Journal of Korea Robotics Society (2025) 20(3):371-380
https://doi.org/10.7746/jkros.2025.20.3.371 ISSN: 1975-6291 / eISSN: 2287-3961

시뮬레이션-실환경 통합 점수 기반 로봇 정책 선택 기법

Unified Sim-and-Real Scoring Methods for Robot Policy Selection

임 수 빈[1] · 이 재 선†
Subin Im[1], Jaeseon Lee†

Abstract: Sim2Real transfer aims to deploy control policies trained in simulation onto real robots, but discrepancies in contact dynamics, friction, and sensor noise often hinder performance. We propose Illy-Net, a hierarchical dual-stream evaluation framework that combines large-scale domain-randomized simulation with a lightweight real-world correction module. Candidate policies are evaluated across diverse simulated domains, yielding normalized performance vectors. A small subset of policies is then tested on real hardware, and a Gaussian Mixture Model (GMM) is trained to estimate real-world success likelihood based on simulation features. Using dimensionality-reduced actor features (via PCA and AutoEncoder), Illy-Net achieves policy ranking with over 70% reduction in on-robot trials, improving top policy success rates from 77.3% to 91.7%. Our approach enables reliable and efficient policy selection for industrial robotic tasks under minimal data and hardware budgets.

Keywords: Sim2Real, Gaussian Mixture Model, Policy Evaluation, Robot Manipulator

1. 서 론

시뮬레이션 환경에서 학습된 강화학습 기반 로봇 제어 정책은 낮은 비용과 빠른 속도로 대규모 병렬 에피소드를 수행할 수 있다는 장점으로 인해 현대 로봇공학 연구의 핵심 방법론으로 자리잡고 있다. 그러나 이러한 정책을 실제 환경에 적용할 경우 센서 지연, 마찰 계수, 운동학 오차, 조명 및 텍스처 차이 등 다양한 물리적·시각적 요인으로 인해 도메인 간 간극(domain gap)이 발생하며, 시뮬레이션에서의 성능이 현실에서 그대로 재현되지 않는 문제가 나타난다. 이 간극을 완화하기 위해 도메인 랜덤화, 도메인 적응, 시스템 식별 등의 접근법이 시도되어 왔으나, 현실 세계의 복잡한 동역학을 정밀하게 재현하려면 상당한 실험 자원과 고비용 하드웨어가 요구된다. 이러한 비용과 긴 실험 주기는 여전히 Sim2Real 전이 학습의 효율성과 확장성을 저해하는 주요한 제약조건으로 작용하고 있다.

본 연구는 이러한 한계를 해결하기 위해 Illy-Net (Integrated Likelihood-Layered sYstem Network) 이라 명명한 새로운 이중 평가·보정 프레임워크를 제안한다. Illy-Net은 최소한의 실험 데이터만으로도 수십 개 이상의 시뮬레이션 기반 학습 정책들 중에서 최적의 후보를 효율적으로 선별할 수 있으며, 아직 실험되지 않은 정책들의 현실 성능까지 정량적으로 추정할 수 있다.

제안된 프레임워크는 두 개의 병렬 흐름으로 구성된다. 첫 번째 흐름에서는 모든 후보 정책을 세 가지 도메인 랜덤화 환경(DR_{val})에서 평가하고 성공률·실패율 등의 원시 지표를 기반으로 n차원의 시뮬레이션 점수 벡터 S_i를 구성한다. 두 번째 흐름에서는 동일한 정책들을 별도의 도메인(DR_j)에서 재실행하여 actor 특성 벡터를 수집하고, 단일 현실 실험 배치에서 얻은 관측 데이터를 동일한 특성 공간에 투영한다. 이때, 시뮬레이션-현실 행동 특성 쌍을 이용해 정책별 가우시안 혼합모델(GMM)을 학습하고, 현실 특성 기반의 로그우도(log-likelihood)를 원시 위험도 R_i로 정의한다. 최종 점수는 $X_i = S_i \cdot R_i$로 계산되며, 정규화 및 가중합이 적용된다.

Received : May. 30. 2025; Revised : Jul. 16. 2025; Accepted : Jul. 17. 2025
※ This work was supported by the ICT R&D program of MSIT/IITP (No. 2022-0-00067, Development of edge brain technology for judgement, control, and collaboration of manufacturing equipment and robots).
1. Graduate Student, Intelligent Robotics, SKKU, Suwon, Korea (sue4084@g.skku.edu)
† Team Leader, Corresponding author: Manufacturing Robot, KITECH, Ansan, Korea (js.lee@kitech.re.kr)

특성 표현력을 향상시키기 위해 주성분 분석(PCA) 및 Auto Encoder를 활용하며, GMM의 강건성을 높이고 예측 불확실성을 줄이기 위해 가우시안 프로세스 회귀(GPR) 등 다른 대리 모델을 병행하여 교차 검증을 수행한다. Illy-Net은 단일 현실 실험 배치만으로도 GMM log-likelihood를 통해 핵심 정보를 보존할 수 있기 때문에 현실 데이터 수집 비용이 높은 산업용 로봇 환경에서도 높은 실용성과 확장성을 갖는다.

기존 방법들은 대규모 도메인 랜덤화, 반복적 시뮬레이터 보정, 픽셀 수준 GAN 기반 도메인 변환, 혹은 작업 특화 대리 모델 구축에 의존하였다. 이에 반해 Illy-Net은 시뮬레이션 점수 S_i와 현실 위험도 R_i라는 두개의 핵심 지표를 GMM log-likelihood로 통합함으로써, 단 한 번의 현실 실험만으로도 미실험 정책들의 성능을 통계적으로 추정할 수 있다는 점에서 차별성을 지닌다. 이러한 구조는 대규모 시뮬레이션 롤아웃이나 반복 학습 없이도 고비용 제약을 갖는 산업현장에 즉시 적용 가능하며, 조립, 경로 계획, 장애물 회피 등 다양한 다운스트림 작업에도 확장 적용될 수 있다.

Sim2Real 간극을 줄이기 위한 최근 연구로는 6 자유도 파지 성공률을 20%p 향상시킨 GPDAN[1], 대규모 시각·물리 랜덤화로 84% 성공률을 달성한 연구[2], 시뮬레이터를 메타 학습하거나(AdaptSim[3]) 적대적으로 탐색하는 접근(EASI[4]), 현실 데이터를 수 분 내로 줄이는 잔차 학습 기법[5,6], 깊이 센서 없이도 투명 물체를 파지 가능한 R2SGrasp[7] 및 RGBGrasp[8], 비전-언어 사전학습을 활용한 전략[9,10], 인과 분석 기반의 실패 진단 기법[11] 등이 있다.

