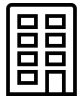
Human-to-Robot Transfer



Sogang University
Vision & Display Systems Lab, Dept. of Electronic Engineering

Presented By
김태우

Contents

- 주제: Human-to-Robot Transfer

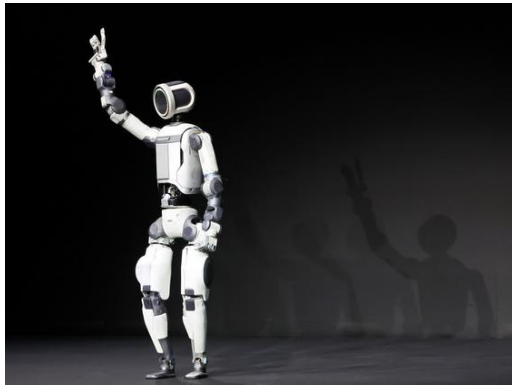
- Background
  - 로봇 패러다임의 변화
  - 데이터 부족(Data Scarcity)의 병목
  - 기존 접근법 분석: Google vs NVIDIA
  - 최신 연구 트렌드
  - Human-to-Robot Motion Retargeting

- 관련 논문 소개
  - Human2Robot: Learning Robot Actions from Paired Human-Robot Videos [arXiv 2025]
  - Phantom: Training Robots Without Robots Using Only Human Videos [CoRL 2025]

1) Brohan, Anthony, et al. "Rt-1: Robotics transformer for real-world control at scale." *arXiv preprint arXiv:2212.06817* (2022).
2) Brohan, Anthony, et al. "Rt-2: Vision-language-action models transfer web knowledge to robotic control. arXiv." *arXiv preprint arXiv:2307.15818* (2023).

Background

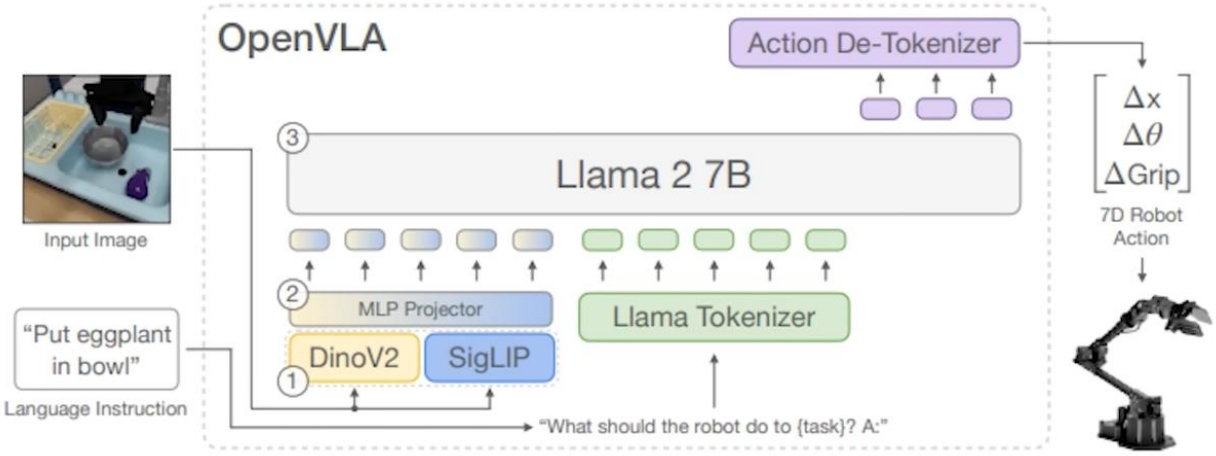
- 선정 배경: 로봇 패러다임의 변화
  - 로봇 제어 방식이 Rule-based에서 Data-driven AI Agent로 변화
- 로봇 시장 및 제어의 변화
  - Hardware: Boston Dynamics의 아틀라스(Atlas) 등 휴머노이드 로봇 성장
  - Software: 기존의 정교한 수식 기반 제어에서 벗어남
    - 대규모 데이터를 학습하는 AI 에이전트 방식으로 변화 양상
    - Google RT-1[1], RT-2[2]와 같이 언어 모델 아키텍처(Transformer)를 로봇 제어에 도입
  - 전망: 미래의 모든 사물은 사람에 최적화, 로봇 또한 사람의 형태(Humanoid)
    - 사람의 행동 양식을 따르는 방향으로 변화할 것



<Boston Dynamics, Atlas>

Background

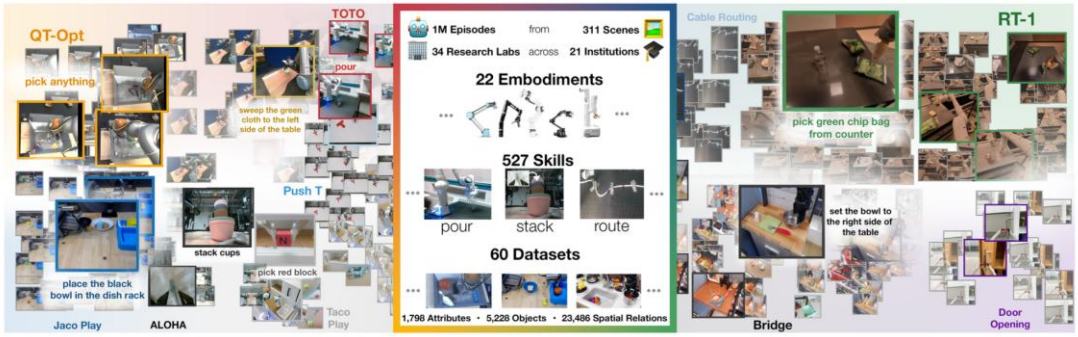
- 시장,학계 문제 제기: 데이터 부족(Data Scarcity)의 병목
  - LLM과 달리 로봇 분야는 학습에 필요한 물리적 행동 데이터가 절대적으로 부족
    - 이를 해결하는 것이 핵심 과제임
  - 데이터의 비대칭성
    - LLM 성공 요인: 인터넷상의 무한한 텍스트 데이터를 학습하여 성공
    - 로봇의 현실: 학습에 필요한 로봇 행동 데이터를 물리 세계에서 취득하는 것은 매우 어렵고, 시간과 비용이 많이 소요됨



https://medium.com/@black_51980/physical-intelligence-for-humanist-robots-08b26aacf8f3

Background

- 기존 접근법 분석: Google vs NVIDIA
  - 데이터 부족을 해결: Google은 실제 데이터, NVIDIA는 시뮬레이션을 활용
- Google (Real-world Data Approach)
  - 다양한 실제 로봇 데이터 기반으로 대량의 학습 데이터 구축 시도
    - RT 시리즈, Open X-Embodiment
- NVIDIA (Simulation Approach)
  - Isaac Sim, Omniverse 등 고도화된 시뮬레이터 활용
    - 가상 공간에서 물리 시간을 가속하여 매우 많은 분량의 경험 학습 (Project GR00T)



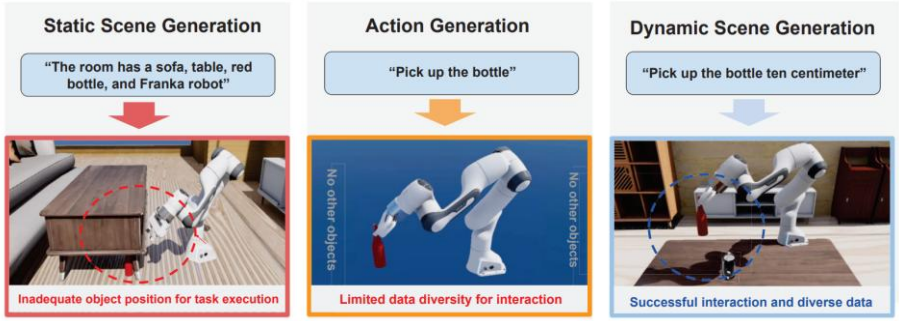
<Open X-Embodiment>

<Nvidia, Simulator based>

1) Lee, Sangmin, Sungyong Park, and Heewon Kim. "DynScene: Scalable Generation of Dynamic Robotic Manipulation Scenes for Embodied AI." *Proceedings of the Computer Vision and Pattern Recognition Conference*. 2025.

