

차세대 서비스 로봇틱스를 위한 심투리얼(**Sim-to-Real**) 자율 주행 및 시맨틱 내비게이션 구축 전략: **NVIDIA Isaac Sim**과 생성형 **AI**의 융합

서론: **2026**년 서비스 로봇틱스의 기술적 패러다임 변화와 인재상

로봇틱스 산업은 현재 단순한 기계적 자동화를 넘어 데이터 기반의 인지(Perception)와 판단(Cognition)이 결합된 '소프트웨어 정의 로봇틱스(Software-Defined Robotics, SDR)'로 급격히 재편되고 있습니다. 특히 귀하가 1지망으로 목표하고 있는 베어로보틱스(Bear Robotics)는 단순한 서빙 로봇 제조사를 넘어, LG전자와의 전략적 파트너십을 통해 모바일 로봇 플랫폼의 표준을 제시하려는 거대한 기술적 도약을 시도하고 있습니다.¹ 이러한 시점에서 단순한 개발자가 아닌 **AI**와 로봇틱스를 아우르는 **R&D 엔지니어**로서의 역량을 증명하기 위해서는, 학부 수준의 정적인 프로젝트를 넘어선, 산업계의 난제를 해결할 수 있는 '엣지 있는' 포트폴리오가 필수적입니다.

귀하가 보유한 **NVIDIA RTX A6000**이라는 강력한 하드웨어 자산은 일반적인 지원자들이 접근하기 힘든 고차원의 시뮬레이션과 거대 **AI** 모델의 운용을 가능하게 하는 결정적인 무기입니다.³ 본 보고서는 귀하의 하드웨어 이점을 극대화하고, 베어로보틱스가 직면한 기술적 난제인 '동적 환경에서의 사회적 주행(Socially Aware Navigation)'과 '시맨틱 이해(Semantic Understanding)'를 해결하기 위한 구체적인 기술 로드맵을 제시합니다.

기존에 구상했던 프로젝트가 정적인 이미지 분석과 단순한 **ROS** 주행에 머물러 있었다면, 본 보고서에서 제안하는 전략은 **NVIDIA Isaac Sim**을 기반으로 한 고정밀 시뮬레이션 환경 구축, ****비전-언어 모델(Vision-Language Models, VLM)****을 활용한 인지 능력 강화, 그리고 확산 모델(**Diffusion Policy**) 기반의 동작 생성을 포함합니다. 이는 베어로보틱스의 채용 담당자와 엔지니어링 팀에게 귀하가 단순히 도구를 사용할 줄 아는 지원자가 아니라, 차세대 로봇 지능(Embodied AI)의 흐름을 주도할 수 있는 인재임을 증명하는 강력한 근거가 될 것입니다.

1. 베어로보틱스의 기술적 지향점과 채용 니즈 분석

성공적인 포트폴리오는 기업의 기술적 갈증을 정확히 타격해야 합니다. 베어로보틱스의 최근 행보와 기술 블로그, 채용 공고, 그리고 업계 동향을 종합적으로 분석해 볼 때, 이들이 해결하고자 하는 문제는 크게 세 가지로 요약됩니다.

1.1. 하드웨어 중심에서 소프트웨어 정의 로봇틱스(SDR)로의 전환

베어로보틱스는 LG전자로부터 6천만 달러 규모의 전략적 투자를 유치하며, 로봇의 하드웨어 제약을 소프트웨어로 극복하고 다양한 기종의 로봇을 통합 관리할 수 있는 플랫폼 기업으로 진화하고 있습니다.¹ 이는 단순히 로봇 하나를 움직이는 것을 넘어, 수십 대의 로봇이 혼잡한 식당 환경에서 서로 통신하고 경로를 조율하는 플릿 매니지먼트(Fleet Management) 기술이 핵심 경쟁력임을 시사합니다.⁵

- 시사점: 귀하의 프로젝트는 단일 로봇의 주행에 그쳐서는 안 되며, 다중 로봇(Multi-Robot) 환경을 고려한 아키텍처를 포함해야 합니다. 이는 ROS 2의 분산 통신 구조를 이해하고 있음을 보여주는 좋은 기회입니다.