이러한 선행 연구들은 Illy-Net 설계에 실질적인 영감을 제공하였으며, 시뮬레이션 점수와 단일 배치 GMM 위험도를 통합함으로써 Sim2Real 효율을 극대화하는 데 핵심적인 동기를 부여하였다. 본 논문은 UR16e 로봇을 활용한 파지(grasping) 시나리오를 통해 Illy-Net의 유효성을 실증하였으며, 소량의 현실 데이터만으로도 실제 성공률을 효과적으로 예측하고 실제 실험을 통해 높은 정확도로 재현 가능함을 확인하였다.

2. 관련연구

Sim2Real (S2R) 전이학습은 시뮬레이션 환경에서 학습된 정책을 실제 로봇 환경에 안정적으로 적용하기 위한 연구 분야로, 일반적으로 네 가지 접근으로 분류된다: ① 시뮬레이터 파라미터 보정(Simulator Parameter Calibration), ② 대리 모델 기반 성능 예측(Surrogate-Based Performance Prediction), ③ 도메인 랜덤화(Domain Randomization), ④ 도메인 적응(Domain Adaptation). 본 절에서는 각 접근을 대표하는 기존 연구를 검토하고, 본 연구의 차별성과 기여를 기술한다.

2.1 시뮬레이터 파라미터 보정

현실과 시뮬레이션 간의 동역학 차이를 줄이기 위한 접근으로, Chebotar 등[12]은 SimOpt를 제안하여, 소수의 실제 실험으로 시뮬레이터 파라미터 분포를 역추정하고 이후 재훈련 과정을 거쳐 정책을 개선하였다. 이러한 방식은 정책 정확도 향상에는 효과적이지만 “실험 → 시뮬레이터 파라미터 조정 → 정책 재학습 → 재검증”이라는 반복 루프를 요구하므로 학습 시간과 자원이 과도하게 소모된다. 본 연구는 이러한 반복을 제거하기 위해, 단일 현실 실험 배치만을 이용하여 GMM 기반 사후 위험 추정으로 정책 성능을 평가하는 프레임워크를 제안한다.

2.2 대리 모델 기반 성능 예측

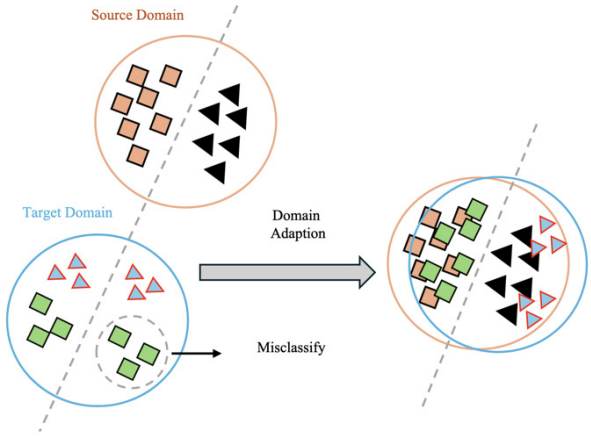
시뮬레이션-현실 간 성능 관계를 함수 근사 형태로 예측하고자 하는 연구도 활발하게 진행되고 있다. Muratore 등[13]은 SPOTA 기법을 통해 현실 성능이 하락하기 전의 최적 중단 시점을 추정하였으나, 개별 정책 간 정량적 우열 판단은 불가능하다. Anwar 등[14]은 능동 실험 선택(active experiment selection)으로 다중 작업 및 정책을 효율적으로 평가했으나 시뮬레이션 결과를 입력으로 사용하지 않아 Sim-Real 간 편차가 큰 경우 예측 오차가 커지는 한계가 있다. 반면, Illy-Net은 각 정책의 시뮬레이션-현실 특성 벡터 쌍을 기반으로 GMM을 학습하고 log-likelihood 기반 위험도를 통해 여러 정책을 동시에 정량 비교할 수 있다는 점에서 차별성을 가진다.

2.3 도메인 랜덤화

Tobin 등[15]은 객체 텍스처, 조명, 카메라 파라미터 등을 무작위화 하여 현실 세계의 변동성을 시뮬레이션에 반영하는 순수 Domain Randomization을 제안하였다. 그러나, 랜덤화 파라미터 수가 증가함에 따라 탐색 공간이 지수적으로 확장되어 수천~수만 회의 시뮬레이션 실행이 요구된다. Mehta 등[16]은 정보 이득에 기반한 Active DR기법을 통해 주요 랜덤화 변수만 선택함으로 효율을 향상시켰으나, 여전히 매 반복마다 온라인 평가가 필요하여 시뮬레이션 비용 부담이 크다. Illy-Net은 상대적으로 간결한 시뮬레이션 구성과 단일 현실 실험을 통해 효과적인 성능 평가가 가능하다는 점에서 실용적 이점을 갖는다.

2.4 도메인 적응

시뮬레이션과 현실 간의 시각적 차이를 축소하는 방식으로 James 등[17]은 GAN 기반 RCAN 모델을 통해 두 도메인을 공통



[Fig. 1] Conceptual illustration of domain adaptation in feature space. Initially, feature distributions from the source domain (orange) and the target domain (blue) are misaligned, leading to misclassification of target samples. Through domain adaptation, the feature distributions become better aligned within a shared latent space, significantly reducing—though not eliminating—classification errors

된 시각 공간에 정렬하였고, Bousmalis 등[18]은 Cycle-GAN을 활용하여 시뮬레이션 이미지를 포토리얼 형태로 변환함으로써 파지 정책을 학습하였다. 하지만 이러한 방식은 시각적 불일치는 줄일 수 있으나 마찰, 백래시, 구동 지연 같은 물리적 오차는 반영하지 못하며 개별 정책 간 성능을 정량 비교하는 데 한계가 있다. 이에 반해 Illy-Net은 actor activation 기반 행동 특성과 실험 성능 데이터를 통합 분석함으로써 시각·물리 도메인 간의 통합적 차이를 반영할 수 있도록 설계되었다[Fig. 1].

2.5 제안하는 방법의 이점

기존 연구들은 대규모 도메인 랜덤화, 반복적 시뮬레이터 보정, 픽셀 수준의 GAN 기반 변환, 혹은 특정 작업에 특화된 대리 모델에 의존해 왔다. 반면 Illy-Net은 시뮬레이션 기반 점수 s_i와 현실 위험도 r_i를 GMM 기반 log-likelihood로 통합함으로써, 단일 현실 실험만으로도 아직 평가되지 않은 정책의 성능을 통계적으로 예측할 수 있다. 이와 같은 방식은 대규모 시뮬레이션 롤아웃 없이도 빠르고 비용면에서 효율적인 정책 선별을 가능하게 하며, 향후 조립, 경로 계획, 장애물 회피 등 다양한 다운스트림 로봇 작업으로의 확장이 가능하다. 더불어 Illy-Net은 실험 자원이 제한된 상황에서 우선 검증해야 할 정책을 순차적으로 선정하는 적응형 실험 설계(active allocation)와 결합하기 쉽다. 위험 추정 모듈은 GMM에 한정되지 않으므로 향후 Gaussian Process, Studentt mixture, 혹은 딥 임베딩 기반 잠재 분포 추정으로 대체하여 불확실성 표현을 강화할 수 있다. 축적된 정책별 위험 로그를 분석하면 장비 마모나 센서 드리프트에 따른 성능 저하를 조기 경보하는 운영 지표로 재활용될 가능성도 있다.