Background

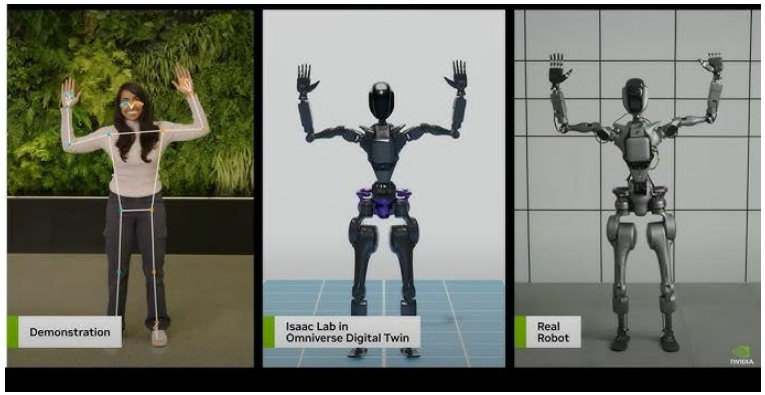
- 최신 연구 트렌드: DynScene[1] (CVPR 2025)
  - 텍스트 지시어로 정적 장면과 동적 액션을 통합 생성하는 DynScene
    - "서랍을 열어라"와 같은 텍스트 명령(Instruction)을 입력
      - Diffusion 모델이 로봇의 초기 위치와 물체, 그리고 실행 궤적(Trajectory)을 생성
    - 방향: 로봇 데이터 부족 문제를 diffusion 기술로 해결
- Training Robots Without Robots
  - 물리 엔진(Isaac Sim) 내에서 실제로 실행 가능한 데이터만 필터링
    - 사람보다 데이터 생성 속도가 26.8배 빠르며, 에이전트의 성공률은 1.84배 높다고 주장
  - 학계 동향: 최근 CVPR, CoRL 등 주요 학회
    - 로봇 없이 시뮬레이션만으로 데이터를 검증하고 학습시키는 논문들이 나오고 있음



<DynScene pipeline>

Background

- 연구 분야: Human-to-Robot Motion Retargeting
  - 핵심 목표 (Objective)
    - 인간 비디오 데이터를 3D 로봇 데이터로 변환하는 파이프라인
    - 사회적 맥락(Social Context)이 포함된 대규모 학습 데이터셋 생성
  - Future work (인간 행동 모사)
    - SMPL 기반 Latent Representation으로 표현, 로봇이 사람처럼 행동하도록 데이터 생성
    - 멀티모달 Diffusion 파이프라인
      - Input: 인간 RGB 영상 + 텍스트 정보(Text Description)
      - Output: 로봇 RGB 데이터 + 행동(Action) 데이터 생성



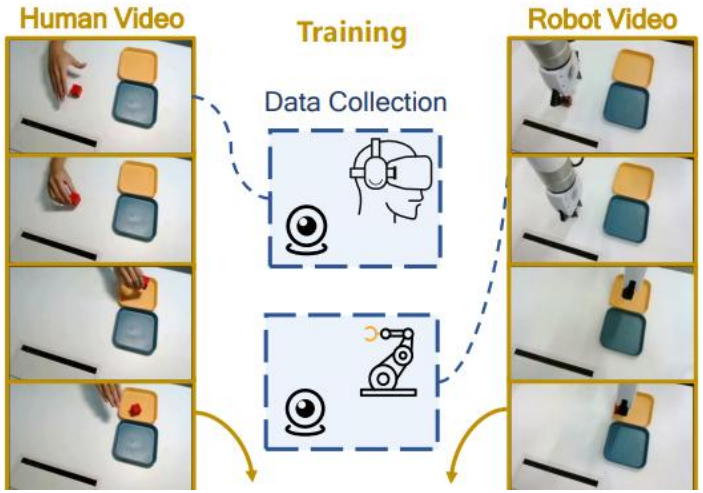
<Nvidia, Human motion retargeting>

<Boston Dynamics, Human motion retargeting>

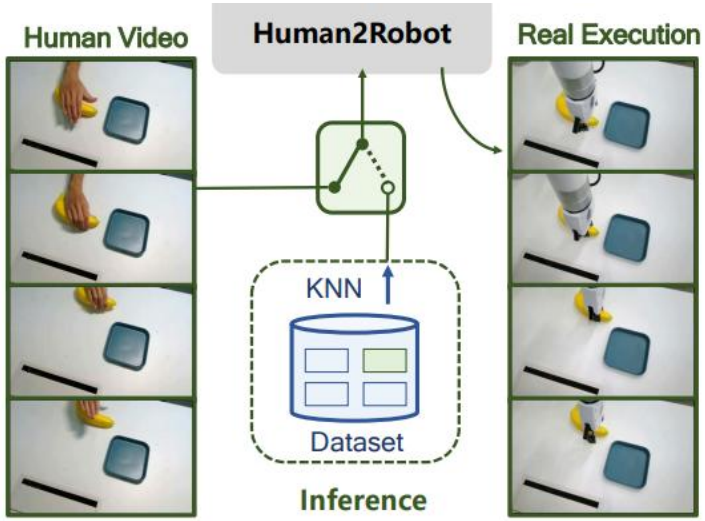
Human2Robot: Learning Robot Actions from Paired Human-Robot Videos

Introduction

- Contribution
  - Human-Robot 매칭 데이터셋 구축
    - 사람의 손과 로봇 팔의 움직임이 시공간적으로 완벽하게 정렬된 데이터 세트 구축
  - Human2Robot Framework 제안
    - 사람의 영상을 입력받아 로봇의 영상을 생성하는 프레임워크
      - 로봇 영상에 행동을 학습하는 2단계 생성형 프레임워크를 개발