1.2. 동적 장애물과 사회적 주행(Social Navigation)의 난제

베어로보틱스의 주력 제품인 '서비(Servi)'와 '서비 플러스(Servi Plus)'는 통제된 공장이 아닌, 예측 불가능한 사람들이 가득한 식당에서 운용됩니다.⁶ 기존의 전통적인 내비게이션 알고리즘(예: DWA, A*)은 사람을 단순한 '장애물'로 인식하여 급정거하거나 경로를 찾지 못하고 멈추는 '동결 로봇 문제(Freezing Robot Problem)'를 자주 일으킵니다.⁸

- 시사점: 베어로보틱스는 사람의 움직임을 예측하고, 사회적 규범(예: 사람 뒤로 지나가기, 마주칠 때 우측통행하기)을 준수하는 주행 알고리즘을 원합니다. 이를 위해 딥러닝 기반의 강화학습(RL)이나 모방학습(Imitation Learning)을 적용한 프로젝트는 매우 높은 평가를 받을 수 있습니다.⁹

1.3. 멀티 센서 퓨전과 고정밀 측위(Localization)

최신 서비 로봇은 LiDAR뿐만 아니라 깊이(Depth) 카메라와 RGB 카메라를 탑재하고 있습니다.¹⁰ 이는 LiDAR가 놓칠 수 있는 낮은 장애물이나 투명한 유리, 그리고 복잡한 시맨틱 정보(테이블 번호, 고객의 손짓 등)를 인식하기 위함입니다.

- 시사점: 귀하의 프로젝트는 단일 센서에 의존하지 않고, **비전(Camera)과 라이다(LiDAR)를 융합(Sensor Fusion)**하는 기술을 보여주어야 합니다. 특히 RTX A6000의 강력한 연산 능력을 활용하여 비전 데이터를 실시간으로 처리하는 파이프라인을 구축하는 것은 강력한 차별화 포인트가 됩니다.³

2. 시뮬레이션 플랫폼의 전략적 선택: Gazebo를 버리고 Isaac Sim으로

귀하의 초기 구상에서 가장 시급하게 개선해야 할 점은 시뮬레이터의 선택입니다. 2026년 시점, 특히 RTX A6000이라는 하이엔드 GPU를 보유한 상황에서 Gazebo(Classic)를 사용하는 것은 페라리 엔진을 경운기에 장착하는 것과 같습니다. 베어로보틱스와 같은 선도 기업들은 이미 NVIDIA Isaac Sim 기반의 워크플로우로 전환하고 있습니다.¹¹

2.1. 기술적 우위: 물리 엔진과 렌더링 품질

Gazebo는 2000년대 초반 기술인 ODE(Open Dynamics Engine) 물리 엔진과 OGRE 렌더링 엔진을 사용합니다. 이는 CPU 의존도가 높아 RTX A6000의 성능을 10%도 활용하지 못하며, 시각적 충실도가 낮아 AI 학습용 데이터로 부적합합니다.³

반면, Isaac Sim은 NVIDIA의 PhysX 5.0 물리 엔진과 RTX 레이 트레이싱(Ray Tracing) 기술을 사용합니다.

- **GPU 가속: A6000의 48GB VRAM**을 온전히 활용하여, 수십 대의 로봇과 수백 개의 동적 객체를 실시간으로 시뮬레이션할 수 있습니다. 6개 이상의 카메라 센서를 장착해도 120 FPS 이상의 퍼포먼스를 유지합니다.³
- **포토리얼리즘(Photorealism):** 실제 빛의 반사, 굴절, 재질감을 완벽하게 모사합니다. 이는 컴퓨터 비전 모델을 학습시킬 때 'Sim-to-Real(시뮬레이션에서 학습한 모델을 현실에 적용할 때 성능이 떨어지는 현상)' 격차를 획기적으로 줄여줍니다.¹²

2.2. AI 파이프라인의 통합성

Gazebo에서 딥러닝 모델을 돌리려면 ROS Bridge를 통해 이미지 데이터를 CPU로 빼내고, 다시 파이썬 스크립트로 변환하여 GPU로 보내는 비효율적인 과정을 거쳐야 합니다. 이는 막대한 지연(Latency)을 발생시킵니다.

