두산로보틱스

ROKEY BOOT CAMP

교육커리큘럼 소개

(기초/입문)



ROKEY BOOT CAMP



1. 교육 개요

- 교육과정 소개
- 교육과정 구성표

교육과정 소개

AI와 로봇의 융합,
미래 산업의 엔지니어를 키웁니다.

AI 로봇 엔지니어 양성과정은 인공지능(AI), 로봇 제어(ROS2), 컴퓨터 비전, 디지털 트윈(Digital Twin) 등 첨단 기술을 융합하여
산업 현장에서 즉시 활용 가능한 실무형 로봇 엔지니어를 양성하는 교육 프로그램입니다.

본 과정은 로봇 산업의 핵심 기술 흐름을 체계적으로 학습하고, AI 기반 자동화 시스템과 협동로봇 제어 기술을 실습 중심으로 익힐 수 있도록 설계되었습니다.
이론과 프로젝트를 결합한 단계별 커리큘럼을 통해 로봇 제어, 데이터 처리, 비전 인식, 디지털 시뮬레이션 등 **AI 로보틱스 융합 역량을 전방위적으로 개발합니다.**

교육 특징

AI + Robotics 융합 학습 체계

- 인공지능 기술과 ROS2 기반 로봇 제어 기술을 통합적으로 학습
- 실제 산업 자동화 시나리오를 반영한 프로젝트 중심 실습

디지털 트윈 기반 시뮬레이션 교육

- Isaac Sim, Gazebo 등 시뮬레이터를 활용해 로봇 동작, 공정 제어, 자율주행 알고리즘을 가상환경에서 구현

현장 맞춤형 실무 중심 프로젝트

- 두산 협동로봇(DooSan Cobot) 및 센서 실습을 통한 실제 로봇 제어 경험
- 팀 기반 프로젝트로 문제 해결력 및 협업 능력 강화

산업 수요 기반 커리큘럼

- 스마트팩토리, 물류 자동화, 서비스 로봇 등 산업현장 수요를 반영한 직무 중심 교육 구성

기대효과

- **ROS2 기반 로봇 제어 실무 능력 확보**
- **AI·비전 융합 로봇 시스템 구현 역량 강화**
- **디지털 트윈 기반 로봇 시뮬레이션 경험 축적**
- **산업현장 즉시 투입 가능한 실전형 엔지니어 양성**
- **AI·로봇·스마트팩토리 분야 취업 경쟁력 강화**

교육과정 구성표

구분	단계	교과목명		교과 유형	기준 시간	차시	교육 시간(h)	교육 방법
기술 이론	[1단계] 기본학습 Basic-Cource	**1. 프로그래밍 기초**						비대면
			1-1. 파이썬 프로그래밍	<기본>	6	21	**126**	
	[2단계] 필수학습 Main course	**1. Computer Vision**						
			1-1. AI(Computer Vision) 개론	<기본>	6	17	**102**	
			1-2. AI(Computer Vision) 응용	<응용>	8	16	**128**	
		2. Robotics						
			2-1. ROS 기초학습	<기본>	6	5	**30**	
			2-2. ROS 프로그래밍 입문	<기본>	6	9	**54**	
			2-3. ROS 프로그래밍 실습	<기본>	6	8	**48**	
		3. DevOps						
			3-1. DevOps 구현	<기본>	6	6	**36**	
실무 프로젝트	[3단계] 기초실무 Intensive Course	0. 재량교과 _ (실무 프로젝트 OT)		<실무>	1	1	**1**	대면
		1. 협동로봇 프로그래밍 실습 (기초)		<실무>	8	9	**72**	
		2. 지능형로봇 프로그래밍 실습 (기초)		<실무>	8	3	**24**	
	[4단계] 통합실무 Practice Course	1. 협동로봇 프로그래밍 실습 (심화)		<실무>	8	21	**168**	
		2. 지능형로봇 프로그래밍 실습 (심화)		<실무>	8	7	**56**	
전체 교육 시간(h)							**845**	



2. 기술이론 교육

[프로그래밍 기초]

- Python 프로그래밍

[프로그래밍 입문]

- AI(Computer Vision)
- ROS-2 (Robot Operating System)
- DevOps 구현

[프로그래밍 기초]

Python 프로그래밍

학습배경

로보틱스에서 Python은 핵심적인 프로그래밍 언어이며, 본 과정의 최종 목표인 **AI 기반 로봇 시스템 개발**을 위해 필수적 역량

- **AI 및 데이터 과학의 표준 언어**
- **ROS2와의 높은 호환성**
- **높은 생산성과 확장성**

따라서 본 교과목은 교육생들이 프로그래밍의 기초를 다지고,
후속 교과인 AI와 ROS2 학습을 성공적으로 이수하며, 최종적으로 **실무 프로젝트를 수행할 수 있는 역량을 갖추게 하는 것**을 목표로 합니다.

학습목표

1. 프로그래밍의 기초 확립

Python의 기본 문법을 완벽히 이해하고, 변수, 제어문, 함수, 클래스 등을 활용하여 논리적인 코드를 작성하는 프로그래밍 사고력을 향상 합니다.

2. 문제 해결 능력 함양

스택, 큐, DFS/BFS와 같은 필수 자료구조와 알고리즘을 학습하여, 복잡한 문제에 직면했을 때 알고리즘적 해결 방안을 설계할 수 있는 능력을 배양 합니다.

3. AI/데이터 분석 역량 확보

Numpy, Pandas, Matplotlib, OpenCV 등 AI 및 데이터 처리에 필수적인 라이브러리 사용법을 익혀, 데이터를 다루고 분석하며 시각화하는 능력을 확보 합니다.