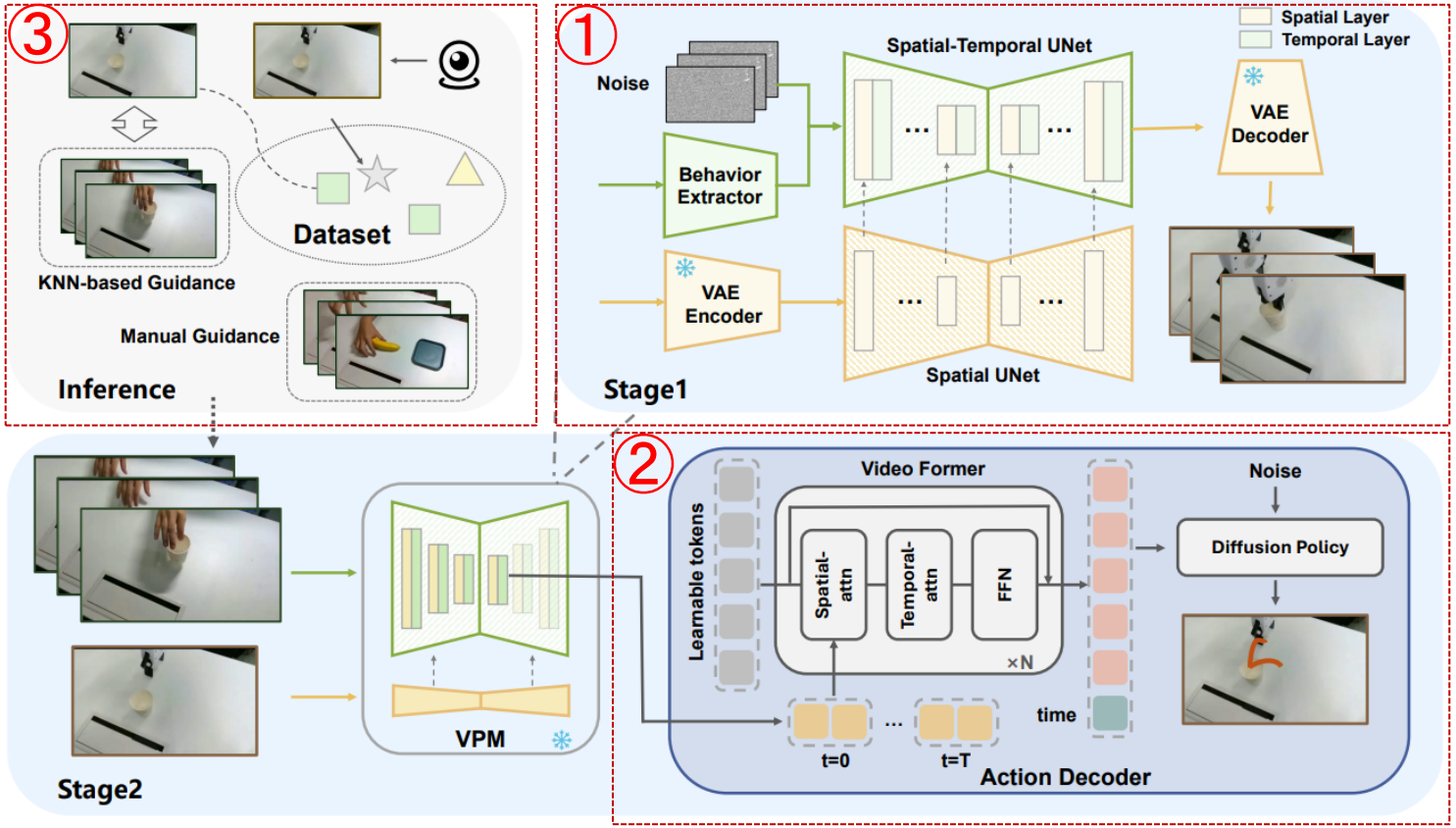
<Human-Robot 매칭 데이터셋 구축>



<Human2Robot Framework 제안>

Method

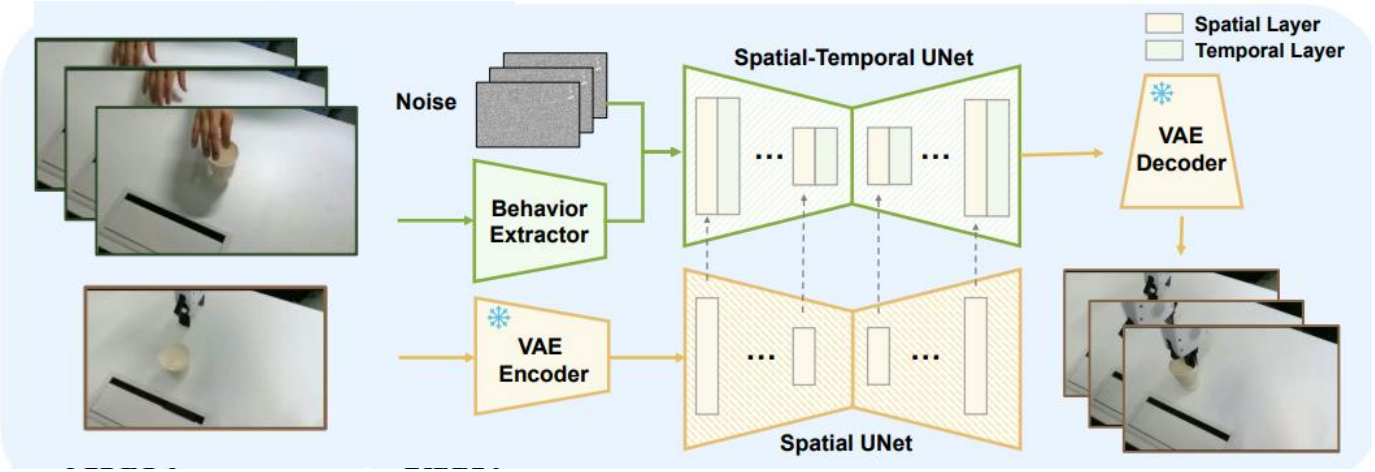
- Framework overview
  - 총 3파트로 나누어 설명



<Human2Robot Framework>

Method

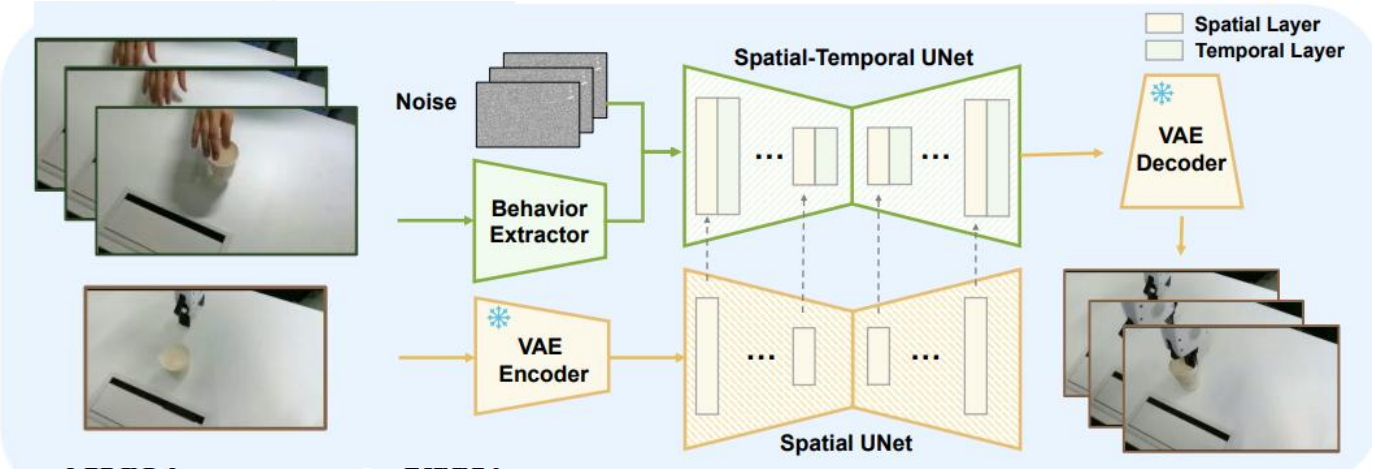
- Part 1: Synthesizing Robot to Human data
  - 입력
    - Human: 사람의 시연 비디오 (Human Video)
    - Robot: 액션의 첫 번째 프레임 (Robot Initial Frame)
  - 출력: Robot RGB image
  - 필요성: Human Data 위에 로봇 팔의 움직임을 Diffusion 모델로 합성
    - 사람의 손 움직임과 로봇의 움직임은 구조적으로 다름
      - 로봇이 이 동작을 한다면 어떤 형태일지 먼저 영상으로 생성



