝 0.15	무게중심 0.10

자세	exp[−2 · Σ 관절 회전 오차²]	k = 2	rad
속도	exp[−0.1 · Σ 관절 각속도 오차²]	k = 0.1	rad/s
손발 끝	exp[−40 · Σ 손발 위치 오차²]	k = 40	m
무게중심	exp[−10 · 무게중심 위치 오차²]	k = 10	m

모든 항이 exp(−k·오차)

오차 0이면 1, 커지면 0 → 항상 0~1이라 더하기 쉽다.
k가 클수록 엄격하다

최종 보상 = 0.7 × 모방 + 0.3 × 과제. 백플립·걷기·공룡까지 같은 식, 같은 가중치 — 바뀌는 건 모션 클립뿐

Peng et al., 2018, DeepMimic §5.3, §10

Intuition: Same 10 cm, Different Penalty

상황	오차 계산	보상	값
손발 4개가 각각 5cm 어긋남 (= 하나만 10cm)	$\Sigma = 4 \times 0.05^2 = 0.01$	$\exp(-40 \times 0.01)$	**0.67**
손발 4개가 각각 10cm 어긋남	$\Sigma = 4 \times 0.10^2 = 0.04$	$\exp(-40 \times 0.04)$	**0.20**
무게중심이 10cm 어긋남	$0.10^2 = 0.01$	$\exp(-10 \times 0.01)$	**0.90**
관절 회전 오차 제곱합 0.1	$\Sigma = 0.1\ \text{rad}^2$	$\exp(-2 \times 0.1)$	**0.82**
관절 회전 오차 제곱합 0.5	$\Sigma = 0.5\ \text{rad}^2$	$\exp(-2 \times 0.5)$	**0.37**

※ 표의 숫자는 이해를 돕기 위해 발표자가 만든 가상 예시이고, k 값만 DeepMimic 논문 값이다

읽는 법

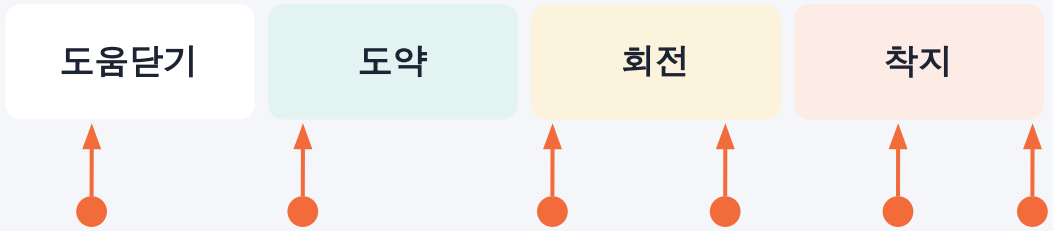
같은 10cm라도 손발 끝은 보상이 0.20까지 떨어지고, 무게중심은 0.90으로 거의 안 깎인다.