4. 실무 개발 역량 강화

파일 입출력, 예외 처리, 웹 데이터 수집(크롤링), 업무 자동화 등 실제 현업에서 마주할 수 있는 문제를 해결하는 실용적인 프로그래밍 기술을 체득 합니다.

[프로그래밍 기초]

Python 프로그래밍

커리큘럼

차시	학습 주제	주요 학습 내용
1	**파이썬 소개 및 환경설정**	파이썬 기초 이해 및 VSCode/Colab 개발 환경 설정
2	**프로그래밍 기초**	변수, 자료형, 연산자 등 프로그래밍 기본 요소 학습
3	**조건식과 제어문**	if-elif-else 조건문을 활용한 프로그램 흐름 제어
4	**리스트와 딕셔너리**	데이터 관리를 위한 리스트, 튜플, 딕셔너리 자료구조 학습
5	**반복문**	for, while 반복문을 이용한 특정 작업 반복 구현
6	**함수(1)**	코드 재사용을 위한 함수의 정의, 호출, 매개변수, 반환 값 학습
7	**함수(2)**	전역/지역 변수, 디폴트 매개변수 등 함수 심화 개념 이해
8	**자료구조와 알고리즘**	선택 정렬 등 기본 알고리즘의 원리 파악 및 구현
9	**클래스(1)**	객체 지향 프로그래밍(OOP)의 핵심인 클래스와 객체 개념 이해
10	**클래스(2)**	모듈 생성, 클래스 상속, 오버라이딩을 통한 코드 구조 확장
11	**PyQt**	PyQt 라이브러리를 활용한 간단한 GUI 프로그램 제작

차시	학습 주제	주요 학습 내용
12	**파일 처리**	외부 데이터 처리를 위한 파일 읽기/쓰기 및 입출력 학습
13	**예외 처리·문자열·람다 함수·map 함수**	예외 처리, 문자열 함수, 람다/map 등 실용적인 고급 문법 학습
14	**정규표현식**	정규 표현식을 이용한 복잡한 문자열 패턴 검색 및 처리
15	**고급함수**	대용량 데이터 효율적 처리를 위한 이터레이터, 제너레이터 학습
16	**알고리즘(1)**	대표 자료구조인 스택(Stack)과 큐(Queue)의 원리 이해 및 구현
17	**알고리즘(2)**	그래프 탐색 핵심 알고리즘인 DFS와 BFS 원리 이해 및 구현
18	**딥러닝 파이썬 패키지(1)**	Pandas, Numpy, Matplotlib 등 데이터 분석 핵심 라이브러리 활용
19	**딥러닝 파이썬 패키지(2)**	statsmodels, scikit-learn, scipy 등 고급 분석 패키지 학습
20	**딥러닝 파이썬 패키지(3)**	seaborn, OpenCV를 활용한 데이터 시각화 및 이미지 처리
21	**Python 활용 심화**	파일/엑셀 자동화, 웹 크롤링 등 실무 자동화 프로그램 제작

[프로그래밍 기초]

Python 프로그래밍

학습자원

구분	내용
강의 방식	실시간 비대면 강의(Zoom)
OS 및 개발 환경	Windows 10/11 또는 Ubuntu 24.04, Python 3.10+
실습 도구	Visual Studio Code
소통 및 Q&A	실시간 줌 채팅, Notion, Slack (기초 DevOps 협업 툴 활용)
주요 교재	파이썬 워크북(길벗)
보조 학습 자료	ROKEY 자체 제작 강의자료, 실제 예제 코드

기대효과

Python 기반 프로그래밍 역량 확보

· 로봇 제어, AI 데이터 처리, 자동화 스크립트 작성 등 다양한 실무 영역에 활용 가능합니다.

AI 및 로봇 응용 개발로의 확장 기반 마련

· ROS2, Computer Vision, AI 모델링 등 후속 과목 수행을 위한 기술적 토대를 마련합니다.

데이터 기반 문제 해결 능력 강화

· 로봇 제어, AI 데이터 처리, 자동화 스크립트 작성 등 다양한 실무 영역에 활용 가능합니다.

코딩 습관 및 협업 능력 강화

· 코드 관리(Git), 문서화, 협업 환경 경험으로 실무형 개발 문화 적응력을 향상 합니다.

다양한 기술 직무로의 진출 가능성 확대

· AI 개발자, 로봇SW엔지니어, 데이터분석가, 스마트팩토리 자동화 엔지니어 등 다양한 기술 기반 직무로 진출할 수 있는 기초 역량을 확보 합니다.

[프로그래밍 입문]

AI(Computer Vision)

학습배경

로봇이 사람처럼 '보고, 판단하고, 행동하는' 시대에 들어서면서 **Computer Vision**은 로봇지능의 핵심 기술로 자리 잡고 있습니다.
산업 현장에서는 협동로봇이 AI 카메라와 함께 **부품 인식·위치 추적·품질 검사** 등을 수행하며, AI 비전 기술은 이러한 자동화 시스템의 정확성과 효율성을 결정짓는 요소입니다.

본 교과목은 Python 언어를 기반으로 **AI 비전의 기초 이론과 프로그래밍 실습을 병행**하여,
AI 로봇 제어, 자율주행, 디지털 트윈 등 고급 프로젝트에 앞서 필요한 **AI·영상 처리·데이터 분석의 기초 체계**를 완성하도록 설계되었습니다.

학습목표

1. AI 비전의 기본 개념과 원리 이해

- 이미지 인식(Image Recognition), 객체 탐지(Object Detection), 분류(Classification) 등 핵심 개념을 이해합니다.
- AI 비전의 구조를 학습과 AI가 시각 정보를 해석하는 과정(전처리 → 특징 추출 → 추론)을 체계적으로 학습합니다.

