

거대 언어모델(LLM) 기반 로봇 제어 자율 주행 시스템 개발



송재원/ 산업공학과 20

송형섭/ 산업공학과 20

AI-ROBOTIC





CONTENT

01

1

프로젝트 개요

2

활용 모델 소개

3

시연 영상

4

영상 설명

5

결론 및 기대효과



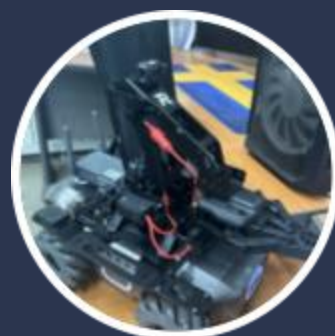
프로젝트 개요

02



- NVIDIA 초소형 컴퓨터
- GPU, CPU 높은 계산 성능
- 로봇 위 탑재 가능

Jetson orin nano



- DJI 로봇
- python 연동 가능
- 섬세한 로봇 제어 가능

Robomaster sdk



- VLM / LLM
(Vision/Large Language Model)
- 이미지/텍스트
결합처리, 텍스트 기반
추론

Open/ Local LLM

**로봇이 실제 주행환경에서 상황을 자율적으로 인식하고
움직임을 스스로 계획해 임무를 수행하도록 설계**



STEP 1

실시간 상황인지(vlm)

- Jetson Orin Nano와 로봇 카메라 연동
- 로봇 화면 기반 실시간 텍스트 상황 인식

STEP 2

실시간 객체 탐지(yolo)

- 실시간 2D 물체 좌표 추적
- Depth 카메라 연동 후 3D 물체 좌표 파악

STEP 3

자동 로봇 플래닝(chat gpt)

- 사용자 명령 인지 후 스스로 행동 계획
- 정밀한 로봇제어
- 강화학습이 필요 없는 zero-shot

🌐 활용 모델 소개

04

FLORENCE-2

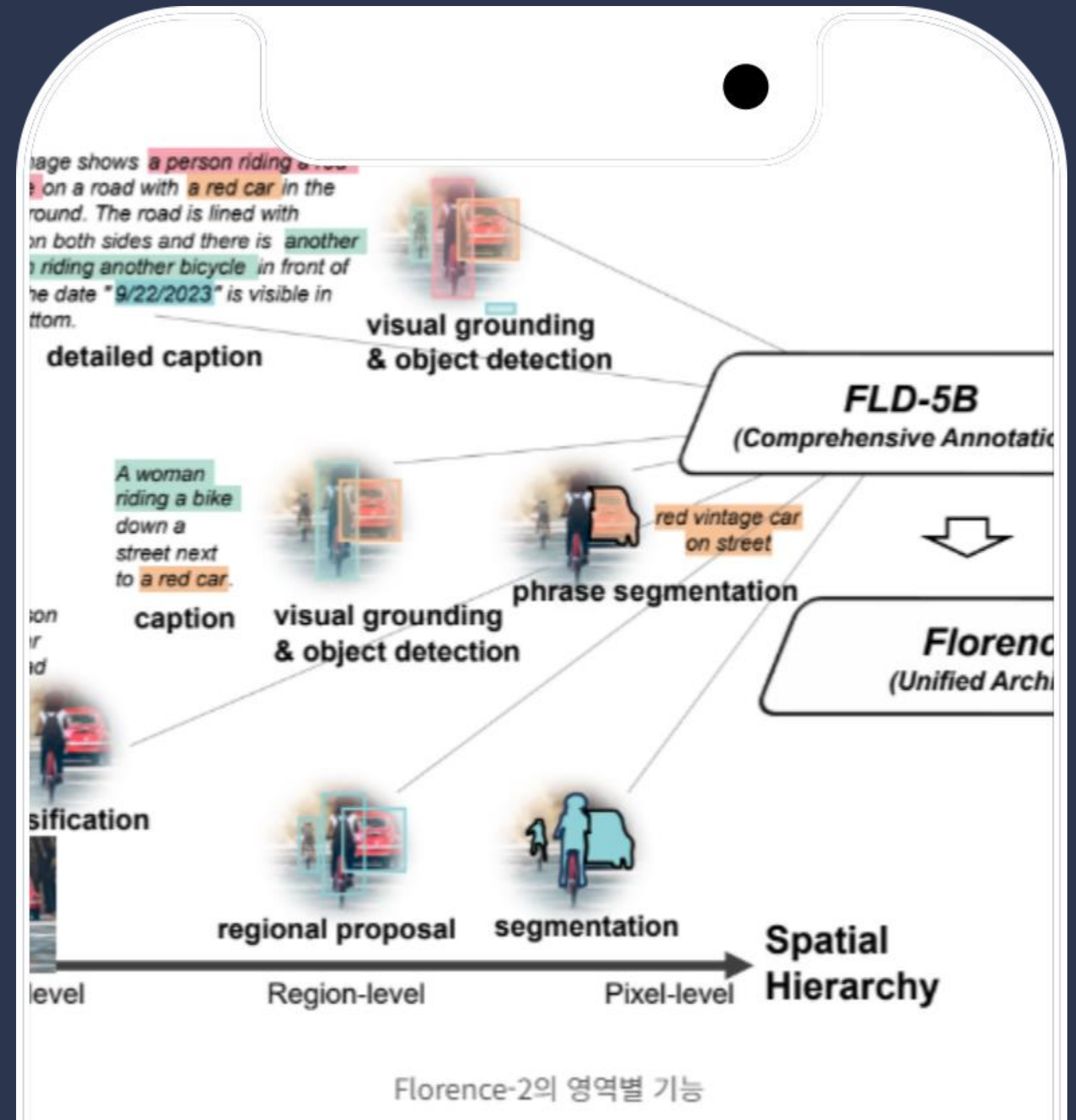
이미지에서 **텍스트**와 **물체의 위치**를 동시에 인식하고 그 결과를 텍스트 형태로 제공