[Table 1] Notion Table (Symbols and Acronyms)

Symbol / Acronym	Description
π_i	i-th candidate policy
s_i	Simulation score vector of policy π_i
$\hat{s}_i$	Min−max normalized simulation score
r_i	Raw risk = − log-likelihood of real-world features under GMM
$\hat{r}_i$	Z-score normalized risk
X_i	Final policy score: $X_i = \hat{s}_i \times \hat{r}_i$

[Table 1]은 본 논문 전반에서 사용된 주요 기호 및 약어를 명확성과 참조를 위해 요약한 것이다.

3. 방 법

3.1 작업 설명

본 연구에서 제안하는 Illy-Net 프레임워크는 총 다섯 단계로 구성된다: ① Object Identification → ② Grasp-Pose Estimation → ③ Grasp Execution → ④ Policy Optimization → ⑤ Illy-Net Evaluation. 모든 실험은 단일 real-world grasp batch를 기반으로 수행되며, 각 단계는 정책 전이의 신뢰도와 파지 성공률을 높이기 위한 일관된 파이프라인을 형성한다. 특히, 마지막 단계에서는 시뮬레이션과 현실에서 수집된 데이터를 활용해 정책의 위험도를 정량적으로 비교함으로써 최적의 후보를 효율적으로 선별할 수 있다.

3.1.1 Object Identification

Zivid Two+ RGB-D 카메라(hand-in-eye 구성)[19]를 활용해 장면을 촬영하고, SAM-6D[20] 알고리즘으로 물체를 분할한 후 6 자유도 자세 (x, y, z, roll, pitch, yaw)를 추정한다. 이를 hand-eye 보정 행렬을 통해 로봇 베이스 좌표계로 변환하여 파지계획에 활용될 변환행렬 $T_{base \leftarrow obj} \in SE(3)$을 얻는다.

3.1.2 Grasp-Pose Estimation

위 자세와 CAD 기반 포인트 클라우드를 Grasp-Net[21]에 입력하여 파지 후보를 생성한다. 표면 법선 및 곡률 정보를 기반으로 OnRobot RG6[22] 그리퍼의 파지 가능점을 추정하고, Soft-NMS[23]를 통해 상위 $k = 10$개의 파지 자세 $\{g1, \ldots, g10\}$를 출력한다. 이 정보는 Python-ROS 노드를 통해 실시간 전송된다.

3.1.3 Grasp Execution

선정된 파지 자세 중 최상위 후보 g1을 기반으로UR16e 로봇에 세분화된 궤적 명령을 전달한다: (i) 수직 상승, (ii) 선형 하강, (iii) 파지. RTDE 인터페이스를 통해 속도 0.25 m/s, 가속도 0.5 m/s² 조건에서 동작하며 force-torque 센서가 0.2초간 측정한 힘을 기준으로 파지 성공 여부를 판단한다. 실패 시 다음 후보로 전환하며 최대 5회 재시도한다.

3.1.4 Policy Optimization

DDPG[24] 알고리즘으로 정책을 학습하며 보상 함수는 다음과 같이 정의된다.

- 파지 성공 시: +1.0
- 파지 실패 시: −0.2
- 파지 자세 오차에 비례 시: 0 ~ −0.3
- 파지 시도 횟수 초과 시: −0.1

에이전트는 상태 s_t(물체 정보 + TCP pose)를 관측하여 행동 a_t(⊿TCP 이동)를 출력한다. Target network는 2,000 스텝마다 soft update 된다. 각 에피소드 후 n차원 시뮬레이션 점수 벡터 s_i와 512차원 actor 특성 벡터 f_i를 저장한다.

3.1.5 Illy-Net Evaluation

s_i와 f_i를 세 개의 검증 도메인 랜덤화 환경 DR_{val}, 별도의 전이 도메인 DR_j, 그리고 단일 실제 배치에서 수집한다. 각 정책별 시뮬레이션-현실 특성 쌍 (f_{sim},f_{real})로 GMM G_i를 학습하고, 실제 특성의 log-likelihood를 원시 위험도 r_i로 정의한다. 최종 점수는 $X_i = s_i \cdot r_i$ 으로 계산되어 상위 정책을 선별한다.

3.2 실험 방법

전체 파이프라인은 다음 세 단계로 구성된다: ① MDP 정의 및 DDPG 학습, ② DR 시뮬레이터 사전 훈련, ③ 단일 배치를 활용한 Illy-Net 보정.

3.2.1 MDP 정의

- 상태(s)

$$S \in \mathbb{R}^{13} = [F_x, F_y, F_z, x_{TCP}, y_{TCP}, z_{TCP}, \phi_{TCP}, \theta_{TCP}, \psi_{TCP}, g]^T \quad (1)$$

- 행동(a)

$$a=[\Delta x, \Delta y, \Delta z, \Delta g]^\top \in [-1,1]^4 \quad (2)$$

$$a_{real}=[D_{Tr}\Delta x, D_{Tr}\Delta y,, D_{Tr}\Delta z, Dg\Delta g] \quad (3)$$

D_{Tr}=40 mm, D_g=55 mm 으로 정의한다.

- 보상(R)

$$R=(1-\frac{F_r}{F_{Rmax}})+w(1-\frac{E_{pose}}{E_{max}})-\lambda t_{step} \quad (4)$$

$$w=0.5,\ F_{Rmax}=5,\ E_{max}=20\text{ mm},\ \lambda=0.05 \quad (5)$$

파지 성공 시 +1을 추가 보상한다.

- Discount factor γ =0.99 (6)

상태 벡터 $s \in \mathbb{R}^{13}$은 파지 상황에 대한 전반적인 정보를 포함하며, 세 축의 힘 센서 정보 (F_x,F_y,F_z), TCP의 위치 (x_{TCP},y_{TCP}, z_{TCP}) TCP의 오일러 각도 (ϕ_{TCP},θ_{TCP},ψ_{TCP}), 그리퍼 열림 상태 g 등 총 13개의 항목으로 구성된다. 모든 항목은 학습 안정성과 일반화를 위해 [0,1] 범위로 정규화된다.