2. Python 기반 Computer Vision 프로그래밍 기초 습득

- OpenCV, Numpy, Matplotlib 등을 활용해 이미지 처리 실습 수행 합니다.
- 필터링, 엣지 검출, 색상 공간 변환 등 기본 영상 처리 알고리즘을 이해 및 구현 합니다.

3. AI 모델을 활용한 영상 인식 실습

- 사전 학습된 딥러닝 모델(YOLO, CNN 등)을 적용하여 실시간 객체 인식 구현 합니다.
- 로봇 시각 시스템의 기초 구성요소(카메라 입력 → 인식 → 판단 → 제어) 학습 합니다.

4. 로봇 비전 시스템의 응용 이해

- 협동로봇의 시각 기반 작업 시나리오(물체 인식, 위치 추정, 작업 명령 전달) 체험 합니다.
- 향후 실습 프로젝트(ROS2, 로봇 제어, 디지털 트윈 등)로의 연결 기반 마련 합니다.

[프로그래밍 입문]

AI(Computer Vision)

커리큘럼

A. AI(Computer Vision) 개론

차시	학습 주제	주요 학습 내용
1	**딥러닝에 꼭 필요한 파이썬**	Colab 활용, Pandas·Numpy, 함수, Keras 비교 / Numpy 연산, 이미지 픽셀, Matplotlib 시각화
2	**Pytorch 기본 기능**	Tensor 개념, Numpy 비교, GPU 활용
3	**인공지능 개론**	퍼셉트론, 활성화 함수, 시각 애니메이션
4	**인공지능 학습**	경사하강법, 시각화, 로봇 위치 오차
5	**비용 함수 정의**	MSE, BCE, CE, 불량품 분류 예제
6	**선형 회귀**	단순회귀, 다항회귀, 로봇 부품 예측
7	**이진 분류**	BCE Loss, Precision·Recall, 불량품 탐지
8	**다중 분류 1**	Softmax, CrossEntropy, 제품 분류
9	**다중 분류 2**	MNIST, FashionMNIST, 데이터 증강

차시	학습 주제	주요 학습 내용
10	**CNN 1**	CNN 구조, 커널·패딩, CIFAR10·부품 이미지
11	**튜닝 기법**	최적화(Adam 등), Dropout, BatchNorm
12	**사전학습 모델 1**	AlexNet, GoogleNet, 전이학습
13	**사전학습 모델 2**	VGG, ResNet, 산업 데이터 적용
14	**사용자 정의 데이터**	데이터 준비, 전이학습, 부품 데이터
15	**CNN 모델 빌드**	Residual Block, ResNet18 구현
16	**자연어처리**	RNN, LSTM, 시계열 예측
17	**Transformer & ViT**	Transformer, ViT, 불량품 검사

[프로그래밍 입문]

AI(Computer Vision)

커리큘럼 B. AI(Computer Vision) 응용

차시	학습 주제	주요 학습 내용
1	**OpenCV 영상정규화 및 Object Detecion 입문**	디지털 영상 기초, 색공간(RGB/HSV), 화소점 연산, 히스토그램 정규화, Bounding Box 기본 개념
2	**OpenCV 기하학적 변환**	필터링(평균/가우시안/샤프닝), Affine/Perspective 변환, 회전·확대·전단, Segmentation 전처리
3	**OpenCV 동영상 처리, Object Tracking**	에지 검출, 직선·원 검출, 객체/외곽선 검출, Optical Flow 활용, 간단한 Tracking 실습
4	**추적 알고리즘 실습**	Meanshift·Camshift 추적, Lucas-Kanade, Dense Flow, Object Tracking 평가
5	**CNN 기반 분류 앱**	MobileNet 구조, Custom Dataset 적용, Transfer Learning
6	**신경망 스타일 전이**	Neural Style Transfer, Content/Style Loss, PyTorch 구현
7	**이미지 생성형 AI**	GAN, DCGAN, VAE 구조, PyTorch VAE
8	**이미지 캡셔닝**	CNN+LSTM 구조, 텍스트 인코딩, 학습 및 캡션 생성

차시	학습 주제	주요 학습 내용
9	**3D 렌더링**	NeRF 개념, TinyNeRF 구현, Volume Rendering, Gaussian Splatting
10	**설명 가능한 AI (XAI)**	XCAM, Grad-CAM, Saliency Map, Integrated Gradients
11	**의료영상 분석**	DICOM 기초, X-ray 전처리, ResNet18 분류, CAM 기반 의료영상 해석
12	**객체 검출 1**	Faster RCNN, Mask RCNN, 작업장 객체
13	**객체 검출 2**	YOLO, SSD, 안전모 탐지
14	**Object Detection & Segmentation 1**	YOLOv5/YOLOv8 구조, Annotation 실습(Roboflow, Label Studio), Bounding Box 학습
15	**Object Detection & Segmentation 2**	성능지표(Precision, Recall, mAP), Segmentation 실습, 모델 개선 및 튜닝
16	**강화학습 개요**	RL 기본개념(Monte Carlo, Q-learning), 간단한 DQN 실습(CartPole)

[프로그래밍 입문]

AI(Computer Vision)

학습자원

구분	내용
강의 방식	실시간 비대면 강의(Zoom)
OS 및 개발 환경	Windows 10/11 또는 Ubuntu 24.04, Python 3.10+, Anaconda
실습 도구	Visual Studio Code, Google Colab, Jupyter Notebook
소통 및 Q&A	실시간 줌 채팅, Notion, Slack
주요 교재	『차근차근 실습하며 배우는 파이토치 딥러닝 프로그래밍』(위키북스) 『실전! 파이토치 딥러닝 프로젝트』(위키북스)
보조 학습자료	강의 노트, GitHub 예제 코드, 비전 알고리즘 실습 가이드

기대효과

AI 비전 핵심 원리 및 프로그래밍 능력 확보

· 영상 데이터 전처리, 객체 탐지, 이미지 분석 등 핵심 알고리즘 실무 적용 가능 합니다.