<ST-Unet>

Method

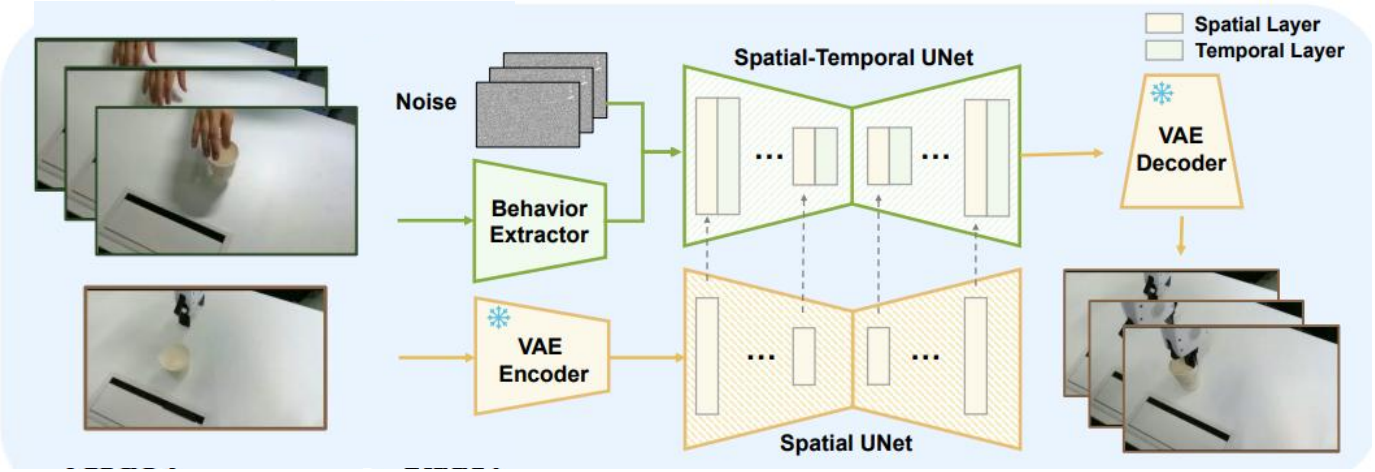
- Part 1: Synthesizing Robot to Human data
  - Feature encoding
    - Human: Behavior Extractor
      - 수행 중인 동작에 대한 행동적 특징(Action Prior)을 인코딩하여 ST-UNet에 전달
        - ✓단순 컨볼루션 층으로 구성
    - Robot: VAE Encoder
      - 구조적인 부분을 담당
      - 기존 Stable Diffusion의 사전 학습된 가중치를 그대로 가져와 고정



<ST-Unet>

Method

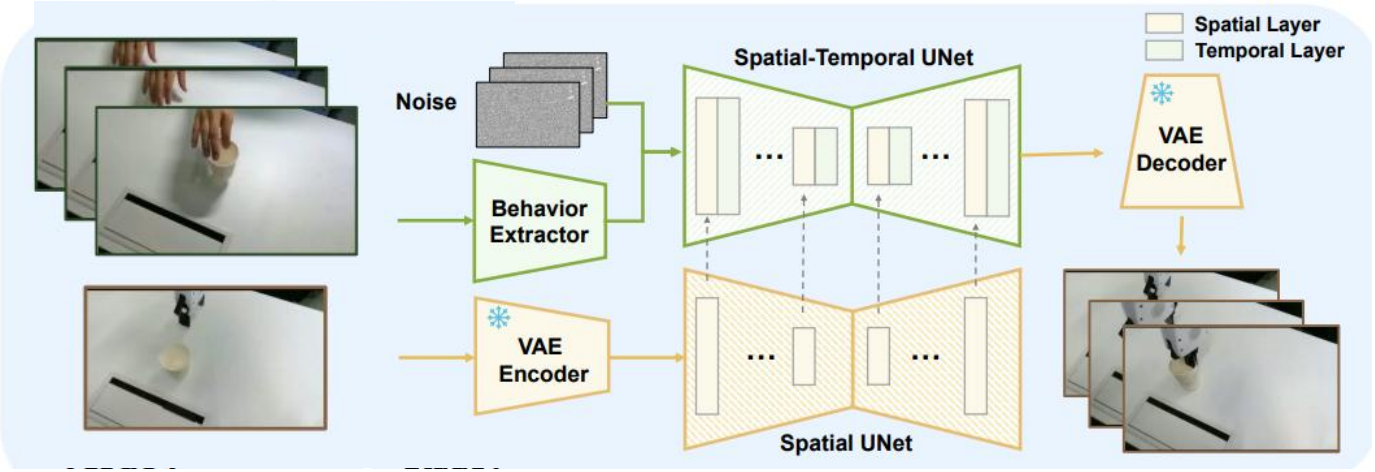
- Part 1: Synthesizing Robot to Human data
  - ST-UNet (Spatial-Temporal UNet)
    - Input: Noise (가우시안 노이즈) + Behavior Features (인간 행동 정보)
  - Why Spatial? (Spatial UNet과 합치는 이유)
    - 로봇의 외형(Appearance) 유지 목적
      - 움직이는 주체가 '로봇 팔'이라는 구조적 형태 정보는 Spatial UNet에서 가져옴
  - 시공간 표현과 결합 (T-Layer)
    - 인간 비디오의 시간적 흐름을 조건으로, 프레임 간의 연관성 학습



<ST-Unet>

Method

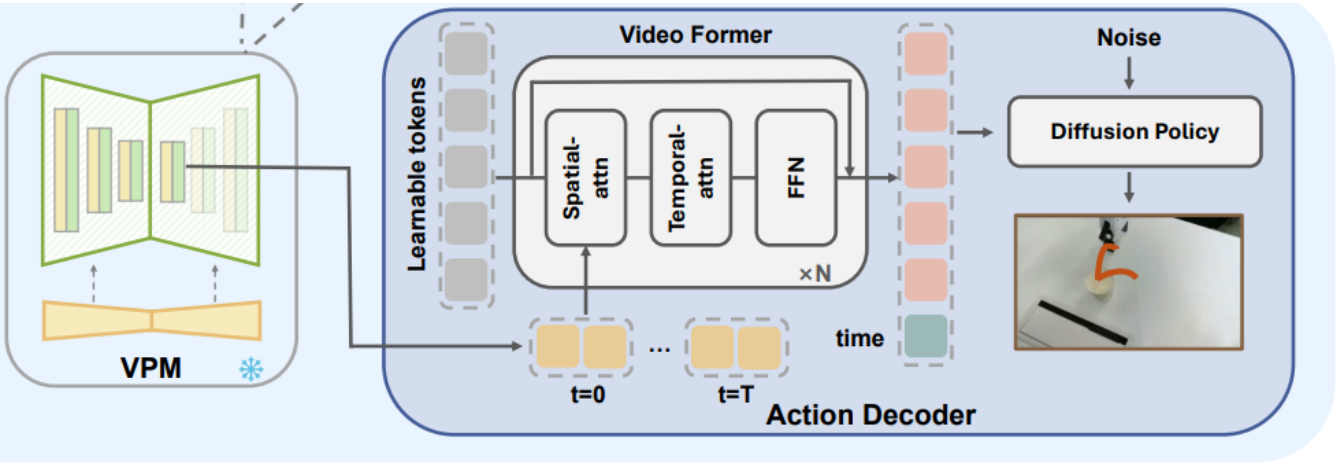
- Part 1: Synthesizing Robot to Human data
  - ST-UNet (Spatial-Temporal UNet)
    - Input: Noise (가우시안 노이즈) + Behavior Features (인간 행동 정보)
    - Output: Latent representation
  - VAE Decoder (Pretrained Stable Diffusion 가중치)
    - 역할: Latent representation → 이미지 공간(pixel space) 로 매핑
    - Output: RGB image



<ST-Unet>

Method

- Part 2: Action Decoding
  - 필요성: 비디오 생성 모델(VPM)은 RGB 영상만 구할 수 있음
    - 로봇을 움직이는 구동 명령(Action Command)을 주지는 않음
  - VPM을 Visual Encoder로 재활용
    - Feature Extraction
      - VPM의 Upsampling Layer 에서 Latent Features을 추출, Action Decoder의 입력 사용