행동 벡터 $a \in \mathbb{R}^4$는 TCP의 선형 이동 (Δx,Δy,Δz) 및 그리퍼의 개폐 정도 Δg를 나타내며, 각 항목은 [-1,1] 범위의 연속 값으로 표현된다. 이를 실제 로봇 제어 명령으로 변환하기 위해, 이동 스케일링 계수 D_{Tr}와 D_g 를 곱하여 실좌표계 기준의 이동 명령 $a^{real} \in \mathbb{R}^4$로 환산한다. 이때 D_{Tr}=40 mm, D_g=55 mm으로 설정하였다.

3.2.2 DDPG 학습

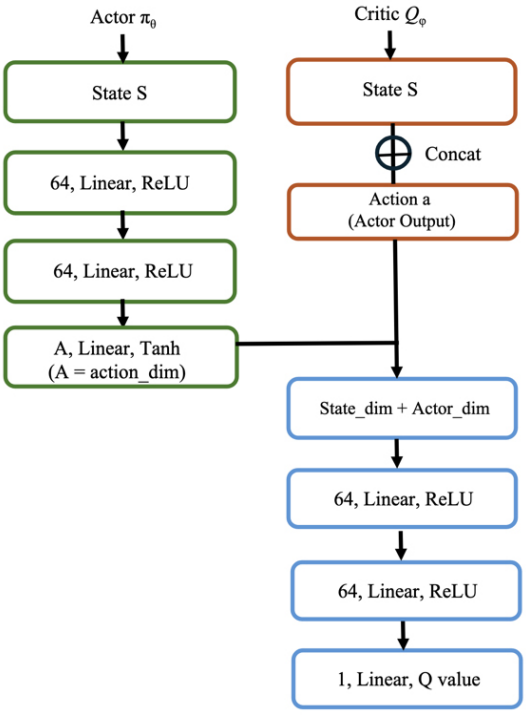
Actor 네트워크는 tanh 활성화를 갖는 64-unit Fully Connected (FC) 층 4개로 구성되며, Critic은 64-unit FC 층 3개로 구성되어 Q값을 추정한다. 학습에는 Adam Optimizer를 사용하며 학습률은 $\eta_A = \eta_C = 3\times10^{-4}$ 이다. 미니배치 크기는 256이며, 리플레이 버퍼 용량은 10^6 transition으로 설정하였다. 탐색 노이즈는 OU-process를 사용하며, 표준편차는 학습 진행에 따라 σ_0=0.3 → 0.01로 점진적으로 감소한다. 타겟 네트워크는 soft update 계수 τ=0.005를 적용하여 2,000 스텝마다 갱신된다. Actor 네트워크의 마지막 hidden layer(512차원)은 정책의 행동 특성을 대표하는 벡터로 활용된다[Fig. 2].

3.2.3 DR 사전학습

시뮬레이션-현실 도메인 간의 일반화 성능을 평가하기 위해 난이도에 따라 세 가지 도메인 랜덤화 환경을 설정하였다: DR1 (Easy), DR2 (Medium), DR3 (Hard). 각 환경에서 에피소드당 5회의 파지 시도를 포함하여 10,000 에피소드를 학습하였으며, 총 50개의 후보 정책 π_i를 생성하였다. 각 정책에 대한 시뮬레이션 로그는 다음과 같은 시뮬레이션 점수 벡터로 요약된다.

$$S_i=[\mu_{succ},\ 1-\frac{\mu_{pose}}{E_{max}},\ 1-\frac{\mu_{trial}}{F_{Rmax}},\ \text{norm}(\mu_R)] \quad (7)$$

여기서 μ_{succ}, μ_{pose}, μ_{trial}, μ_R은 각각 파지 성공률, 자세 오차, 시도 횟수, 보상 벡터의 평균값이다



[Fig. 2] The actor network (green) takes the normalized state vector s∈R^{13} as input and passes it through two fully connected layers with 64 units each and ReLU activation, followed by a final linear layer with tanh activation to output the action a∈R^4. The critic network (blue) receives both the state and the actor-generated action as input, concatenates them, and feeds the combined vector through two additional fully connected ReLU layers (64 units each), followed by a final linear layer to produce the scalar Q-value. The output of the last hidden layer of the actor (512 dimensions) is used as a compact policy feature vector for risk modeling in Illy-Net.
This structure enables stable value approximation while supporting feature extraction for downstream risk evaluation

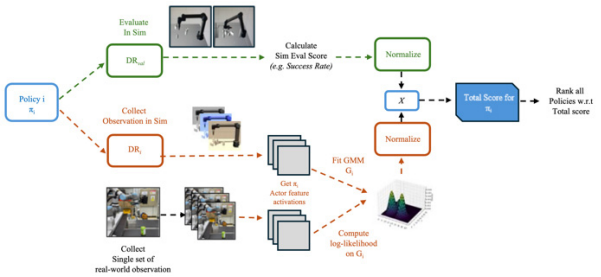
3.2.4 Illy-Net 평가 및 보정

• 실험 배치 수집

전체 50개 후보 정책 중 무작위로 25%를 선택하여 UR16e 로봇으로 각 정책에 대해 5회씩 실험을 수행한다. 이때 수집된 실제 행동 특성 벡터 f_i^{real}은 이후 위험도 추정에 활용된다.

• GMM Fitting

각 정책 i에 대해 시뮬레이션 특성 f_i^{sim}과 대응하는 현실특성f_i^{real}을 결합한 벡터 $z_i=[f_i^{sim}, f_i^{real}]$를 기반으로 3개의 혼합 성분을 갖는 GMM G_i를 학습한다. 현실 특성에 대한 GMM의 log-likelihood ℓ_i를 해당 정책의 원시 위험도(raw risk) $r_i = -\ell_i$로 정의한다. 미실험 정책은 f_i^{sim}만 사용한다. 이때 실험하지 않은 정책도 시뮬레이션 특징만으로 위험도를 추정받아 최종 점수 X_i 계산에 포함될 수 있다.



[Fig. 3] Overview of the Illy-Net evaluation pipeline: Each candidate policy π_i is evaluated in multiple domain-randomized simulation environments DR_{val} to compute a simulation score s_i, which is then normalized. In parallel, both simulation and a single batch of real-world executions are used to extract actor feature vectors. These vectors are combined to fit a Gaussian Mixture Model G_i, and the log-likelihood of the real-world feature under G_i is computed to obtain the raw risk r_i. The risk is also normalized. The final policy score $X_i = \hat{S}_i \cdot \hat{r}_i$ is calculated by multiplying the two normalized values. All candidate policies are ranked based on their X_i, and the top-ranked policies are selected for real-world validation

• Normalization

시뮬레이션 점수 벡터 $\hat{S}_i$는 min-max 정규화를 적용하고, 위험도 $\hat{r}_i$는 평균 0, 표준편차 1의 정규분포에 맞추어 Z-score 방식으로 정규화하여 $\hat{r}_i$를 얻는다.