로봇 시각 시스템 구현의 기초 역량 확보

· 협동로봇, 자율주행, 검사 시스템 등 산업용 비전 분야로 확장 가능 합니다.

AI 및 데이터 기반 문제해결력 향상

· 데이터를 분석하고 시각적으로 표현하는 능력 강화로 AI/로봇 융합 개발에 활용 합니다.

딥러닝 및 AI 고급 과목으로의 진입 기반 마련

· 추후 “AI 비전 응용 프로젝트”, “ROS2 연동 실습”, “디지털 트윈 구현” 등 실무 프로젝트로의 확장 가능성 확보 합니다.

취업 직무 연계성 강화

· AI 비전 엔지니어, 로봇 SW개발자, 데이터 비전 분석가, 스마트팩토리 기술 엔지니어 등 관련 직무로 직접 연계 가능한 기술 기반을 마련 합니다.

[프로그래밍 입문]

ROS-2 (Robot Operating System)

학습배경

ROS(Robot Operating System)는 전 세계 로봇 개발자들이 공통으로 사용하는 **로봇 제어용 표준 프레임워크**로,
산업용 협동로봇, 자율주행로봇, 서비스로봇 등 거의 모든 로봇 응용 분야에서 활용되고 있습니다.
특히 ROS2는 기존 ROS1의 한계를 극복하고,
실시간성, 멀티로봇 통신, 보안성, 확장성이 강화된 최신 버전으로 스마트팩토리, 물류 자동화, 자율주행, 디지털 트윈 등 산업현장에서 필수 기술로 자리 잡았습니다.

본 교과목은 **AI 로봇 엔지니어의 필수 기반 기술**로서,
로봇의 동작 원리와 데이터 흐름을 체계적으로 학습하고 실제 ROS2 기반 프로젝트 수행을 위한 토대를 다지는 것을 목표로 합니다.

학습목표

1. ROS2 기본 개념 및 구조 이해

- 로봇 운영체제(ROS2)의 핵심 구성 요소(노드, 토픽, 서비스, 액션)를 이해하고, 분산 제어 구조와 통신 메커니즘의 작동 원리를 학습합니다.

2. ROS2 개발환경 구축 및 실습 역량 습득

- Ubuntu 기반 ROS2 Humble 설치 및 개발환경 구성
- 워크스페이스 생성, 패키지 빌드, 노드 실행 등 기본 프로그래밍 실습 수행

3. 센서·로봇 간 데이터 흐름 및 메시지 통신 이해

- 토픽 기반의 퍼블리셔/서브스크라이버 구현을 통해 로봇의 데이터 교환 구조 학습
- 실제 센서 데이터(예: 카메라, 라이다 등)를 활용한 ROS2 메시지 송수신 실습

4. 로봇 제어 및 시뮬레이션 실습 기반 마련

- Gazebo, RViz2 등 시각화·시뮬레이션 도구를 활용하여 로봇 동작 원리와 시뮬레이션 구현
- 이후 실습(로봇 제어, 디지털 트윈, AI 비전 연동 등)으로 확장 가능한 실무 기초 역량 확보

5. 로봇 소프트웨어 개발 프로세스 이해

- ROS2 기반 프로젝트 구조 관리, 빌드 시스템(colcon), 노드 간 의존성 설계 등 실무 개발 흐름을 이해하고 프로젝트 수행 능력 강화

[프로그래밍 입문]

ROS-2 (Robot Operating System)

커리큘럼

A. ROS-2 (Robot Operating System) 기초 학습

차시	학습 주제	주요 학습 내용
1	**컴퓨터 구조, 리눅스와 운영체계**	컴퓨터 구조, 리눅스 운영체제 및 CLI 명령어 기초 학습
2	**리눅스 CLI 실습, 네트워크와 통신**	리눅스 CLI 심화 및 네트워크 통신 기본 원리 이해
3	**로봇센서 소개, ROS 소개 및 활용**	로봇 센서의 종류와 역할, ROS2 기본 개념 소개
4	**ROS2 설치, 패키지 설명**	ROS2 설치 및 talker-listener 예제를 통한 패키지 실행 실습
5	**ROS2를 활용한 turtlesim 실습 및 gazebo 소개**	turtlesim 시뮬레이터를 활용한 ROS2 핵심 통신 방식(Topic, Service 등) 종합 실습

[프로그래밍 입문]

ROS-2 (Robot Operating System)

커리큘럼 **B. ROS-2** (Robot Operating System) **프로그래밍 입문**

차시	학습 주제	주요 학습 내용
1	**ROS2 프로그래밍 기초**	ROS2 프로그래밍의 기본 규칙을 이해하고 Python으로 첫 ROS2 패키지 생성
2	**인터페이스 패키지**	ROS2 통신의 기반이 되는 인터페이스 패키지 정의 및 실습
3	**토픽, 서비스 패키지**	단방향/양방향 통신 방식인 Topic과 Service 패키지 생성 및 실습
4	**rclpy 이해**	Python으로 ROS2 노드를 제어하는 rclpy 라이브러리와 launch 파일 핵심 기능 이해
5	**액션 패키지**	비동기 장시간 작업을 위한 Action 패키지와 노드 설정값 관리(Parameters) 방법 학습
6	**인터페이스 프로그래밍(응용 _ 1)**	계산기 예제를 통한 인터페이스 프로그래밍 응용 실습 (1)
7	**인터페이스 프로그래밍(응용 _ 2)**	행맨 게임 예제를 통한 인터페이스 프로그래밍 응용 실습 (2)
8	**ROS2 응용**	ROS2 데이터 기록/재생 도구인 bag 파일의 이해 및 사용법 학습
9	**ROS2 응용 핵심정리**	ROS2의 핵심 개념들을 복습하고 응용 프로그래밍 실습을 통한 최종 정리