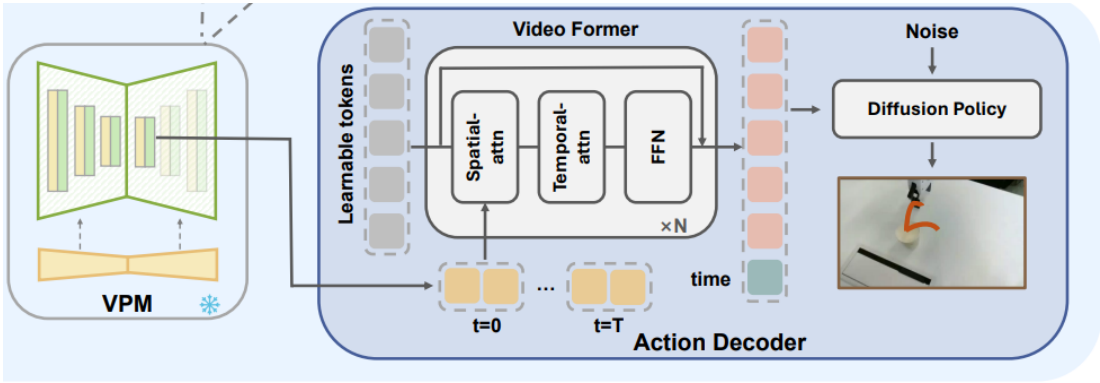
<Action Decoder>

Method

- Part 2: Action Decoding
  - Video Former
    - 추출된 시공간 특징을 학습 가능한 토큰(Learnable Tokens)으로 압축 및 요약
    - Spatial-Attn & Temporal-Attn을 순차적으로 적용하여 동작의 Context을 요약
  - Diffusion Policy[1] (Action Generation)
    - 요약된 특징(F_{VF})을 조건(Condition)으로 사용
      - 노이즈로부터 로봇의 궤적(Trajectory)을 생성
    - Output: Action vector (7차원)
      - Gripper 위치 (Translation)
      - Gripper 회전 각도 (3차원)
      - Gripper 개방 여부 (0/1)

$$\text{Action} = [\ \underbrace{x, y, z}_{\text{Translation}(\mathbb{R}^3)}\ ,\ \underbrace{r_x, r_y, r_z}_{\text{Rotation}(\mathbb{R}^3)}\ ,\ \underbrace{g}_{\text{Gripper}(\mathbb{R}^1)}\ ] \in \mathbb{R}^7$$

<Action Vector>

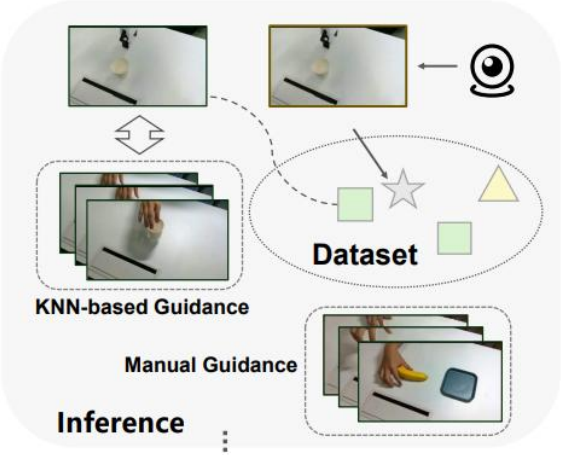


<Action Decoder>

Method

- Part 3: Inference (KNN Supporting)
  - Input: 현재 시점의 로봇 카메라 RGB 데이터
    - 가용 데이터 (Memory Bank): 로봇과 Calibrated 된 사람 손 데이터
  - Output
    - 현재 로봇과 가장 유사한 특징을 가진 Human Video 인덱스
  - 방법
    - Feature 를 추출(Dinov2)하고 KNN 을 사용해 가장 유사한 Human video 인덱스 탐색

→ 탐색한 비디오를 추론에 활용



<KNN 기반, 유사 비디오 탐색>

1) Chi, C. "Diffusion policy: visuomotor policy learning via action diffusion. arXiv preprint." *arXiv preprint arXiv:2303.04137* (2023).
2) Xu, Mengda, et al. "Xskill: Cross embodiment skill discovery." *Conference on robot learning*. PMLR, 2023.
3) Hu, Yucheng, et al. "Video prediction policy: A generalist robot policy with predictive visual representations." *arXiv preprint arXiv:2412.14803* (2024).

Experiment

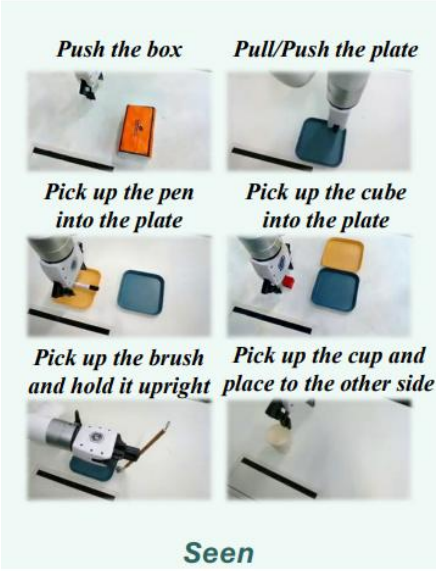
- Setups
  - Dataset: H&R (Human-to-Robot)
    - 구축 방식: VR 텔레오퍼레이션을 통해 수집
    - 특징: 인간의 손과 로봇 팔의 움직임이 시공간적으로 완벽히 정렬(Paired)된 비디오
    - 데이터규모: 총 2,600 에피소드
  - Tasks
    - Basic Tasks (총 8개): 3가지 카테고리로 구성

•**Pick and place the cup:** 컵을 집어서 다른 위치로 옮기기
•**Pick and place the cube:** 큐브(육면체)를 집어서 접시 위에 올리기
•**Pick and place the pencil:** 펜을 집어서 접시 위에 올리기 (얇은 물체 집기)
•**Push the box:** 상자를 왼쪽에서 오른쪽으로 밀기 (비파지 조작, Non-prehensile)
•**Pull the plate:** 접시를 당겨오기
•**Push the plate:** 접시를 밀어내기
•**Pick up and hold the brush upright:** 붓을 집어서 세우기 (회전 제어 중요)
•**Handwriting (Play data):** 종이 위에서 의미 없이 끄적이는 행동 (자유로운 궤적 학습용)

  - Evaluation metric
    - Success Rate (성공률): 각 태스크 당 20회 수행 후 성공 횟수 측정
  - Baselines (비교 모델)
    - Diffusion Policy[1] (DP): CLIP 언어 조건을 사용하는 Action Diffuser
    - XSkill[2]: 비디오 간 정렬에 의존 (Human Video를 단순 라벨로 활용)
    - Video Prediction Policy[3] (VPP): 비디오 예측 사전 학습 활용, Language-conditioned 의존

Experiment

- Seen task Test
  - 동작 특성에 따라 3가지 카테고리로 그룹화하여 평가
    - Push & Pull, Pick & Place, Rotation (Brush handling)
    - Human2Robot: 평균 성공률 95% 달성
  - Ablation study
    - VPM 없이 사람 영상을 Action Decoder에 직접 입력 (23%로 하락)
    - VPM 학습 없이 바로 디코딩 (10%로 하락)