Isaac Sim은 Omniverse 플랫폼 위에서 구동되며, 시뮬레이션 데이터가 GPU 메모리 상에서 직접 PyTorch 텐서로 변환됩니다. 즉, 시뮬레이션부터 AI 추론까지 데이터가 GPU를 떠나지 않는 Zero-Copy 파이프라인이 가능합니다. 이는 귀하가 목표로 하는 "딥러닝 방법론의 100% 이해와 적용"을 증명하기에 최적의 환경입니다.³

2.3. 베어로보틱스와의 적합성

최근 베어로보틱스와 협력하는 많은 로보틱스 기업들이 NVIDIA의 Isaac 생태계(Isaac ROS, Isaac Sim)를 채택하고 있습니다.¹¹ 포트폴리오 영상에서 Gazebo의 투박한 그래픽 대신, Isaac Sim의 영화 같은 렌더링 품질과 물리 정확도를 보여주는 것은 면접관에게 "즉시 전력감"이라는 인상을 심어줄 것입니다.

3. 프로젝트 상세 제안: "시맨틱-소셜 내비게이션을 위한 엣지 AI 플릿 시스템"

이제 분석된 니즈와 도구를 바탕으로, 귀하의 포트폴리오를 위한 구체적이고 창의적인 프로젝트를 정의합니다.

3.1. 프로젝트 개요 및 차별화 포인트

프로젝트명: *Sim-to-Real Autonomous Restaurant Fleet: Visual-Language Navigation & Diffusion-Based Social Compliance*

이 프로젝트는 기존의 "지도상의 좌표 A에서 B로 이동"하는 단순한 내비게이션을 넘어섭니다.

대신, 인간의 언어 명령을 이해하고, 시각적으로 환경을 해석하며, 사람과 자연스럽게 상호작용하는 지능형 로봇을 구현합니다.

핵심 차별화 요소:

- 1. 시각-언어 모델(VLM) 기반 명령 수행: "주방으로 가라"는 좌표 명령 대신, "*"창가 쪽 테이블에 물이 쏟아졌으니 가서 치워줘"*라는 자연어 명령을 내리면, 로봇이 카메라로 '창가 테이블'과 '물 쏟아짐'을 인식하여 스스로 목표 지점을 설정합니다.¹⁵
- 2. 확산 모델(Diffusion Policy) 기반 주행: 복잡한 인파 속에서 로봇이 정지하지 않고 유연하게 회피 주행을 하도록, 최신 생성형 AI 기술인 Diffusion Model을 제어 정책(Policy)으로 사용합니다.¹⁷
- 3. 생성형 월드 구축(Generative World Building): Isaac Sim의 Python API를 활용하여, 매 시뮬레이션마다 식당의 구조, 테이블 배치, 조명 환경이 무작위로 바뀌는 '절차적 환경 생성'을 구현합니다. 이는 AI 모델의 일반화(Generalization) 성능을 입증하는 강력한 증거가 됩니다.¹⁹

3.2. 시스템 아키텍처 설계 (RTX A6000 최적화)

귀하의 로컬 노트북과 원격 GPU 서버(A6000)를 연결하는 하이브리드 아키텍처는 실제 로봇 개발 환경(로봇 본체와 관제 PC)과 유사하여 실무 역량을 어필하기 좋습니다.

구성 요소	역할 및 기술 스택	설명
Simulation Server (A6000)	NVIDIA Isaac Sim (Headless Mode)	식당 환경, 로봇 동역학, 센서(LiDAR, RGB-D) 시뮬레이션. GPU 가속 물리 엔진 구동.
AI Compute (A6000)	PyTorch & Isaac Lab	VLM(시각-언어 모델) 추론, Diffusion Policy 학습 및 실행. 48GB VRAM을 활용한 대규모 모델 구동.
Middleware	ROS 2 Humble	로봇 제어, 센서 데이터 통신, 플릿 관제 통신. DDS 기반의 실시간 데이터 교환.
Local Client (Laptop)	Omniverse Streaming Client	서버의 시뮬레이션 화면을 저지연으로 스트리밍하여 시각화. 개발 및 디버깅 인터페이스(VS Code).

4. 단계별 구현 로드맵 및 기술적 깊이

이 프로젝트는 5-7일 정도의 집중적인 개발 기간을 가정하여 설계되었습니다. AI의 도움을 받아 충분히 수행 가능하도록 단계별 가이드를 제공합니다.