[프로그래밍 입문]

ROS-2 (Robot Operating System)

커리큘럼

C. ROS-2 (Robot Operating System) 프로그래밍 실습

차시	학습 주제	주요 학습 내용
1	**ROS2 CLI 및 심화 기능1**	ROS2 CLI의 다양한 명령어와 ros2env.zip 등 심화 기능 사용법 학습.
2	**ROS2 심화 기능 1**	통신 효율을 높이는 IPC 및 DDS의 QoS(서비스 품질) 개념 이해와 활용.
3	**ROS2 심화 기능 2**	노드의 재사용성을 높이는 Component 개념과 RQT 플러그인 사용법 실습.
4	**ROS2 심화 기능 2 및 시뮬레이션 개발**	노드 관리(Lifecycle), 보안, 로봇 모델링(URDF/TF2) 등 고급 기능 이해.
5	**시뮬레이션 개발 / 두산 로봇과 ROS2**	Gazebo 시뮬레이션 환경 구축 및 두산 로봇을 활용한 SLAM, Nav2 연동 실습.
6	**ROS2와 OpenCV**	ROS2와 OpenCV를 연동한 차선 인식(Lane detect) 시스템 구현 및 통합.
7	**ROS2와 3차원 영상 및 시각화**	Point Cloud 데이터를 RVIZ, Open3D로 시각화하는 3차원 영상 처리 학습.
8	**ROS2 프로그래밍 실습 핵심정리**	ROS2 프로그래밍 실습 전 과정에 대한 핵심 내용 복습 및 최종 정리.

[프로그래밍 입문]

ROS-2 (Robot Operating System)

학습자원

구분	내용
강의 방식	실시간 비대면 강의(Zoom)
OS 및 개발 환경	Ubuntu 24.04, ROS2 Jazzy, Python 3.10+
실습 도구	Visual Studio Code, turtlesim, Gazebo, RViz2, RQt
소통 및 Q&A	실시간 줌 채팅, Notion, Slack (기초 DevOps 협업 툴 활용)
주요 교재	ROKEY 자체 제작 강의자료
보조 학습 자료	ROS2 공식 문서(docs.ros.org), GitHub 실습 예제 코드

기대효과

ROS2 핵심 구조 이해 및 프로그래밍 역량 확보

· 노드, 토픽, 서비스, 액션, 파라미터 등 ROS2 통신 구조를 이해하고 Python(rclpy) 기반 패키지를 직접 개발할 수 있습니다.

리눅스 기반 로봇 개발환경 활용 능력 강화

· Ubuntu CLI, colcon 빌드, launch 파일, bag 기록/재생 등 실무 로봇 개발환경을 능숙하게 다룰 수 있습니다.

시뮬레이션·시각화 기반 개발 역량 확보

· Gazebo, RViz2, RQt를 활용하여 URDF/TF2 로봇 모델링, SLAM·Nav2 연동 등 시뮬레이션 기반 개발을 수행할 수 있습니다.

비전·센서 데이터 연동 기초 역량 확보

· ROS2와 OpenCV, Point Cloud 연동 경험을 통해 AI 비전 및 3차원 영상 처리 응용으로 확장할 수 있습니다.

실무 프로젝트로의 연계 기반 마련

· AI 기반 협동로봇, 디지털 트윈 등 후속 실무 프로젝트 수행을 위한 기술적 토대를 마련합니다.

[프로그래밍 입문]

DevOps 구현

학습배경

지능형 로봇 개발은 단순히 알고리즘을 구현하는 것을 넘어, **데이터를 효율적으로 관리**하고, **소스 코드를 체계적으로 협업**하며, 개발된 애플리케이션을 **안정적으로 배포 및 운영**하는 전 과정을 포함합니다.
DevOps는 이러한 개발과 운영의 전반적인 생산성을 높이는 핵심적인 방법론입니다.

- **데이터베이스 (SQL)**: 로봇이 수집하는 센서 데이터, 학습 데이터, 로그 등을 저장하고 관리하기 위한 필수 기술입니다.
- **버전 관리 (Git)**: 여러 개발자가 로봇 소프트웨어의 코드를 동시에 수정하고, 변경 이력을 추적하며 협업하기 위한 표준 도구입니다.
- **컨테이너화 (Docker)**: 개발 환경과 실제 로봇의 운영 환경을 일치시켜 "제 PC에선 됐는데..."와 같은 문제를 원천적으로 차단하고, 애플리케이션을 신속하고 일관성 있게 배포하기 위한 필수 기술입니다.

본 교과목은 교육생들이 로봇 개발의 전 과정에서 마주할 수 있는 데이터, 협업, 배포 문제를 해결할 수 있는 **기본적인 DevOps 역량을 갖추게 하는 것**을 목표로 합니다.

학습목표

1. 데이터 관리 능력

관계형 데이터베이스의 기본을 이해하고, SQL을 사용하여 데이터를 생성(Create), 조회(Read), 수정(Update), 삭제(Delete)할 수 있는 능력을 향상 합니다.