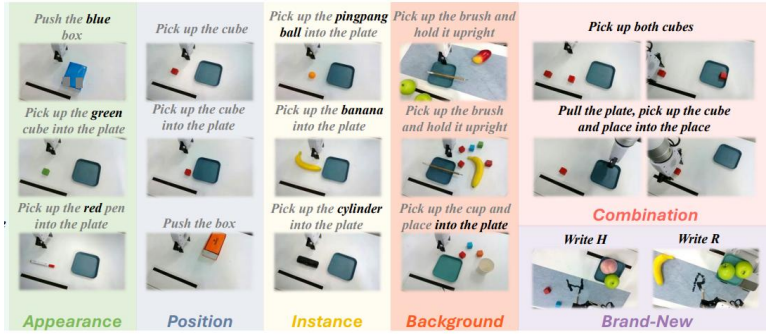


	Push & Pull	Pick & Place	Rotation	Average
DP (Chi et al. 2023)	50	20	15	28
XSkill (Xu et al. 2023b)	70	40	50	53
VPP (Hu et al. 2024)	95	70	75	80
Action Decoder w. Human	50	10	10	23
HUMAN2ROBOT *w/o. Pretrain*	20	10	0	10
HUMAN2ROBOT *w. KNN*	90	75	80	82
HUMAN2ROBOT *(ours)*	**100**	**90**	**95**	**95**

<Table 1: Seen task Test results>

Experiment

- Generalization Test
  - Unseen Scenarios (학습하지 않은 환경)
    - Appearance (색상/재질): 학습에 없는 다른 색의 동일 물체 조작
    - Instance (새로운 물체): 학습 때 본 적 없는 바나나, 공 등 조작
    - Background (배경 변화): 방해 물체 및 새로운 배경 환경
    - Task Combination (복합 작업 연속 수행): 단일 작업들의 순차적 조합
      - 예: 접시 당긴 후 물체 놓기
  - Brand-New Task (완전히 새로운 작업)
    - Handwriting (글씨 쓰기): 글자를 쓰는 법을 배운 적 없으나, 사람을 보고 따라 함
      - One-shot Generalization 능력을 통해 70% 성공률 달성



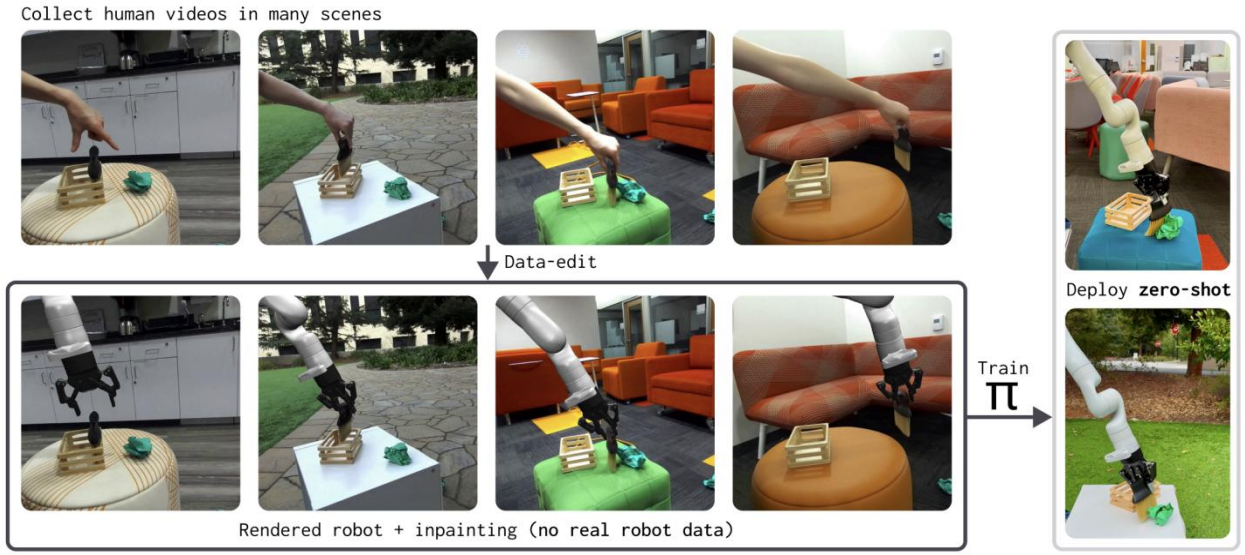
Generalization	XSkill	VPP	H2R(ours)
Appearance	0	80	**100**
Position	20	50	**80**
Instance	0	0	**70**
Background	0	0	**80**
Combination	0	0	**50**
Brand-New	0	0	**70**

<Unseen task Test results>

Phantom: Training Robots Without Robots Using Only Human Videos

Introduction

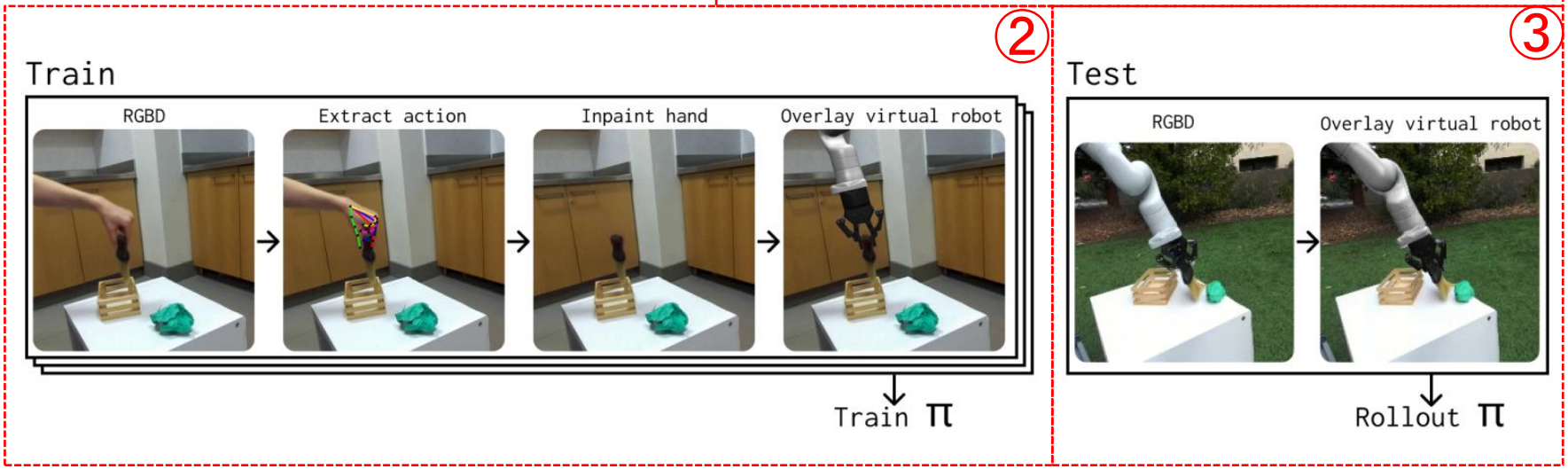
- Contribution
  - Zero-Robot Data Generation
    - 로봇 데이터 없이 오직 인간 비디오만으로 정책을 학습
      - 로봇 teleoperation 병목 현상 제거, 확장성(scalability) 문제를 해결
  - Zero-Shot Generalization
    - 학습에 사용되지 않은 새로운 배경(Scene)이나 조명 환경(OOD)에서도 즉시 배포 가능
      - : 인간의 팔을 지우고, 가상 로봇을 합성(Overlay)하여 시각적 도메인 해결



<Phantom pipline>

Method

- 총 3개의 파트로 설명
  - Part 1: Collect Human videos
  - Part 2: Human-Robot data Transfer
  - Part 3: Zero-shot Testing