Phase 1: 개발 환경 구축 및 Isaac Sim 입문 (Day 1-2)

가장 먼저 할 일은 로컬 노트북에서 A6000 서버를 마치 내 컴퓨터처럼 쓸 수 있는 환경을 만드는 것입니다.

- **원격 접속 설정:** VS Code의 Remote - SSH 확장을 사용하여 서버에 접속합니다. 이때, Isaac Sim의 GUI를 로컬에서 보기 위해 **Omniverse Streaming Client**를 활용합니다. 이는 서버에서 렌더링된 화면을 비디오 스트림으로 노트북에 전송해주므로, 고사양 그래픽 작업도 끊김 없이 가능합니다.²⁰
- **Isaac Sim 설치 및 Python 환경:** 서버에 Isaac Sim을 컨테이너 또는 워크스테이션 버전으로 설치합니다. Isaac Sim은 파이썬과 완벽하게 통합되므로, 복잡한 C++ 컴파일 없이도 로봇을 제어할 수 있습니다.³
- **기본 예제 실행:** Isaac Examples > ROS2 > Navigation 예제를 실행하여 로봇이 가상 환경에서 움직이는 것을 확인합니다.

Phase 2: 절차적 식당 환경 생성 (Day 3)

단순히 제공된 맵을 쓰는 것은 포트폴리오로서 매력이 떨어집니다. Python 스크립트로 식당을 '생성'하십시오.

- **Procedural Generation:** Isaac Sim의 replicator 모듈이나 Core API를 사용하여 테이블, 의자, 주방 집기 등을 랜덤하게 배치하는 스크립트를 작성합니다. "테이블 간격은 최소 1m 유지", "주방은 구석에 배치"와 같은 규칙을 코드에 적용합니다.
- **시사점:** 이는 베어로보틱스의 로봇이 다양한 식당 구조에 적응해야 하는 현실적인 문제를 시뮬레이션 레벨에서 해결하려는 시도로 해석됩니다.²¹

Phase 3: VLM 기반 시맨틱 내비게이션 구현 (Day 4-5)

여기가 귀하의 딥러닝 역량을 보여줄 핵심 파트입니다.

- **비전-언어 모델(VLM) 통합:** 오픈소스로 공개된 경량화 VLM(예: NanoLLaVA 또는 CLIP-Seg)을 A6000에 로드합니다. A6000의 VRAM이 넉넉하므로, 시뮬레이션과 동시에 무거운 AI 모델을 돌려도 성능 저하가 없습니다.³
- **로직 구현:**
 1. 사용자(노트북)가 텍스트 명령 입력: "빈 테이블을 찾아줘."
 2. 로봇의 RGB 카메라 이미지가 VLM에 입력됨.
 3. VLM이 이미지 내에서 '빈 테이블' 영역을 세그멘테이션(Segmentation) 하거나 바운딩 박스로 검출.
 4. 검출된 픽셀 좌표를 3D 월드 좌표로 변환(Depth 센서 활용)하여 내비게이션

- 목표점(Goal)으로 설정.
5. ROS 2의 `/goal_pose` 토픽으로 좌표 전송.

Phase 4: 확산 모델(Diffusion Policy) 기반 주행 (Day 6)

전통적인 경로 계획법 대신, 최신 AI 트렌드인 **Generative AI for Control**을 적용합니다.

- 데이터 수집: Isaac Sim에서 로봇을 직접 조종(Teleoperation)하여 사람을 피하고 자연스럽게 주행하는 데이터를 약 30분~1시간 분량 수집합니다.
- 모델 학습: 이 데이터를 바탕으로 **Diffusion Policy** 모델을 학습시킵니다. 이 모델은 현재의 관측 데이터(Lidar, Camera)를 입력받아, 다음에 취해야 할 일련의 동작(Velocity Command)을 '생성'해냅니다.
- 장점: 기존 알고리즘은 장애물 앞에서 멈칫거리는 경우가 많지만, 확산 모델은 사람의 주행 패턴을 모방하여 훨씬 유려하고 사회적으로 용인되는(Socially Compliant) 회피 기동을 보여줍니다.¹⁷

Phase 5: 시각화 및 데모 제작 (Day 7)

포트폴리오의 가치는 보여지는 것에서 결정됩니다.