→ k 값이 '무엇을 정확히 따라 할지'를 정한다

k 값: Peng et al., 2018, §5.3 · 계산은 발표자

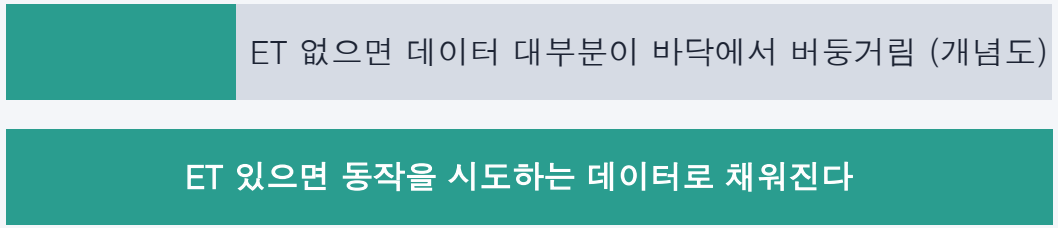
Two Tricks: RSI and Early Termination

RSI · 참조 동작의 아무 지점에서 시작



항상 처음부터 시작하면 착지를 못 배워 점프가 손해 → 공중 회전을 영영 못 만난다. 무작위 시점에서 시작하면 착지부터도 배운다

ET · 넘어지면 바로 끝낸다

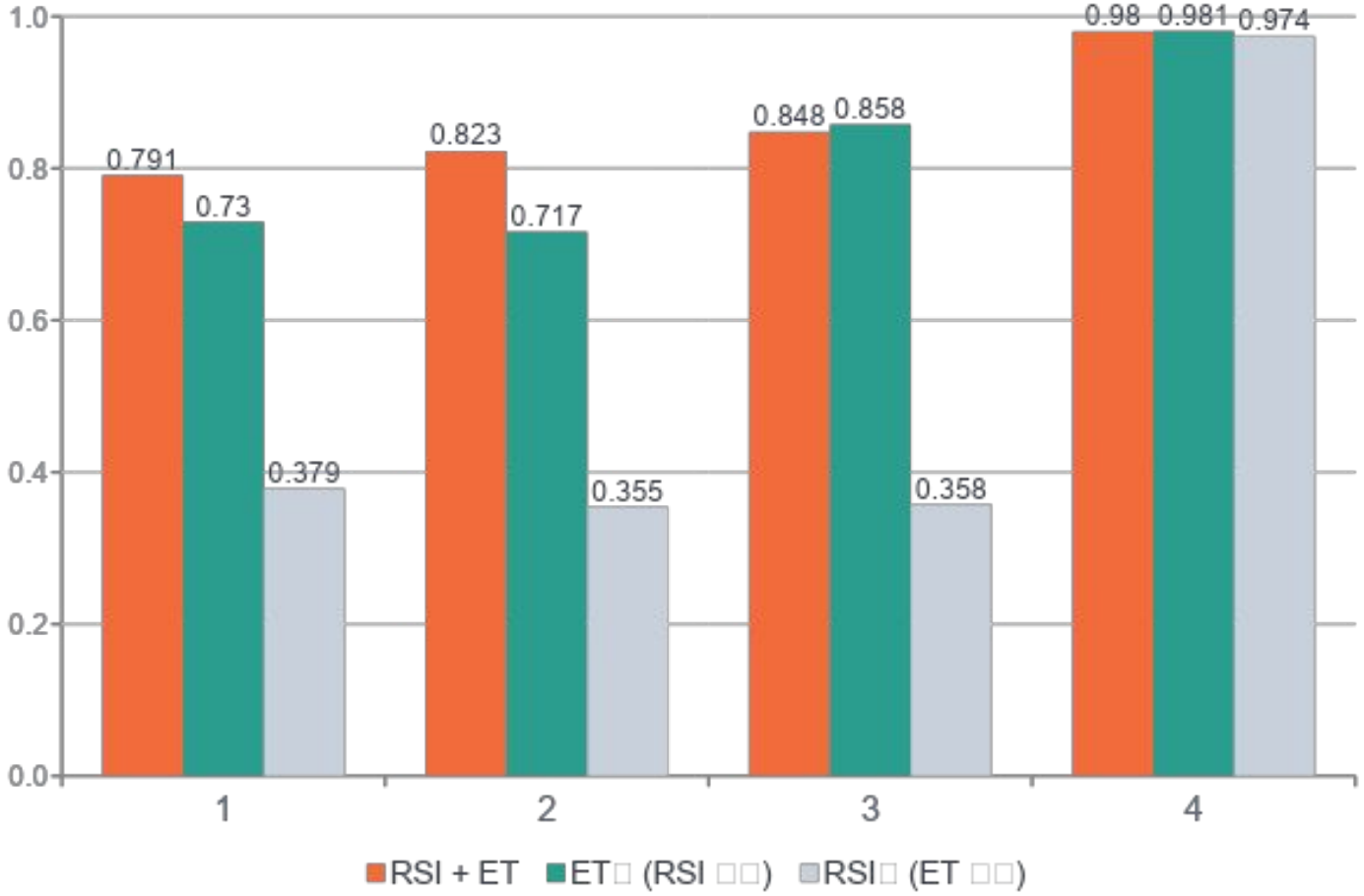


몸통·머리가 땅에 닿으면 종료, 남은 보상 0 → 넘어지지 않기가 강한 목표가 되고 쓸모없는 샘플이 사라진다

알고리즘(PPO)은 평범했다. 에피소드를 어디서 시작하고 언제 끝낼지가 고난도 동작의 성패를 갈랐다.
이후 트래킹 연구에서도 RSI·ET는 널리 쓰이는 출발점이 됐다