<Phantom pipline>

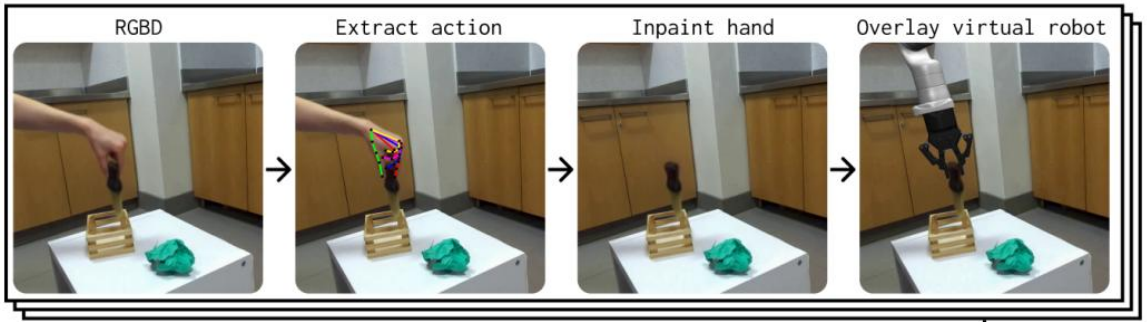
Method

- Part 1: Collect Human videos
  - Human Video를 통한 확장성 (Scalability)
    - 고가의 로봇 장비나 텔레오퍼레이션 장치 없이, RGB-D 카메라만으로 데이터 수집
      - 로봇의 Parallel Jaw Gripper를 모사하기 위해, 엄지와 검지만을 사용하는 형태 조작
    - 야외, 라운지 등 다양한 환경(In-the-wild)에서 수집하여 데이터 분포 다양화
  - RGB-D 카메라 활용 이유
    - 정밀한 3D Action 추출 (Precise Action Extraction)
    - 단안(Monocular) 기반 손 포즈 추정(HaMeR)은 본질적으로 Scale Ambiguity를 가짐

<Collect Human videos>

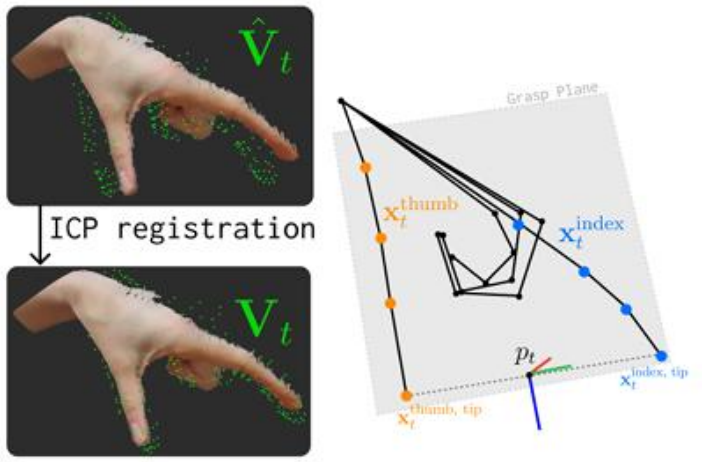
Method

- Part 2: Human-Robot data Transfer
  - 3D Hand Mesh Recovery
    - 목적: Human Video 프레임$(I_{h,t})$으로부터 3D Hand Keypoints를 추출
    - Robot Action (p_t, R_t, g_t)으로 매핑
  - Model: HaMeR[1)]
    - Input: Single RGB Frame $(I_{h,t})$
    - Output: 3D Hand Mesh Vertices $(\hat{V}_t)$ & 21 Keypoints
  - ICP Align (Refinement)
    - Depth Map에서 추출한 손 영역 Point Cloud (P_t)에 HaMeR의 예측 Mesh $(\hat{V}_t)$ 정합
    - Result: Scale Ambiguity가 제거된 정밀한 Absolute 3D Hand Pose 획득



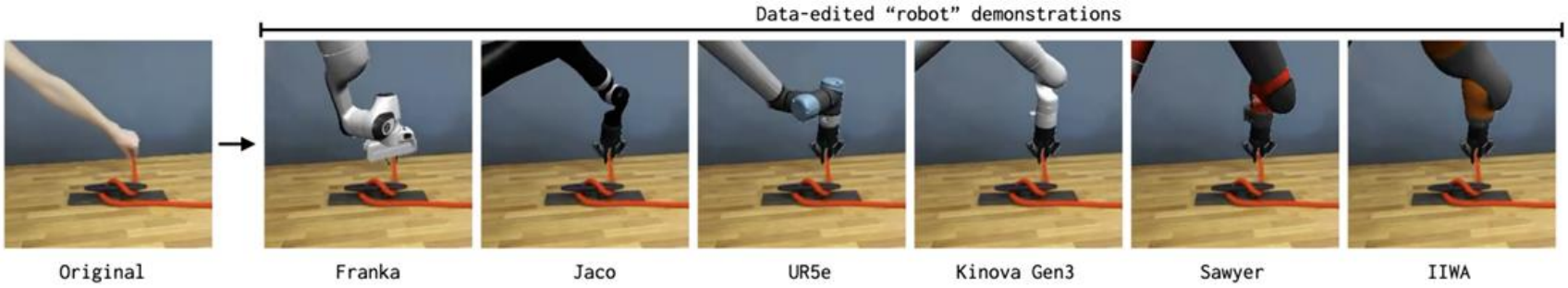
Method

- Part 2: Human-Robot data Transfer
  - Robot 캘리퍼 위치 추정: Hand ICP Align with Depth
    - ICP 보정된 Hand Keypoints를 Robot End-effector Action (p_t, R_t, g_t)으로 매핑
  - Action Mapping Logic
    - Position (p_t)엄지 끝(x^{thumb})과 검지 끝(x^{index})의 중점(Midpoint)
    - Rotation (R_t): 엄지와 검지 키포인트들이 이루는 평면(Plane)의 법선 벡터(Normal Vector)를 이용해 계산
    - Gripper Width (g_t): 엄지와 검지 끝 사이의 유클리드 거리(Distance)



Method

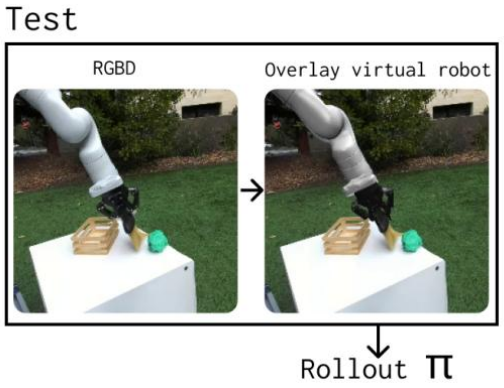
- Part 2: Human-Robot data Transfer
  - Visual Domain Transfer Pipeline
    - Inpainting: SAM2[1]로 사람 팔을 Segmentation 후 E2FGVI[2]로 제거
    - Rendering: Mujoco 엔진을 이용해 추출된 Action (p_t, R_t) 위치에 가상 로봇 생성
      - 실제 하드웨어와 기구학적(Kinematic)으로 동일한 Digital Twin 사용
      - End-effector Action (p_t, R_t) 을 입력으로 받아, Inverse Kinematics 사용
        - ✓사람의 손 위치에 로봇 손이 정확히 위치하도록 7-DOF 관절 값을 역으로 산출
    - Overlay: 실제 카메라 파라미터(Extrinsics)와 Depth 정보를 사용해 로봇 합성