- **Cinematic Rendering:** Isaac Sim의 'Movie Capture' 기능을 사용하여, 로봇의 시점과 전지적 시점(Top-down view)을 4K 해상도로 렌더링합니다. 이때 레이 트레이싱을 켜서 실사 영화 같은 퀄리티를 뽑아냅니다.²²
- **Data Overlay:** 영상 위에 로봇이 인식하고 있는 시맨틱 정보(예: "사람 감지: 회피 모드 작동", "목표물: 3번 테이블")를 그래픽으로 오버레이하여 로봇의 '생각'을 시각화합니다.

5. 기술적 심층 분석: 왜 이 프로젝트가 특별한가?

이 섹션에서는 귀하가 면접이나 기술 문서에서 강조해야 할 "Deep Insight"를 다룹니다.

5.1. 딥러닝 방법론의 구체적 적용: 확산 모델의 제어 적용

단순히 "딥러닝을 썼다"가 아니라, ***"왜 확산 모델인가?"***에 대한 논리를 갖추어야 합니다.

- 멀티모달 분포(Multimodal Distribution) 처리: 좁은 통로에서 사람이 다가올 때, 로봇은 왼쪽으로 피하거나 오른쪽으로 피할 수 있습니다. 전통적인 회귀(Regression) 모델은 이 두 가지 선택의 '평균'인 중앙으로 가다가 충돌합니다. 하지만 확산 모델은 데이터 분포를 학습하므로, 왼쪽이나 오른쪽 중 확실한 하나의 경로를 선택하여 생성할 수 있습니다.¹⁷ 이 설명을 곁들이면 귀하가 딥러닝의 본질을 꿰뚫고 있다는 인상을 줍니다.

5.2. Sim-to-Real 격차 해소 전략: 도메인 랜덤화(Domain Randomization)

시뮬레이션 프로젝트의 가장 큰 약점인 "현실에서도 되나요?"라는 질문에 대한 방어 논리입니다.

- 전략: Isaac Sim의 **Replicator**를 사용하여 텍스처, 조명, 카메라 노이즈, 바닥의 마찰 계수

등을 매 프레임마다 무작위로 변경하며 학습시킵니다.¹²

- 어필: "저는 단순히 시뮬레이션에서 잘 되는 모델이 아니라, 현실의 불확실성을 견딜 수 있는 강건한(**Robust**) 모델을 만들기 위해 도메인 랜덤화를 적용했습니다."라고 기술하십시오.

5.3. ROS 2와 데이터 지향적 아키텍처

베어로보틱스의 핵심 기술인 플릿 매니지먼트와 연결되는 부분입니다.

- 전략: 로봇의 상태 정보(배터리, 위치, 현재 작업)를 ROS 2 메시지로 발행하고, 이를 중앙 관제 모듈(간단한 Python 스크립트)이 수집하여 시각화하는 대시보드를 만드십시오. 이는 귀하가 단일 로봇 개발을 넘어 시스템 전체를 보는 시야를 가졌음을 보여줍니다.²⁴

6. 현실적인 구현 팁 및 AI 활용 가이드

AI 및 로보틱스 비전공자로서, 이 복잡한 기술 스택을 단기간에 소화하기 위한 실질적인 조언입니다.

6.1. AI(LLM)에게 일을 시키는 법

- 나쁜 프롬프트: "Isaac Sim에서 로봇 네비게이션 코드 짜줘." (결과가 너무 방대하고 예러가 많음)
- 좋은 프롬프트: "Isaac Sim Python API를 사용해서 DynamicCuboid 객체를 (0,0,0) 좌표에 생성하고, 물리를 적용하는 스크립트를 작성해줘. ROS 2 Humble 브릿지와 연동되어야 해."
- 전략: 전체 코드를 한 번에 요청하지 말고, 기능 단위(모듈)로 쪼개서 요청하십시오. 예를 들어, "Lidar 센서를 부착하는 함수", "VLM 모델을 로드하는 클래스", "ROS 2 토픽을 발행하는 노드" 등으로 나누어 개발한 뒤 통합하는 것이 효율적입니다.