Peng et al., 2018, §6.1–6.2 · 그림은 발표자 개념도

Ablation: Remove One, and It Breaks



ET를 빼면: 0.36 근처로 붕괴

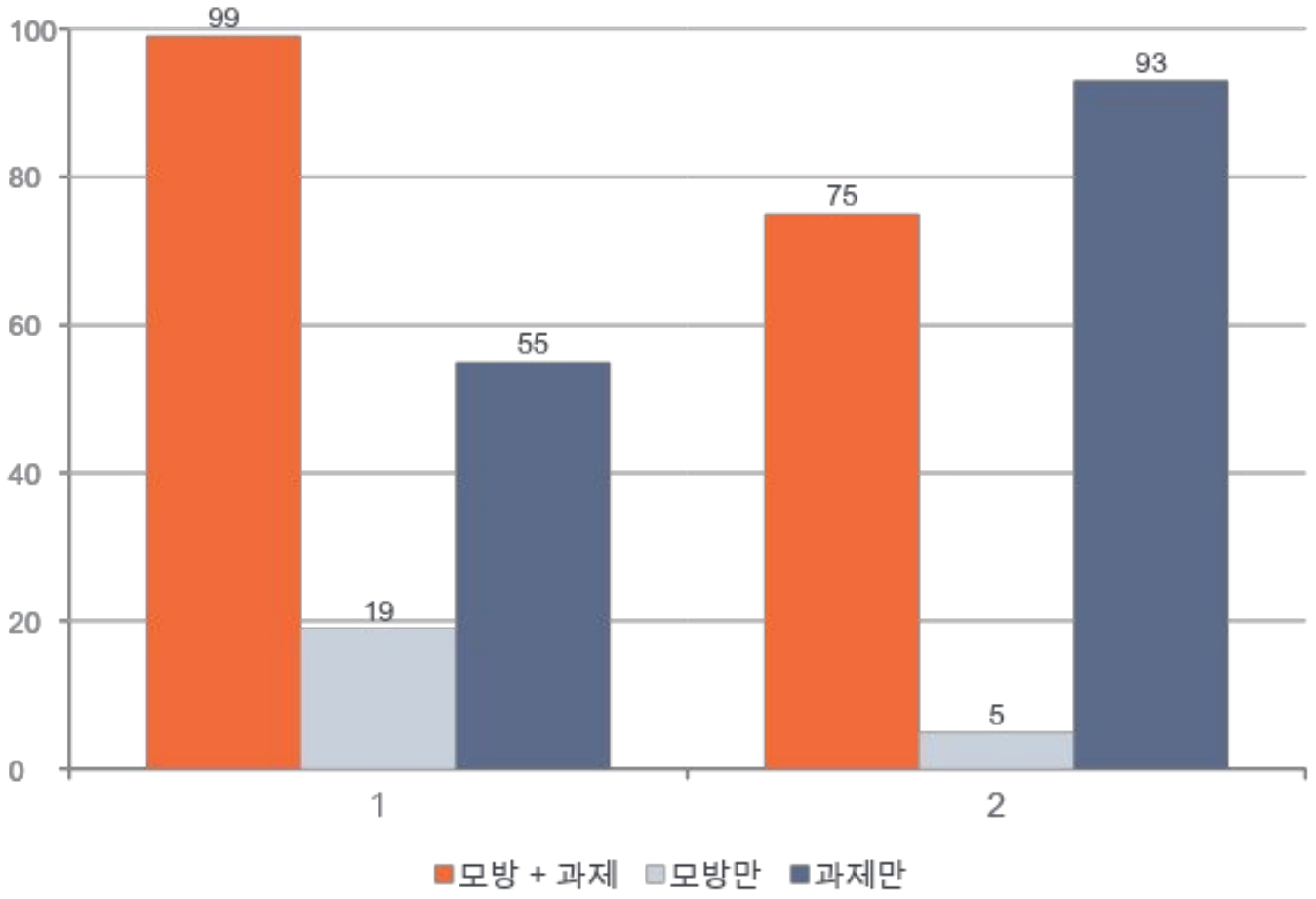
바닥에 누운 채 동작을 흉내 내는 국소해. 쉬운 걷기만 멀쩡하다

RSI를 빼면: 숫자는 멀쩡한데…

백플립 0.730은 공중 회전이 아니라 '뒤로 작게 뛰기'. 정규화 보상만 보면 놓친다 → 영상으로 확인

Peng et al., 2018, Table 5, §10.4 · 평가 시 ET 끄고 고정 시작 상태 · 대부분 1회 학습

Tracking Gives Style: Imitation vs. Task



모방만 → 과제 실패

동작은 예쁘지만 목표를 못 맞춘다 (5~19%)

과제만 → 이상한 성공

던지기 93%인데, 공을 던지지 않고 들고 목표까지 뛰어간다

둘 다 → 자연스럽게 성공

클립에서 필요한 만큼만 벗어난다

Peng et al., 2018, Table 4, Fig. 8 · 각 100회 평가

From One Clip to Thousands: Universal Trackers

DeepMimic의 한계

- 기본은 기술마다 정책을 따로 학습
- 기술 하나에 며칠 학습
- 위상 ϕ로 클립 시간을 고정
- 여러 클립 통합은 소수에서만 검증

→ 하나의 정책이 아무 모션이나 따라가게 할 수 없을까?