• 최종점수

각 정책의 최종 스코어는 $X_i = \hat{S}_i \cdot \hat{r}_i$로 정의되며, 값이 큰 순으로 정책을 랭킹한다. 최상위 5개의 정책만을 실제 환경에서 최종 검증 대상으로 선정한다[Fig. 3].

3.2.5 Hyperparameter

Actor-critic 학습과 surrogate 기반 보정 단계에서 사용된 주요 Hyperparameter는 [Table 2]에 요약되어 있다[Table 2]. 별도 언급이 없는 한, 동일한 설정이 세 개의 도메인 랜덤화 시뮬레이터와 단일 실험 배치 기반의 현실 적응 과정 모두에 일관되게 적용된다.

[Table 2] Hyperparameter Setting

Parameter	Value
Actor-Critic Learning Rate	3×10^{-4}
Mini-batch Size (B)	256
Replay Buffer (D)	1M transition
OU Noise σ_0	0.3 → 0.01
Target Update Rate (τ)	0.005
GMM Components (K)	3 (300 EM iterations)

학습률 3×10^{-4}는 수렴 속도와 안정성의 균형을 맞추기 위해 경험적으로 설정되었다. 초기 탐색을 충분히 보장하고 점진적인 활용을 유도하기 위해, 표준편차 스케줄 $\sigma_0 = 0.3 \rightarrow 0.01$을 갖는 Ornstein-Uhlenbeck (OU) process를 사용하였다. Replay buffer는 최대 100만개의 전이(transition)를 저장하도록 구성되어 샘플 부족을 방지하면서도 GPU(그래픽 처리 장치) 메모리 한계를 초과하지 않도록 한다. 타겟 네트워크는 soft update 계수 $\tau = 0.005$를 사용하여 bootstrap 업데이트 시 점진적으로 변화하도록 설정하였다.

Surrogate 모듈에서는 구성요소 수 K = 3의 GMM을 기반으로 하며 300회 expectation-maximization (EM) 반복 학습을 통해 사전 grid search 결과 가장 적절한 likelihood 적합성과 계산 비용의 절충점을 제공하는 구성으로 설정되었다. 이러한 Hyperparameter 조합은 다섯 개의 랜덤 seed와 모든 작업 도메인에서 안정적인 학습곡선을 보이는 것으로 확인되었다. [Table 2]는 본 논문에서 사용된 주요 Hyperparameter를 요약한 것으로 Actor-Critic 학습과 Illy-Net 평가 단계 모두에 적용된다.

4. 실 험

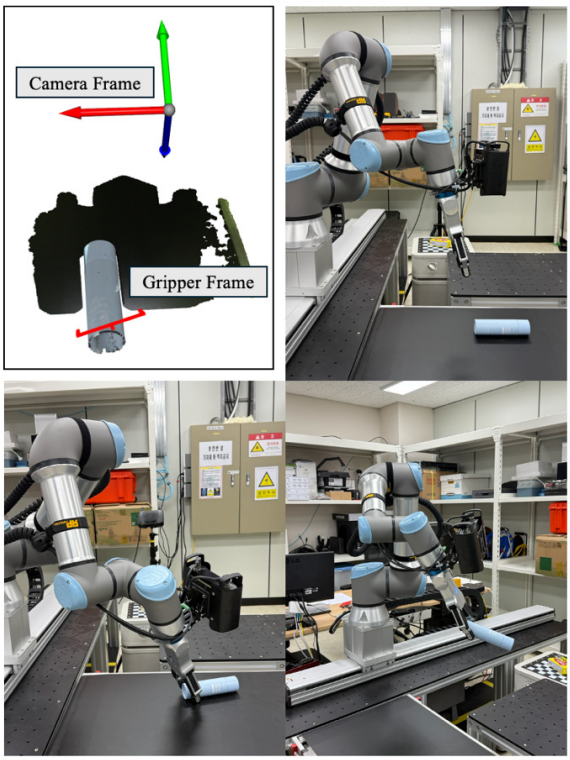
4.1 시스템 구성 및 데이터 수집

본 연구의 실험 플랫폼은 실제 생산 환경에 가까운 구성을 기반으로 하며, 시뮬레이션-현실 간 성능 차이를 분석하기 위한 정밀 계측 장비를 포함한다. 로봇으로는 UR16e (6자유도, 반복 정밀도 ±0.05 mm)를 사용하였으며, 2 m 길이의 Y축 전동 슬라이드에 장착해 1.8 m 폭의 컨베이어 전역을 커버하도록 구성하였다. 본 실험에서 Y-axis Slide(슬라이드 축)는 별도로 제어하지 않으며, TCP 좌표는 로봇 및 슬라이드 기준 절대 원점으로 환산하여 실험을 진행하였다.

엔드이펙터에는 hand-in-eye 방식으로 OnRobot RG6 그리퍼(최대 스트로크 110 mm)와 Zivid Two+ RGB-D 카메라(프레임레이트 45 Hz)를 탑재하였다. Hand-eye 보정은 least-squares 기반 Tsai-Lenz 알고리즘으로 수행되었으며, 평균 재투영 오차는 0.41픽셀이다. 제어용 PC는 Ubuntu 20.04, CUDA 12.4 환경에서 동작하며, Intel Xeon Silver 4410Y CPU, 512 GB DDR5 RAM(메모리), NVIDIA RTX 6000 Ada GPU 4장(VRAM 총 49 GB 상당, 비디오 메모리)으로 구성되어 있다.

3차원 물체 모델은 Artec Leo 스캐너(정밀도 0.1 mm)를 이용해 획득하고, Artec Studio 19에서 정합·후처리한 후 NVIDIA Isaac Sim으로 임포트하였다[Fig. 4]. 본 논문에서는 복잡한 형상을 직접 다루기보다는, 현실 산업 현장에서 빈번히 관측되는 기본 원형(Primitive) 형상—원통형, 사각기둥형, 마름모꼴—

[Fig. 4] Grasping Task Experimental Objects and Setup: Three representative object models used in the experiments—a cylindrical column (left), a rectangular prism (center), and a rhomboidal prism (right)—were selected to reflect common primitive shapes in industrial packaging. The setup illustrates the real-world robot workspace and sensing configuration



[Fig. 5] Real-World Grasping Pipeline and Execution Sequence: The top-left image visualizes grasp candidate generation using SAM-6D and point cloud alignment in the camera frame. The top-right image shows the initial object capture and pose estimation. The bottom-left depicts the UR16e robot executing a grasping motion toward the selected candidate. The bottom-right shows the object being lifted and moved after successful grasping. Each step corresponds to one phase in the real-world grasping episode

만을 선정하였다. 이는 향후 복잡한 형상도 해당 기본 형상의 조합으로 분해하여 적용할 수 있다는 전제 하에, 초기 실험 범위를 효율적으로 제한하기 위함이다.