6.2. 로컬 노트북과 A6000의 역할 분담

- GPU 서버 (A6000): 무거운 연산(렌더링, 물리 시뮬레이션, AI 모델 추론, ROS 2 네비게이션 스택)을 전담합니다.
- 로컬 노트북: 가벼운 작업(VS Code를 통한 코딩, Rviz2를 통한 데이터 시각화, Omniverse Client를 통한 화면 확인)만 수행합니다.
- 팁: SSH 연결 시 tmux나 screen 같은 터미널 멀티플렉서를 사용하여, 인터넷 연결이 끊겨도 서버의 시뮬레이션이 죽지 않도록 하십시오.

6.3. 베어로보틱스 현직자 관점에서의 조언

베어로보틱스는 현재 '확장성(Scalability)'과 '안정성(Reliability)'에 집중하고 있습니다. 따라서 프로젝트의 README나 기술 문서에 다음과 같은 키워드를 포함시키는 것이 좋습니다:

- **Scalable Architecture:** 로봇이 1대일 때와 100대일 때 모두 작동 가능한 구조인가?
- **Safety Critical:** 사람을 감지했을 때 어떤 안전 메커니즘이 작동하는가? (예: VLM 판단 실패 시 즉시 정지하는 안전 레이어 구현)

- **Data-Driven:** 규칙 기반 하드코딩이 아닌, 데이터(학습)에 기반한 의사결정 구조인가?

결론: 당신의 포트폴리오가 가야 할 길

귀하가 제안한 프로젝트는 단순한 '구현'을 넘어 베어로보틱스의 미래 비전인 소프트웨어 정의 로봇틱스와 시맨틱-소셜 내비게이션을 정확히 관통하고 있습니다.

1. **Gazebo** 대신 **Isaac Sim**을 선택함으로써, 최신 로봇틱스 트렌드와 고성능 컴퓨팅(A6000) 활용 능력을 증명했습니다.
2. **VLM**과 **Diffusion Policy**를 적용함으로써, 단순 개발자가 아닌 AI 연구자로서의 깊이를 보여주었습니다.
3. 동적 환경과 플릿 시뮬레이션을 통해, 베어로보틱스의 실제 비즈니스 도메인(식당, 호텔)에 대한 깊은 이해를 나타냈습니다.

이 프로젝트는 "혼자 컴퓨터에서 구현"하는 것이지만, 그 결과물은 실제 로봇 수습 대를 운용해본 경험 이상의 기술적 통찰력을 담게 될 것입니다. 지금 바로 Isaac Sim을 설치하고, 2026년형 로봇틱스 엔지니어로서의 첫 줄을 코딩하십시오.

참고 문헌 및 자료 출처

- **Isaac Sim & Sim-to-Real:** ¹¹
- **RTX A6000 Optimization:** ³
- **Bear Robotics Technology & Strategy:** ¹
- **Advanced AI (VLM, Diffusion):** ¹⁵
- **Fleet Management & ROS 2:** ⁵

참고 자료

1. LG Makes Strategic Investment in Bear Robotics | LG Global, 1월 8, 2026에 액세스, <https://www.lg.com/global/newsroom/news/corporate/lg-makes-strategic-investment-in-bear-robotics/>
2. LG Makes Strategic Investment in Bear Robotics, 1월 8, 2026에 액세스, <https://lgcorp.com/media/release/27434>
3. a6000_optimized_guide.md.pdf
4. LG invests \$60 million in Bear Robotics as robot OS war intensifies - KED Global, 1월 8, 2026에 액세스, <https://www.kedglobal.com/robotics/newsView/ked202403120012>
5. Fleet Management - Bear API, 1월 8, 2026에 액세스, <https://cloud.api.bearrobotics.ai/v1.1/resources/FleetManagement/>
6. Servi - The Service Robot - Airpuria, 1월 8, 2026에 액세스, <https://airpuria.com/products/bear-robotics-servi-service-robot>
7. Servi Plus: Bear Robotics' Flagship AI Service Robot for Hospitality and Beyond, 1월 8, 2026에 액세스, <https://www.bearrobotics.ai/servi-plus>
8. RoboMealMate: Innovating Restaurant Service Hospitality with AI (Artificial