PHC · 2023

시뮬레이션 아바타

1만 개 모션 클립을 하나의 컨트롤러로. 어려운 모션마다 네트워크 용량을 점진적으로 추가(PMCP), 넘어져도 리셋 없이 스스로 회복

GMT · 2025

실제 휴머노이드

쉬운·어려운 모션의 학습 비율을 조절하는 적응적 샘플링 + 모션 영역마다 전문가를 두는 Mixture-of-Experts

BeyondMimic · 2025

실제 휴머노이드

같은 설정·하이퍼파라미터로 공중 옆돌기·스핀킥·전력질주를 실제 로봇에 바로 이전, 확산 모델로 기술을 조합

Luo et al., ICCV 2023 (PHC, arXiv:2305.06456) · Chen et al. 2025 (GMT, arXiv:2506.14770) · Liao et al. 2025 (BeyondMimic, arXiv:2508.08241)

Tracking as an Interface: Teleoperation

사람 동작
VR · 카메라

→

실시간 트래킹 정책
로봇이 따라 함

→

시연 데이터 수집
전신 텔레오퍼레이션

→

자율 정책
모방학습(BC)

OmniH2O · 2024

- VR 헤드셋, 음성, RGB 카메라로 같은 정책을 조종 — 운동학적 자세를 공통 인터페이스로
- 실제 로봇엔 정보가 적다 → 특권 정보를 쓰는 교사 정책을 희소 입력 학생 정책으로 증류
- 모은 시연으로 자율 정책 학습, 데이터셋 OmniH2O-6 공개

HumanPlus · 2024

- 카메라 한 대로 사람 몸과 손을 읽어 실시간으로 따라 함 (shadowing)
- 약 40시간의 사람 모션 데이터로 시뮬레이션에서 학습한 저수준 정책
- 따라 하며 모은 시연 최대 40개로 BC → 과제 성공률 60~100%

He et al. 2024 (OmniH2O, arXiv:2406.08858) · Fu et al. 2024 (HumanPlus, arXiv:2406.10454; 자체 제작 33자유도 휴머노이드)

Three Walls on Real Robots

1 몸이 다르다

사람과 로봇은 자유도·관절 범위·힘이 다르다. 사람 모션을 그대로 따라 하면 불가능한 동작이 섞인다

해법

ExBody: 상체만 따라 하고 하체는 속도 명령만 / 리타게팅을 잘 하기 (다음 발표)

2 정보가 적다

시뮬레이션은 정확한 상태를 알지만 실제 로봇은 센서가 제한적이다

해법

특권 정보 교사 → 학생 증류 (OmniH2O, 지난주 RMA·Perceptive와 같은 구조)

3 물리가 다르다

시뮬레이터와 실제 모터·접촉 동역학이 어긋나 민첩한 동작이 실패한다

해법

ASAP: 실제 데이터로 '보정 행동 (delta action)' 모델을 배워 시뮬레이터를 현실에 맞춘 뒤 재학습 (데모)

Cheng et al. 2024 (ExBody) · He et al. 2024 (OmniH2O) · He et al. 2025 (ASAP, arXiv:2502.01143; Unitree G1)

04

Conclusion

한 장의 지도 · 남은 문제 · 가져갈 것

The Big Picture

2005~	고전 WBC	모델 + 최적화로 여러 과제를 우선순위대로 (Sentis & Khatib, Atlas DRC)
2018	DeepMimic	모방 보상 + RSI + ET로 모션 클립 추종 — tracking의 원형 (시뮬레이션)
2023	PHC	1만 개 클립을 하나의 정책으로, 넘어져도 회복 (시뮬레이션)
2024	ExBody · OmniH2O · HumanPlus	실제 휴머노이드에서 전신 트래킹과 텔레오퍼레이션
2025	ASAP · GMT · BeyondMimic	민첩한 동작의 sim-to-real, 범용 트래커, 기술 조합

시뮬레이션 캐릭터 → 실제 휴머노이드, 클립 하나 → 수천 개, 미리 녹화 → 실시간 텔레오퍼레이션

각 장의 출처 참고

Open Problems

1 물체·환경과의 접촉

지금의 트래킹은 주로 자기 몸만 본다. 상자를 들거나 계단을 짚는 상호작용은 참조 모션에서 쉽게 깨진다 → 다음 발표 OmniRetarget

2 참조 모션의 질에 의존

리타게팅이 잘못되면 로봇이 불가능한 동작을 따라 하려 한다

3 '닮음'의 정의는 여전히 수작업

보상 항·k 값·가중치를 사람이 정한다. 판별자가 닮음을 판단하는 AMP(2021) 같은 대안

4 모션을 누가 만들어 주나

트래커는 따라가기만 한다. 상황에 맞는 참조 모션을 생성하는 상위 계획(확산 모델 등)과의 결합

발표자 정리 · Peng et al. 2021 (AMP) · Liao et al. 2025 (BeyondMimic) · Yang et al. 2025 (OmniRetarget, arXiv:2509.26633)