Grasp 데이터셋은 초기 학습 안정성을 확보하기 위해 펜던트 조작(pendant teaching) 방식으로 수집되었다. 각 물체의 상단 및 측면에서 30° 간격으로 파지 자세를 샘플링하였고, 총 100개의 유효 파지(prime grasp) 궤적을 정의하였다. 각 자세는 로봇 기준 좌표계에서 object 변환행렬($T_{base \leftarrow obj}$)로 저장되며, 시뮬레이션과 실제 실험 간 정확한 정합을 보장한다

4.2 Real-World Episode

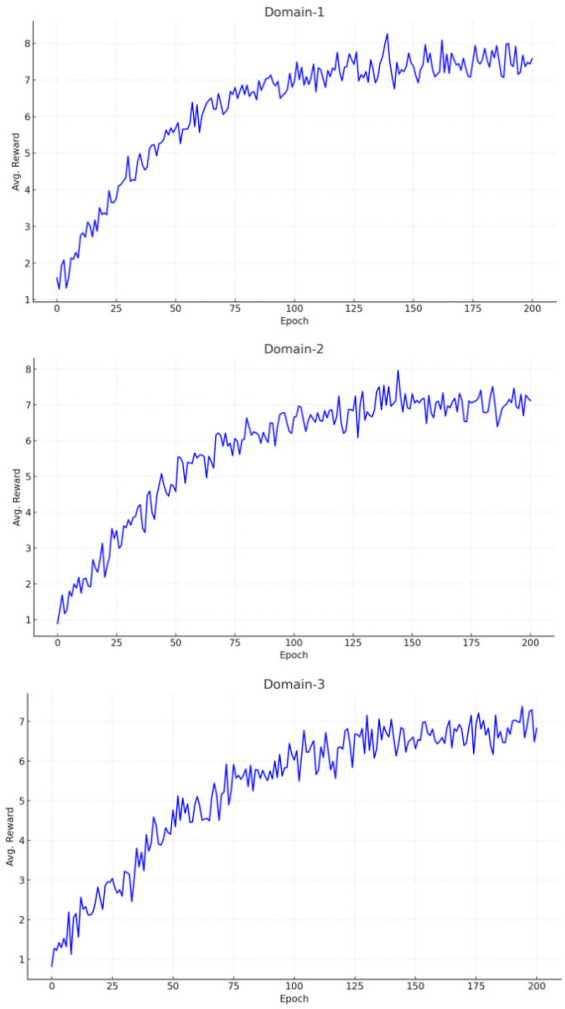
본 실험에서는 하나의 real-world episode를 네 단계로 구성하여 수행하였다[Fig. 5]. 첫째, Zivid Two+ RGB-D 카메라가 장면을 캡처한 뒤, SAM-6D 알고리즘이 물체를 분할하고 6자유도 자세(x, y, z, roll, pitch, yaw)를 추정한다. 이때 사전 보정된 hand-eye 행렬을 적용하여 카메라 기준의 자세를 로봇 베이스 좌표계로 변환하면, 평균 38 ms 내에 $T_{base \leftarrow obj}$가 계산된다. 둘째, 변환된 자세와 포인트 클라우드를 Grasp-Net에 입력하여 파지 후보 $\{g_k\}$를 생성한다. 파지 점 후보는 물체의 표면 법선 및 곡률 정보를 바탕으로 계산되며, 후보 생성 전체에 평균 190 ms가 소요된다. 셋째, 제어기는 최우선 파지 후보 g_1을 선택하여 URScript 명령으로 다음 세 단계의 궤적을 수행한다: (i) 수직 방향으로의 안전 상승, (ii) 선형 하강, (iii) 속도 0.25 m/s에서의 파지. 파지 실패 시에는 다음 후보 g_2로 전환하며, 최대 5회까지 반복 시도한다. 넷째, 각 에피소드 종료 후 actor 네트워크의 마지막 hidden layer에서 추출한 512차원 잠재 표현을 actor feature로 저장한다. 이 feature는 Illy-Net의 GMM 기반 보정에 사용된다.

4.3 시뮬레이터 실험

세 가지 난이도(쉬움·보통·어려움)의 도메인 랜덤화 환경(DR1, DR2, DR3)의 구성은 [Table 3]에 요약하였다. 각 환경에서 총 10,000 에피소드를 학습을 수행한 결과, 환경 복잡도가 증가할수록 탐색 성능의 분산이 증가하고 수렴 속도는 느려지는 경향을 보였다. 그럼에도 모든 도메인 환경에서 파지 성공률 85% 이상인 ε-optimal 정책을 도출하는데 성공하였다 [Fig. 6].

[Table 3] Domain Configuration

Domain	Object Spacing	Obstacles
Actor/Critic Learning Rate	Wide	None
Mini-batch Size (B)	Narrow	Single Obstacle
Replay Buffer (D)	Narrow with Random	Multiple Obstacle



[Fig. 6] Training Reward Curves for Domain-1, Domain-2, and Domain-3: Across all three domain-randomized environments, the average reward increases steadily over 200 epochs. Domain-1 (easy) shows rapid early convergence, while Domain-2 (medium) and Domain-3 (hard) exhibit slower learning due to increased variability. Nevertheless, all training curves stabilize after approximately 150 epochs, indicating successful convergence in each domain

- DR1(Easy): 약 1,000 스텝 내에 파지 성공률이 97% 이상으로 빠르게 수렴
- DR2(Medium): 학습 초기 안정성이 상대적으로 낮았으나 약 3,000 스텝 이후 94% 수준에서 안정적으로 수렴
- DR3(Hard): 학습 초기 분산이 $\sigma^2 \approx 0.18$로 크게 나타났으며, 10,000 스텝까지 점진적으로 향상되어 최종적으로는 86~89% 범위에서 수렴

이러한 사전 학습 과정을 통해 총 50개의 후보 정책 $(\pi_1, \pi_2, \ldots, \pi_{50})$을 확보하였으며 각 정책은 Illy-Net을 통한 후속 평가와 보정의 대상이 된다.

4.4 Illy-Net 보정 실험

Illy-Net 기반 보정 실험은 총 50개의 후보 정책 중 무작위로 25% (13개)를 선택하고, 각 정책당 5회의 실험을 수행하는 방식으로 구성되었다. 결과적으로 총 65회의 실제 파지 실험을 통해 현실 데이터 기반 보정을 진행하였다.

각 정책별로 시뮬레이션에서 수집된 512차원의 actor feature와, 실제 환경에서 획득한 동일 차원의 feature를 결합하여, 세 개의 성분을 갖는 GMM을 학습하였다. 이후 GMM을 통해 실제 actor feature의 log-likelihood 값을 계산하고, 이를 정책의 원시 위험도 $r_i = -\log P(f_i^{real} | G_i)$로 정의하였다.