- Intelligence) - An-Najah National University, 1월 8, 2026에 액세스,
<https://repository.najah.edu/bitstreams/c8ef740b-c8ea-4b7e-929b-82c3ee730cd4/download>
9. Social robot navigation: a review and benchmarking of learning-based methods - PMC, 1월 8, 2026에 액세스, <https://pmc.ncbi.nlm.nih.gov/articles/PMC12739477/>
 10. Servi is agile enough to cover a range of responsibilities, reducing service effort and stress for better service, happier staff, and less turnover. - RobotLAB, 1월 8, 2026에 액세스,
<https://www.robotlab.com/hubfs/servi%20and%20plus%20Robotlab%20location.pdf>
 11. Humanoid Deploys NVIDIA Technologies to Power Advanced Humanoid Robots, 1월 8, 2026에 액세스,
<https://www.analyticsinsight.net/artificial-intelligence/humanoid-deploys-nvidia-technologies-to-power-advanced-humanoid-robots>
 12. Domain Randomization With Replicator — Getting Started With Isaac Sim, 1월 8, 2026에 액세스,
<https://docs.nvidia.com/learning/physical-ai/getting-started-with-isaac-sim/latest/synthetic-data-generation-for-perception-model-training-in-isaac-sim/03-domain-randomization-with-replicator.html>
 13. Simulation: A Game-Changer — Getting Started With Isaac Lab, 1월 8, 2026에 액세스,
<https://docs.nvidia.com/learning/physical-ai/getting-started-with-isaac-lab/latest/transferring-robot-learning-policies-from-simulation-to-reality/01-reinforcement-learning/02-simulation.html>
 14. Isaac Sim - Robotics Simulation and Synthetic Data Generation - NVIDIA Developer, 1월 8, 2026에 액세스, <https://developer.nvidia.com/isaac/sim>
 15. Visual language maps for robot navigation - Google Research, 1월 8, 2026에 액세스, <https://research.google/blog/visual-language-maps-for-robot-navigation/>
 16. VLM-Social-Nav: Socially Aware Robot Navigation Through Scoring Using Vision-Language Models - GMU CS Department, 1월 8, 2026에 액세스,
https://cs.gmu.edu/~xiao/papers/vlm_social_nav.pdf
 17. NoMaD: Goal Masking Diffusion Policies for Navigation and Exploration, 1월 8, 2026에 액세스, <https://general-navigation-models.github.io/nomad/>
 18. [2505.08712] NavDP: Learning Sim-to-Real Navigation Diffusion Policy with Privileged Information Guidance - arXiv, 1월 8, 2026에 액세스,
<https://arxiv.org/abs/2505.08712>
 19. Simulate Robotic Environments Faster with NVIDIA Isaac Sim and World Labs Marble, 1월 8, 2026에 액세스,
<https://developer.nvidia.com/blog/simulate-robotic-environments-faster-with-nvidia-isaac-sim-and-world-labs-marble/>
 20. Livestream Clients — Isaac Sim Documentation - NVIDIA, 1월 8, 2026에 액세스,
https://docs.isaacsim.omniverse.nvidia.com/4.2.0/installation/manual_livestream_clients.html
 21. Scaling Robotic Simulation with Marble | World Labs, 1월 8, 2026에 액세스,
<https://www.worldlabs.ai/case-studies/1-robotics>

22. How to create a movie capture tool in the standalone workflow? - NVIDIA Developer Forums, 1월 8, 2026에 액세스, <https://forums.developer.nvidia.com/t/how-to-create-a-movie-capture-tool-in-the-standalone-workflow/236611>
23. Movie Capture — Omniverse Extensions, 1월 8, 2026에 액세스, https://docs.omniverse.nvidia.com/extensions/latest/ext_core/ext_movie-capture.html
24. Large Language Models for Multi-Robot Systems: A Survey - arXiv, 1월 8, 2026에 액세스, <https://arxiv.org/html/2502.03814v3>
25. open-rmf/rmf_demos: Demonstrations of Open-RMF - GitHub, 1월 8, 2026에 액세스, https://github.com/open-rmf/rmf_demos
26. Servi & Servi Mini: Hospitality's Best AI Robot Waiter Assistants - Bear Robotics, 1월 8, 2026에 액세스, <https://www.bearrobotics.ai/servi>
27. Mind and Motion Aligned: A Joint Evaluation IsaacSim Benchmark for Task Planning and Low-Level Policies in Mobile Manipulation - arXiv, 1월 8, 2026에 액세스, <https://arxiv.org/html/2508.15663v1>
28. Open-RMF, 1월 8, 2026에 액세스, <https://www.open-rmf.org/>