2. 소스코드 협업 능력

Git을 사용하여 개인 프로젝트의 소스 코드를 관리하고, GitHub를 통해 다른 팀원과 체계적으로 협업하는 방법을 체득 합니다

3. 애플리케이션 배포/운영 능력

Docker의 컨테이너 개념을 이해하고, 개발한 애플리케이션을 Docker 이미지로 만들어 어떤 환경에서든 동일하게 실행시키는 능력을 배양 합니다.

[프로그래밍 입문]

DevOps 구현

커리큘럼

차시	학습 주제	주요 학습 내용
1	**SQL 기초 및 데이터 조작**	SQL 기초 문법을 익혀 데이터를 생성, 조회, 필터링하는 방법을 학습
2	**SQL 심화 및 Python 연동**	테이블 조인(JOIN), 그룹화 등 고급 SQL을 다루고 Python으로 DB를 제어
3	**Git 로컬 버전 관리**	Git의 핵심 워크플로우를 이해하고, 명령어를 통해 로컬 저장소의 버전을 관리
4	**GitHub를 활용한 협업**	GitHub 원격 저장소를 이용해 협업하고, 브랜치(Branch)를 활용하며 충돌을 해결
5	**Docker 기초 및 컨테이너 활용**	Docker 컨테이너의 개념을 이해하고, 이미지를 활용해 애플리케이션을 실행
6	**Docker 이미지 생성 및 배포**	Dockerfile로 직접 이미지를 만들고, Docker Compose로 다중 컨테이너를 관리

[프로그래밍 입문]

DevOps 구현

학습자원

구분	내용
강의 방식	실시간 비대면 강의(Zoom)
OS 및 개발 환경	Windows 10/11 또는 Ubuntu 24.04
실습 도구	· Database: postgreSQL · Version Control: Git SCM, Visual Studio Code, GitHub · Conteiner: Docker Desktop 또는 Docker
소통 및 Q&A	실시간 줌 채팅, Notion, Slack (기초 DevOps 협업 툴 활용)
주요 교재	ROKEY 자체 제작 강의자료
보조 학습 자료	실습 가이드, 실습 예제 코드

기대효과

데이터 기반 개발 역량 확보

· 로봇 운영에 필요한 데이터를 직접 설계하고, SQL을 통해 관리할 수 있는 기초 데이터 엔지니어링 역량을 갖추게 됩니다.

체계적인 협업 능력 체득

· Git과 GitHub를 활용하여 팀 프로젝트의 소스 코드를 효율적으로 관리하고, 충돌을 해결하며 체계적으로 협업할 수 있게 됩니다.

안정적인 배포/운영 능력 배양

· Docker를 통해 개발 환경에 구애받지 않고 애플리케이션을 안정적으로 패키징하고 배포할 수 있는 능력을 갖추게 됩니다.

최종 프로젝트 완성도 향상

· 단순 코딩을 넘어, 데이터 관리, 버전 관리, 배포까지 고려한 완성도 높은 결과물을 만들 수 있는 실무 역량을 마련합니다.

ROKEY BOOT CAMP



3. 실무 프로젝트 교육

[주제-1] AI기반 협동로봇 프로젝트

[주제-2] 디지털트윈 프로젝트

[주제-1]

AI(Computer Vision)기반 협동 로봇 작업 어시스턴트 구현 프로젝트

Keyword : AI기반 협동로봇

Vision AI, 음성인식, 대형언어모델(LLM)을 통합하여 협동로봇이 사람의 지시와 주변 환경을 스스로 인식하고 대응할 수 있는 지능형 로봇 어시스턴트 시스템을 설계하고 구현합니다.

학습배경

산업 현장에서는 단순 반복 작업을 자동화하는 수준을 넘어, **사람의 언어와 시각적 정보를 이해해 함께 협업하는 로봇 시스템**이 빠르게 확산되고 있습니다.
특히 Vision AI, 음성인식, LLM 등 AI 기술의 융합은 로봇이 사람의 지시를 이해하고 자율적으로 판단·동작할 수 있는 기반을 제공합니다.

본 프로젝트는 이러한 차세대 로보틱스 트렌드를 반영하여, **AI·ROS2·로봇 하드웨어 제어 기술을 통합적으로 다루는 실무형 엔지니어**를 양성하는 것을 목표로 합니다.

학습목표

1. ROS2 기반 로봇 제어 기술 습득

센서 데이터 수집, 로봇 구동, 제어 노드 간 통신 구조를 이해하고 구현

2. AI 비전 모델 기반 객체 인식 기술 실습

YOLO 딥러닝 모델을 활용해 카메라 영상에서 물체를 탐지하고 위치를 인식

3. 음성인식·언어모델 기반 인터랙션 구현

음성 입력을 통해 명령을 인식하고(STT), LangChain 기반 LLM을 통해 로봇 행동을 결정하는 음성 제어 시스템 개발

4. 하드웨어 연동 및 통합 테스트 수행

Intel RealSense 카메라, 협동로봇, 그리퍼 등 물리 장비를 통합하여 실제 산업 환경 수준의 자동화 시스템 구현

5. 팀 기반 프로젝트 수행 및 협업 역량 강화

4~5인 팀 단위로 역할을 분담해 개발, 통합, 테스트, 발표까지 전 과정을 수행

[주제-1]

AI(Computer Vision)기반 협동 로봇 작업 어시스턴트 구현 프로젝트

Keyword : AI기반 협동로봇

Vision AI, 음성인식, 대형언어모델(LLM)을 통합하여 협동로봇이 사람의 지시와 주변 환경을 스스로 인식하고 대응할 수 있는 지능형 로봇 어시스턴트 시스템을 설계하고 구현합니다.