다음으로, 시뮬레이션 점수 벡터 s_i는 0 ~ 1 범위로 min-max 정규화하고, 위험도 r_i는 Z-score로 표준화하였다. 이 두 값을 가중 내적하여 최종 점수 $X_i = \hat{S}_i \cdot \hat{r}_i$를 계산하고, 전체 정책을 이 점수 기준으로 일괄 랭킹하였다.

사전학습에 사용된 세 도메인(Domain1, 2, 3)의 학습 설정은 [Table 4]와 같이 구성되었으며, 각 도메인별로 5개의 독립 seed를 활용해 총 10,000 에피소드씩 학습하였다. Domain3의 경우 높은 복잡도를 반영해 더 큰 배치 크기와 더 많은 업데이트 주기를 설정하였다.

상위 5개의 정책을 선택하여 추가 실험을 진행한 결과, 파지 성공률은 단순 도메인 랜덤화 방식의 77.3%에서 91.7%로 14.4%p 향상되었다.

또한, 현실 실험을 통한 정책 평가 횟수는 기존의 250회에서 65회로 약 74% 감소하여, 성능 향상과 더불어 물리적 실험 비용 및 장비 마모를 크게 줄일 수 있었다. 이로써 Illy-Net은 실험 효율성과 정책 신뢰도 모두를 확보하는 보정 기법으로 유의미한 효과를 입증하였다. 더 나아가, 동일한 실험 예산 하에서 더 많은 후보 정책을 사전 평가할 수 있으므로 초기 정책 탐색의 폭을 넓히면서도 물리적 검증 부담을 통제할 수 있다. 또한 GMM 기반 위험도 추정 모듈은 모듈식 구조로 구현되어 있어 제조 라인·작업물·센서 구성 변경 시에도 최소 재학습으로 재사용 가능하다. 축적된 정책별 점수 ($\hat{s}_i$, $\hat{r}_i$, X_i)기록은 장기 분석을 통해 장비 마모나 캘리브레이션 드리프트를 조기 감지하는 운영 모니터링 지표로 확장될 잠재력도 가진다.

[Table 4] Learning Configuration

Parameter	Domain1	Domain2	Domain3
Max Episode Length	200	200	200
Total Episode	10,000	10,000	10,000
Independent Seeds	5	5	5
Learning Rate	0.0003	0.003	0.003
Batch Size	64	64	128
Discount Factor	0.99	0.99	0.99
GAE	0.95	0.95	0.95
Epochs/Updates	10	10	15
Clip Range	0.2	0.2	0.15

5. 결 과

본 연구에서는 대규모 시뮬레이션 평가와 경량(real-world) 대리 보정을 결합하여, 제한된 실험 데이터만으로도 신뢰성 높은 정책 선택을 가능하게 하는 계층형 이중 프레임워크 Illy-Net을 제안하였다. 첫 번째 흐름에서는 각 내재적 정책 π_i를 조명, 레이아웃 난이도, 마찰 계수 등 다양한 외재적 도메인 변형 환경에서 실행하여, 장면별 정규화된 스칼라 성능 지표를 산출하고 이를 시뮬레이션 점수 벡터 S_i로 구성하였다.

두 번째 흐름에서는 정책당 5회의 실제 파지 실험을 통해 실측 점수 $\zeta(e_{rw}, \pi_i)$를 얻고, Actor 네트워크의 잠재 벡터(또는 PCA·Autoencoder로 축소한 특징)을 이용해 GMM을 적합하여 log-likelihood 기반의 위험도를 계산하였다. 이때, 실험되지 않은 정책이라도 시뮬레이션 특징만으로 위험 추정이 가능하다는 점이 핵심이다. 두 흐름은 계층형 곱셈 스코어를 통해 통합되며, 시뮬레이션의 폭넓은 탐색 효율과 실제 실험의 고충실도 보정 신호를 동시에 활용할 수 있게 한다.

실제 적용 가능성을 고려하여, Variational GP나 Student-t mixture 등의 기법을 도입함으로써 근사 오차를 줄이고 예측 견고성을 강화할 수 있다. 실험 결과 Illy-Net은 실험되지 않은 정책의 75%에 대해 MAPE 8% 미만으로 성공률을 예측하였고, 단 65회의 실험만으로 Top-5 파지 성공률을 77.3%에서 91.7%로 향상시켜, 실험 횟수를 74% 절감하면서도 효율을 네 배 이상 높이는 성과를 보였다.

종합적으로 Illy-Net은 시뮬레이션 점수와 실환경 위험도를 계층적으로 통합하여 정보 기반(policy ranking) 메커니즘을 제공한다. 단일 배치 실험만으로 도메인 간 간극을 보정할 수 있으며, 인코더와 대리 모델은 모듈식 구조로 교체가 가능하여 비용 민감형 산업 로봇 환경에도 실용적이고 확장 가능한 솔루션이 될 수 있다.

향후 연구에서는 파지 성공률과 자세 오차 뿐 아니라 로봇 궤적의 충돌 위험과 모션 비용 등 path-planning 지표를 점수 벡터에 통합할 예정이다. 이를 위해 RRT*나 T-RRT 같은 샘플링 기반 플래너를 이용하여 이동 거리 및 에너지 소비를 추정하고, 파지와 경로를 동시에 최적화하는 접근을 시도할 계획이다. 또한 시뮬레이터의 현실 정합성을 높이기 위해 기존의 도메인 랜덤화 방식 대신, 실제 계측한 조도(lux), 마찰 계수, 질량, 관성 등의 물리 파라미터를 시뮬레이터에 직접 주입하고, SimOpt 기반 피드백 루프를 통해 분포를 주기적으로 보정하는 방식으

로 전환할 계획이다. 마지막으로, 잠재 표현 학습 강화를 위해 기존 Actor 네트워크의 hidden layer 대신 self-supervised 인코더와 contrastive learning을 적용하여 보다 풍부하고 구별력 있는 임베딩을 학습하며, 대리 모델은 Variational Gaussian Process나 Student-t mixture로 확장하여 예측의 견고성과 일반화를 더욱 높일 계획이다.

References

[1] L. Zheng, W. Ma, Y. Cai, T. Lu, and S. Wang, "GPDAN: Grasp-pose domain adaptation network for sim-to-real 6-DoF grasping," *IEEE Robotics and Automation Letters*, vol. 8, no. 8, pp. 4585-4592, Aug., 2023, DOI: 10.1109/LRA.2023.3286816.

[2] J. Huber, F. Hélénon, H. Watrelot, F. B. Amar, and S. Doncieux, "Domain randomization for Sim2real transfer of automatically generated grasping datasets," *2024 IEEE International Conference on Robotics and Automation (ICRA)*, Yokohama, Japan, pp. 4112-4118, 2024, DOI: 10.1109/ICRA57147.2024.10610677.