<Overlapping robots>

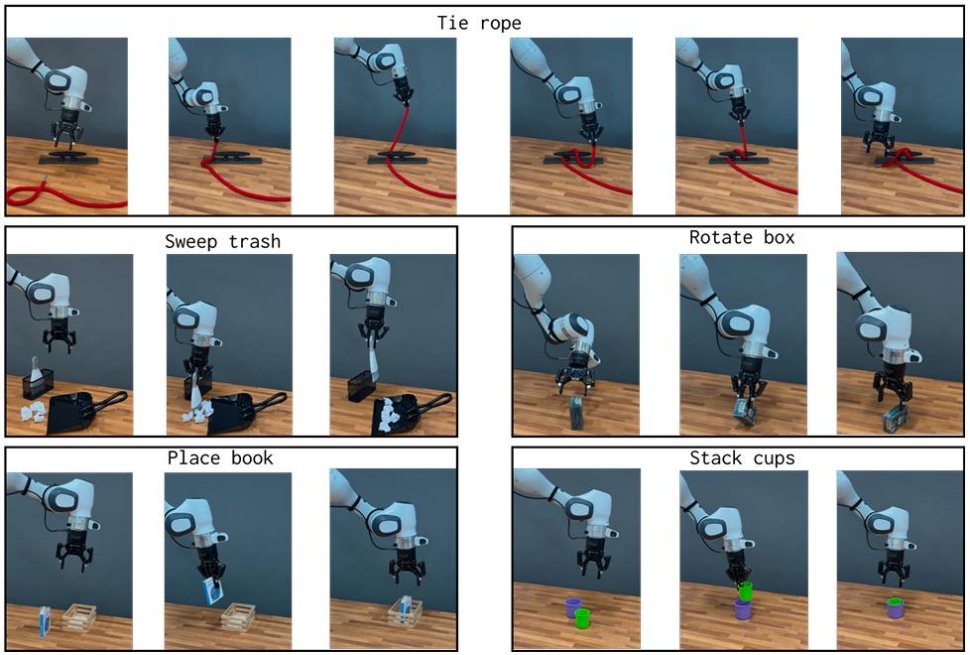
Method

- Part 3: Zero-shot Testing
  - Inference-Time Overlay Strategy
  - Problem: 학습 데이터와 테스트 데이터(Real Robot) 간의 차이
    - 미세한 색감/질감 차이 (Visual Domain Gap) 존재
  - Solution: Test-Time Overlay
    - Real Robot 구동 시에도 학습 때와 동일한 가상 로봇(Virtual Robot)을 영상 위에 덧씌움
      - 관절 상태(Proprioception)를 받아 실시간 렌더링 및 합성
  - Effect: 학습과 테스트 환경을 시각적으로 완벽히 일치시킴
    - 별도의 Domain Adaptation 없이 Zero-shot Deployment 가능
  - 생성된 Data pair 을 사용하여 Diffusion Policy (Π) 학습



Experiments

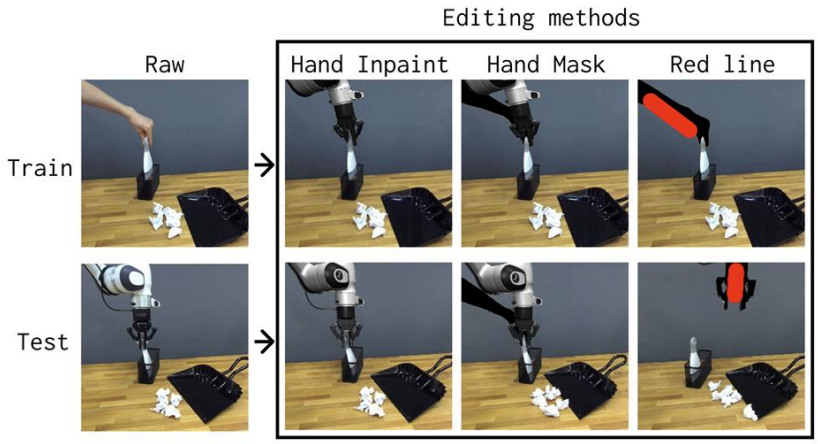
- Setup & Baselines
  - Metric
    - 성공률: 각 태스크 당 25회의 실제 실행 (Real-world Rollouts) 수행 후 성공 횟수 측정
  - Tasks: 5 Diverse Tasks (Rigid, Deformable, Multi-object)
    - Pick & Place Book
    - Stack Cups (Precision)
    - Tie Rope (Deformable)
    - Sweep Trash (Dynamic)
    - Rotate box



<5 Diverse Tasks >

Experiments

- Setup & Baselines
  - Baselines (Data Editing Strategy):
    - Hand Inpaint: 사람 팔 제거 후 가상 로봇 합성
    - Hand Mask: 사람 팔을 검은색 마스크로 가림
    - Red Line (EgoMimic style): 팔을 지우고 빨간 선으로 표시
    - Vanilla: 원본 Human Video 그대로 사용
  - Key Results (In-distribution)



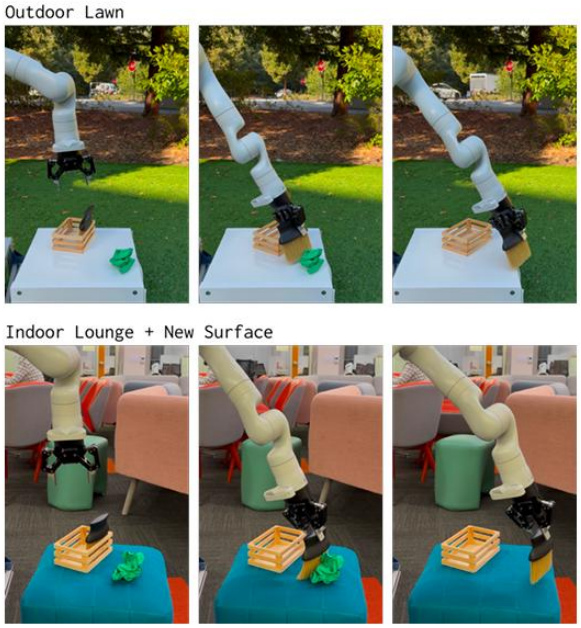
<Data Editing Strategy>

	Pick/ Place Book	Stack Cups	Tie Rope	Rotate Box
Hand Inpaint	0.92	0.72	0.64	0.72
Hand Mask	0.92	0.52	0.60	0.76
Red Line	0.0	0.0	0.0	0.0
Vanilla	0.0	0.0	0.0	0.0

<Results>

Experiments

- Zero-shot Generalization to Novel Scenes
  - Task: Sweep Trash (빗자루로 쓰레기 쓸기)
    - Training Data: 950 Human Demos (다양한 배경 포함)
  - 3 Levels of OOD Scenarios (Unseen during Training)
    - Indoor Lounge: 훈련 데이터에 없던 새로운 가구 배치의 실내 라운지
    - Indoor Lounge + New Surface: 새로운 파란색 천(Surface) 위에서 수행
    - Outdoor Lawn: 완전히 새로운 야외 잔디밭 환경, 동적인 배경



<Test Data(Unseen during Training)>

<New Surface>

	Outdoor lawn	Indoor lounge	Indoor lounge + OOD surface
Hand Inpaint	0.72	0.84	0.64
Hand Mask	0.52	0.76	0.68

<Zero-shot testing>

Experiments

- Analysis: Data Efficiency (Scalability vs Precision)
  - 로봇 데이터와 제안 방법론 비교 분석
    - Trade-off: Scalability vs Precision
  - Human Video
    - Scalability (↑):로봇 없이 대규모 수집 가능
    - Precision (↓):RGB-D 기반 손 포즈 추정 오차 존재
  - Comparison Result:
    - Small Data (50 demos): Robot Data(52%) > Human Data (44%) → 정밀도 차이 존재
    - Scale-up (300 demos): Human Data (84%) ≈ Robot Data 100개 (88%)

# of demos	Robot only	Human only
25	0.16	—
50	0.52	0.44
100	0.88	0.64
300	—	0.84

<Data Efficiency 분석>