커리큘럼

차시	학습 주제	주요 학습 내용	실습 / 프로젝트
1	**개발환경 구축 및 센서 데이터 처리 기초**	· 패키지 구성 · RGB-D 카메라 데이터 수집 및 시각화	· 필수 라이브러리 설치 · RQt를 활용한 Realsense 카메라 토픽 확인 · Rviz2를 활용한 point cloud 출력 확인
2	**그리퍼 조작 및 카메라 calibration**	· RG2 그리퍼 Modbus 통신 · 카메라 calibration	· Python 코드를 활용한 Modbus 그리퍼 조작 · Realsense 카메라 calibration 실습
3	**Vision AI 기반 객체 인식**	Ultralytics 라이브러리의 YOLO 모델 활용	YOLO 모델 inference 튜토리얼
4	**Custom 데이터셋 구축**	· Roboflow 데이터셋 구축 방법 · YOLO 모델 학습 방법	· Roboflow를 활용한 custom 데이터셋 구축 · YOLO 모델 학습 및 테스트
5	**Vision AI를 활용한 Pick & Place text**	· Pick & Place 개요 · CLI 명령문 기반 Pick & Place text 실행 방법	학습된 YOLO 기반 Pick & Place text 실습
6	**로봇 호출한 후 음성 명령을 텍스트로 변환하기**	Wake up word와 STT(Speech To Text) 개요	Wake up word와 STT tutorial 실습
7	**텍스트를 바탕으로 한 키워드 추출**	Keyword extraction 개요	Keyword extraction tutorial 실습
8	**Vision과 Voice를 융합한 Pick & Place**	Pick & Place voice 실행 방법	Pick & Place voice tutorial 실습 진행
9	**AI 협동 로봇 통합 시스템 최종 테스트**	전체 프로세스 통합 및 오류 검증	· 최종 시연 테스트 · 발표 보고서 최종 정리
10	**결과 발표 및 보고서 작성**	팀별 프로젝트 발표 및 평가	시연 영상 및 코드 제출

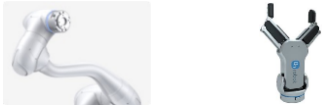
[주제-1]

AI(Computer Vision)기반 협동 로봇 작업 어시스턴트 구현 프로젝트

Keyword : AI기반 협동로봇

Vision AI, 음성인식, 대형언어모델(LLM)을 통합하여 협동로봇이 사람의 지시와 주변 환경을 스스로 인식하고 대응할 수 있는 지능형 로봇 어시스턴트 시스템을 설계하고 구현합니다.

학습자원

구분	이미지	세부 내용
로봇 하드웨어		두산 협동로봇 (M0609), Modbus 기반 그리퍼
센서/비전 장비		Intel RealSense RGB-D 카메라
AI/비전 소프트웨어	OpenCV YOLO	OpenCV, YOLO
로봇 프레임워크 /소프트웨어	RViz	ROS2 Jazzy, RQt, RViz2
개발 환경	ubuntu	Ubuntu 24.04, Python
협업 도구	git	GitHub, Slack 협업 툴

[주제-1]

AI(Computer Vision)기반 협동 로봇 작업 어시스턴트 구현 프로젝트

Keyword : AI기반 협동로봇

Vision AI, 음성인식, 대형언어모델(LLM)을 통합하여 협동로봇이 사람의 지시와 주변 환경을 스스로 인식하고 대응할 수 있는 지능형 로봇 어시스턴트 시스템을 설계하고 구현합니다.

기대효과

ROS2 기반 로봇 제어 실무 능력 확보

· 실제 산업용 협동로봇의 ROS2 연동 및 동작 프로그래밍 역량 강화

AI·비전·언어 융합 기술 실습 경험

· Vision AI와 LLM 기반 로봇 제어 기술을 동시에 다루는 통합형 프로젝트 경험 축적

산업현장 대응형 포트폴리오 확보

· 코드, 영상, 보고서 등 실질적 결과물 중심의 포트폴리오 구축

로봇·AI 융합 직무로의 진출 기반 마련

· 로보틱스 소프트웨어 엔지니어, AI 제어 알고리즘 개발자, AI 로보틱스 연구원 등 다방면 진출 가능

[주제-2]

디지털 트윈 기반 로봇 자동화 시뮬레이션 시스템 구현 프로젝트

Keyword : 디지털트윈

Isaac Sim과 ROS2를 활용하여 가상의 로봇 및 센서 시스템을 구축하고,
자율주행·로봇암 제어·자동화 공정 시뮬레이션을 구현하는 실무형 디지털 트윈 프로젝트입니다.

학습배경

디지털 트윈(Digital Twin)은 현실 시스템을 가상 공간에 그대로 구현하여 **시뮬레이션과 예측을 통해 효율적 의사결정**을 지원하는 핵심 기술입니다.
스마트공장, 자율주행, 로봇공학 등에서 실제 환경 구축 전 가상 시운전(Virtual Commissioning)을 통해 비용 절감과 리스크 최소화를 가능하게 합니다.

본 프로젝트는 NVIDIA Isaac Sim과 ROS2를 결합하여, 로봇 모델링, 센서 시뮬레이션, 자율주행 및 매니퓰레이터 제어, 시스템 통합 및 테스트를 직접 수행함으로써 **현장 중심형 로봇 엔지니어의 실무 역량을 강화**하는 것을 목표로 합니다.