[3] Y. Z. Ren, H. Dai, B. Burchfiel, and A. Majumdar, "AdaptSim: task-driven simulator adaptation for efficient sim-to-real transfer," *Conference on Robot Learning (CoRL)*, vol. 229, pp. 3434-3452, 2023, [Online], https://proceedings.mlr.press/v229/ren23b.html.

[4] H. Dong, H. Fu, W. Xu, Z. Zhou, and C. Chen, "EASI: evolutionary adversarial simulator identification for robotic manipulation," *Advances in Neural Information Processing Systems 37 (NeurIPS)*, 2024, [Online], https://proceedings.neurips.cc/paper_files/paper/2024/hash/0caf026b344e6c455efc12fe3d254e9f-Abstract-Conference.html.

[5] X. Zhang, C. Wang, L. Sun, Z. Wu, X. Zhu, and M. Tomizuka, "Efficient sim-to-real transfer of contact-rich manipulation skills with online admittance residual learning" *arXiv:2310.10509,* 2023, DOI: 10.48550/arXiv.2310.10509.

[6] Y. Jiang, C. Wang, R. Zhang, J. Wu, and L. Fei-Fei, "TransiC: sim-to-real policy transfer by learning from online correction" *arXiv:2405.10315*, 2024, DOI: 10.48550/arXiv.2405.10315.

[7] J. F. Cai, Z. Chen, X.-M. Wu, J.-J. Jiang, Y.-L. Wei, and W.-S. Zheng, "Real-to-sim grasp: rethinking the gap between simulation and real world in grasp detection," *arXiv:2410.06521*, 2024, DOI: 10.48550/arXiv.2410.06521.

[8] C. Li, K. Shi, K. Zhou, H. Wang, J. Zhang, and H. Dong, "RGBGrasp: image-based object grasping by capturing multiple views during robot arm movement with Neural Radiance Fields," *arXiv:2311.16592*, 2024, DOI: 10.48550/arXiv.2311.16592.

[9] A. D. Vuong, M. N. Vu, B. Huang, N. Nguyen, H. Le, T. Vo, and A. Nguyen, "Language-driven grasp detection," *arXiv:2406.09489*, 2024, DOI: 10.48550/arXiv.2406.09489.

[10] A. Yu, A. Foote, R. Mooney, and R. Martín-Martín, "Natural language can help bridge the sim-to-real gap," *arXiv:2405.10020*, 2024, DOI: 10.48550/arXiv.2405.10020.

[11] P. Huang, X. Zhang, Z. Cao, S. Liu, M. Xu, W. Ding, J. Francis, B. Chen, and D. Zhao, "What went wrong? closing the sim-to-real gap via differentiable causal discovery," *arXiv:2306.15864*, 2023, DOI: 10.48550/arXiv.2306.15864.

[12] Y. Chebotar et al., "Closing the sim-to-real loop: adapting simulation randomization with real-world experience," *2019 International Conference on Robotics and Automation (ICRA)*, Montreal, QC, Canada, pp. 8973-8979, 2019, DOI: 10.1109/ICRA.2019.8793789.

[13] F. Muratore, F. Treede, M. Gienger, and J. Peters, "Domain randomization for simulation-based policy optimization with transferability assessment," *The 2nd Conference on Robot Learning,(PMLR)*, 2018, [Online], https://proceedings.mlr.press/v87/muratore18a.html.

[14] A. Anwar, R. Gupta, Z. Merchant, S. Ghosh, W. Neiswanger, and J. Thomason, "Efficient evaluation of multi-task robot policies with active experiment selection," *arXiv:2502.09829*, 2025, DOI: 10.48550/arXiv.2502.09829.

[15] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Vancouver, BC, Canada, pp. 23-30, 2017, DOI: 10.1109/IROS.2017.8202133.

[16] B. Mehta, M. Diaz, F. Golemo, C. J. Pal, and L. Paull, "Active domain randomization," *arXiv:1904.04762*, 2019, DOI: 10.48550/arXiv.1904.04762.

[17] S. James, P. Wohlhart, M. Kalakrishnan, D. Kalashnikov, A. Irpan, J. Ibarz, S. Levine, R. Hadsell, and K. Bousmalis, "Sim-to-real via sim-to-sim: data-efficient robotic grasping via randomized-to-canonical adaptation networks," *arXiv:1812.07252*, 2019, DOI: 10.48550/arXiv.1812.07252.

[18] K. Bousmalis et al., "Using simulation and domain adaptation to improve efficiency of deep robotic grasping," *2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, QLD, Australia, pp. 654-661, 2018, DOI: 10.1109/ICRA.2018.8460875.

[19] Zivid, "Zivid Two+ 3D Camera – Datasheet," [Online], https://www.zivid.com/hubfs/User%20guides%20and%20datasheets/Zivid%202+%20M60%20Datasheet.pdf, Accessed: Apr. 20, 2025.

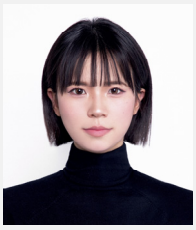
[20] J. Lin, L. Liu, D. Lu, and K. Jia, "SAM-6D: segment anything model meets zero-shot 6-D object pose estimation," *arXiv:2311.15707*, 2024, DOI: 10.48550/arXiv.2311.15707.

[21] J. Pang, M. A. Lodhi, and D. Tian, "Grasp-Net: geometric residual analysis and synthesis for point cloud compression," *arXiv:2203.01923*, 2022, DOI: 10.48550/arXiv.2209.04401.

[22] OnRobot, *RG6 Gripper – Datasheet*, [Online], https://onrobot.com/en/products/rg6-finger-grippert, Accessed: Apr. 20, 2025.

[23] N. Bodla, B. Singh, R. Chellappa, and L. S. Davis, "Soft-NMS – Improving object detection with one line of code," *2017 IEEE International Conference on Computer Vision (ICCV),* Venice, Italy, 2017, pp. 5562-5570, DOI: 10.1109/ICCV.2017.593.

[24] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra, "Continuous control with deep reinforcement learning," *arXiv:1509.02971*, 2015, DOI: 10.48550/arXiv.1509.02971.



임 수 빈

2023 국립강릉원주대학교 전자공학과(학사)

2023~현재 성균관대학교 지능형로봇학과 (석사)

2024~현재 한국생산기술연구원 학생연구원

관심분야: Robot Manipulator, A.I./Deep Learning, RL

이 재 선

2000 고려대학교 전자공학과(공학사)

2006 한국과학기술원 디지털미디어전공 (공학석사)

2013~현재 서울대학교 전기컴퓨터공학부 (공학박사수료)

2011~현재 한국생산기술연구원 중소제조공정 지원팀장

관심분야: HCI/HRI, A.I./Machine Learning, UX/UI