학습목표

1. 디지털 트윈 개념 및 산업 활용 이해

- 디지털 트윈에 대한 개념 및 용어 이해
- 스마트공장, 물류, 제조 등 산업 전반에서의 디지털 트윈 응용 사례 이해
- 실제 생산 시스템을 가상 환경에서 시뮬레이션하는 구조 학습

2. Isaac Sim 및 ROS2 기반 로봇 시뮬레이션 기술 습득

- 가상 환경에서 로봇 모델을 불러오고, 센서 및 주행 기능을 구현
- ROS2를 통한 로봇 제어 및 시뮬레이션 연동 기술 학습

3. 자율주행 및 매니퓰레이터 제어 기술 실습

- 모바일 로봇의 SLAM 및 경로 계획 실습
- 로봇암(Pick & Place) 공정 제어 로직 설계 및 검증

4. 통합 자동화 시나리오 구축 능력 향상

- 가상 환경 내 다중 로봇 간 상호 작용 및 공정 자동화 구현
- 모듈 간 통신·시각화·데이터 로깅 기능 통합

5. 협업 기반 실무 프로젝트 경험 축적

- 팀 단위(4~5명)로 역할 분담을 통해 설계–개발–테스트 전 과정을 수행

[주제-2]

디지털 트윈 기반 로봇 자동화 시뮬레이션 시스템 구현 프로젝트

Keyword : 디지털트윈

Isaac Sim과 ROS2를 활용하여 가상의 로봇 및 센서 시스템을 구축하고,
자율주행·로봇암 제어·자동화 공정 시뮬레이션을 구현하는 실무형 디지털 트윈 프로젝트입니다.

커리큘럼

차시	학습 주제	주요 학습 내용	실습 / 프로젝트
1	**Digital Twin 및 Isaac Sim 개요**	디지털 트윈 개념 이해, Isaac Sim 개발환경 설정	Isaac Sim 설치 및 기본 조작 실습
2	**로봇 모델 불러오기 및 센서 시뮬레이션**	URDF 및 USD 이해, 로봇 모델링, RGB-D 센서 연동	Isaac Sim 와 ROS2를 이용한 로봇 및 센서 시각화
3	**시뮬레이션 기반 로봇 제어**	로봇암 제어, 모션 계획, 객체 조작 실습	Pick & Place 시뮬레이션 구현
4	**자율주행 시뮬레이션**	모바일 로봇 자율주행, SLAM, 경로 계획	가상 맵 기반 자율주행 구현
5	**ROS2 연동 및 데이터 통신**	Isaac Sim과 ROS2 간 메시지 교환 및 제어 테스트	ROS2-Topic 통신 테스트
6	**가상환경 자동화 시나리오 설계**	생산 공정 흐름 설계 및 로직 구현	자동화 공정 알고리즘 설계
7	**매니퓰레이터 제어 로직 개발**	매니퓰레이터의 모션 제어 및 충돌 감지	로봇암 경로 제어 실습
8	**자율주행 및 인지 시스템 통합**	이동 로봇과 로봇암 간 협업 시나리오 구현	복합 자동화 시뮬레이션
9	**시스템 통합 테스트**	전체 프로세스 통합 및 오류 검증	데이터 로깅 및 성능 분석
10	**프로젝트 발표 및 평가**	최종 시연, 결과 보고서 작성	팀별 결과 발표 및 피드백

[주제-2]

디지털 트윈 기반 로봇 자동화 시뮬레이션 시스템 구현 프로젝트

Keyword : 디지털트윈

Isaac Sim과 ROS2를 활용하여 가상의 로봇 및 센서 시스템을 구축하고,
자율주행·로봇암 제어·자동화 공정 시뮬레이션을 구현하는 실무형 디지털 트윈 프로젝트입니다.

학습자원

구분	이미지	세부 내용
소프트웨어 환경	NVIDIA ISAAC, ROS2 Jazzy, ubuntu, Python	Isaac Sim, ROS2 Jazzy, Ubuntu 24.04, Python
로봇 시뮬레이션 구성요소		URDF 모델, USD 모델, 센서 데이터 모듈
하드웨어 환경		고성능 노트북 (GPU 탑재)
협업 및 버전관리	git, Slack	GitHub, Slack 협업 툴
시각화 도구	GAZEBO, RViz2	RViz2, Gazebo, Isaac Sim Visualizer

[주제-2]

디지털 트윈 기반 로봇 자동화 시뮬레이션 시스템 구현 프로젝트

Keyword : 디지털트윈

Isaac Sim과 ROS2를 활용하여 가상의 로봇 및 센서 시스템을 구축하고,
자율주행·로봇암 제어·자동화 공정 시뮬레이션을 구현하는 실무형 디지털 트윈 프로젝트입니다.

기대효과

디지털 트윈 실무 역량 확보

· IsaacSim 및 ROS2 기반의 가상 시운전, 공정 자동화 시뮬레이션 능력 습득

로봇 시스템 통합 이해 및 구현 능력 강화

· 센서·로봇·제어 로직 통합 경험을 통한 실제 산업 현장 대응력 강화

자율주행 및 로봇 제어 기술 숙련

· SLAM, 객체 인식, 경로 계획 등 로봇AI 핵심 기술 습득

스마트팩토리 및 로봇 엔지니어 직무 진출 기반 확보

· 로봇 SW개발자, ROS엔지니어, 디지털 트윈 엔지니어, 자동화 솔루션 개발자 등 진출 가능

팀 협업 및 프로젝트 수행 경험 축적

· 설계–개발–통합–평가 전 과정을 수행하며 실무 프로젝트 관리 역량 향상

로봇이 세상을 바꾼다면,
그 로봇을 바꾸는 건 바로 여러분이 될 것 입니다.

두산 ROKEY BOOT CAMP!
여러분의 **미래를 설계하는 첫걸음**이 됩니다.