모델 활용 예시



'labels' :[samsung laptop with charging cable plugged into power strip]

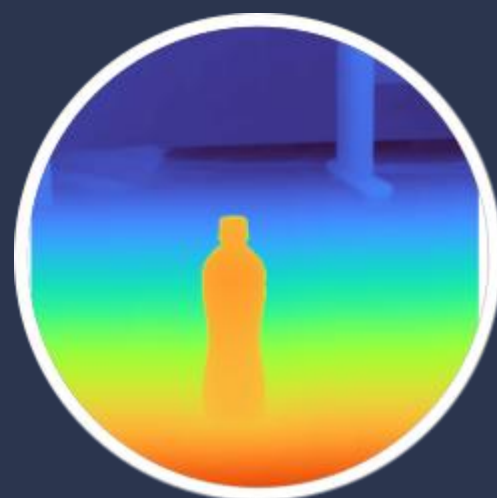
'bboxes' :[490.258, 204.660, 539.200, 292.5]



Depth Pro

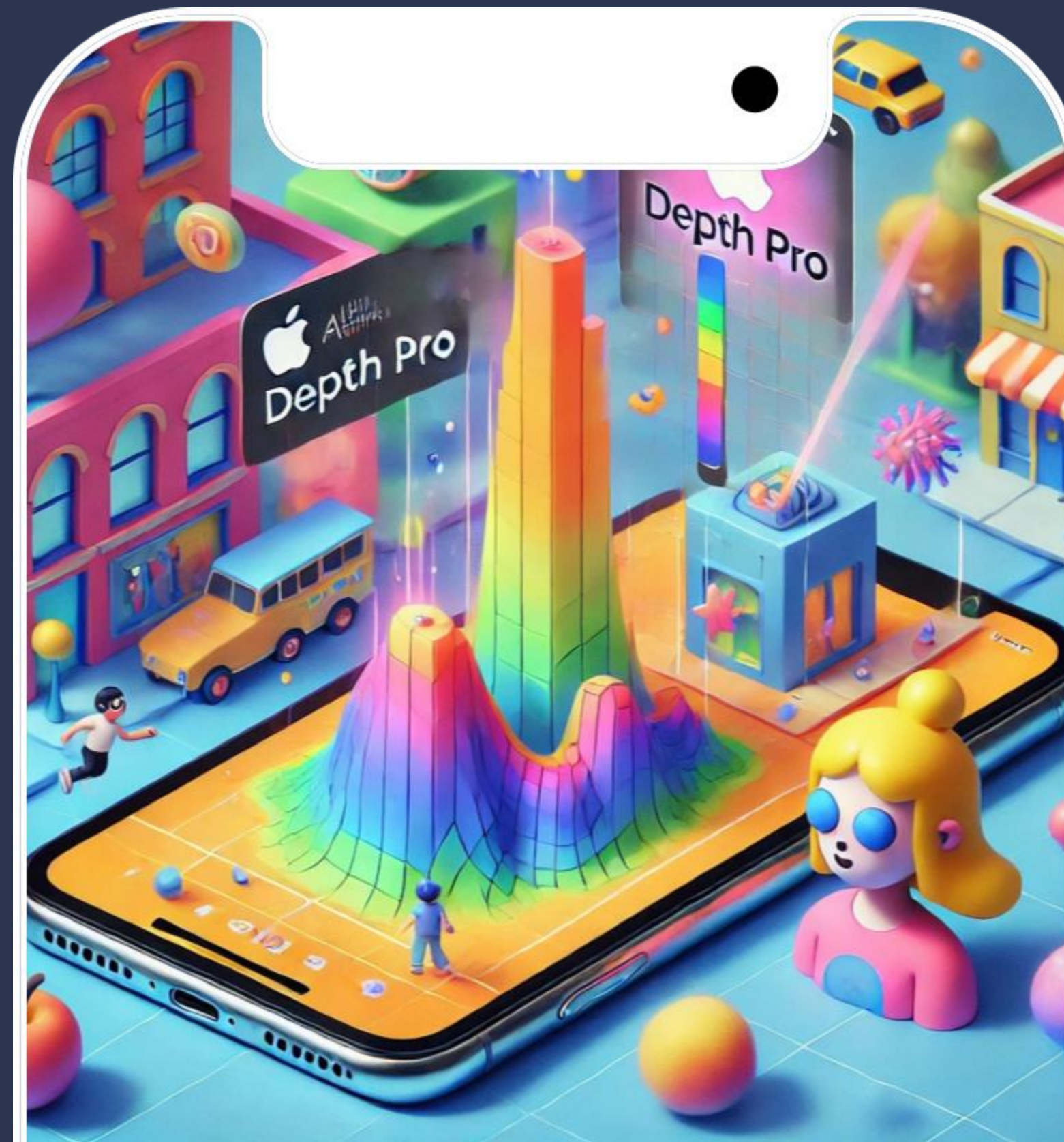
단일 2D 이미지에서 고해상도
3D 깊이 지도 실시간으로 생성

모델 활용 예시



labels :[the bottle]

Actual Distance : 0.3814 meters



GPT-4o

주어진 정보를 기반으로 **추론**을
수행하여 **논리적이고 일관된** 답변을 제공

모델 활용 예시



Enter prompt: put the **coca cola**
can right next **the starbucks cup**
more closely

1. **Rotate**: (" ")
2. **Move foward**: (" ")
3. **Grip**: (" ")
4. **Release**: (" ")





Prompt engineering

07

전체적인 행동 계획

```
class ActionPlanner:
    def generate_robot_movement_plan(self, user_input, im
        distance_difference = coca_distance - starbucks_d

    response = openai.ChatCompletion.create(
        model="gpt-4o",
        messages=[
            {
                "role": "system",
                "content": (
                    "You are an expert in robotic nav
```

system

- ◆ 세부적인 계획이나 명령
- ◆ 일관된 명령

로봇 제어 함수

```
def rotate_robot(self, step):
    response = openai.ChatCompletion.create(
        model="gpt-4o",
        messages=[
            {
                "role": "system",
                "content": (
                    "If the action step contain
                    "return the details in the
                    "If no rotation-related de
```

control

- ◆ Robomaster sdk 제어 함수
- ◆ 값을 통한 제어

최종 제어

```
def plan_actions(self, action_steps):
    for step in action_steps:

        self.rotate_robot(step)
        self.forward_robot(step)
        self.grip_robot(step)
        self.release_robot(step)
```

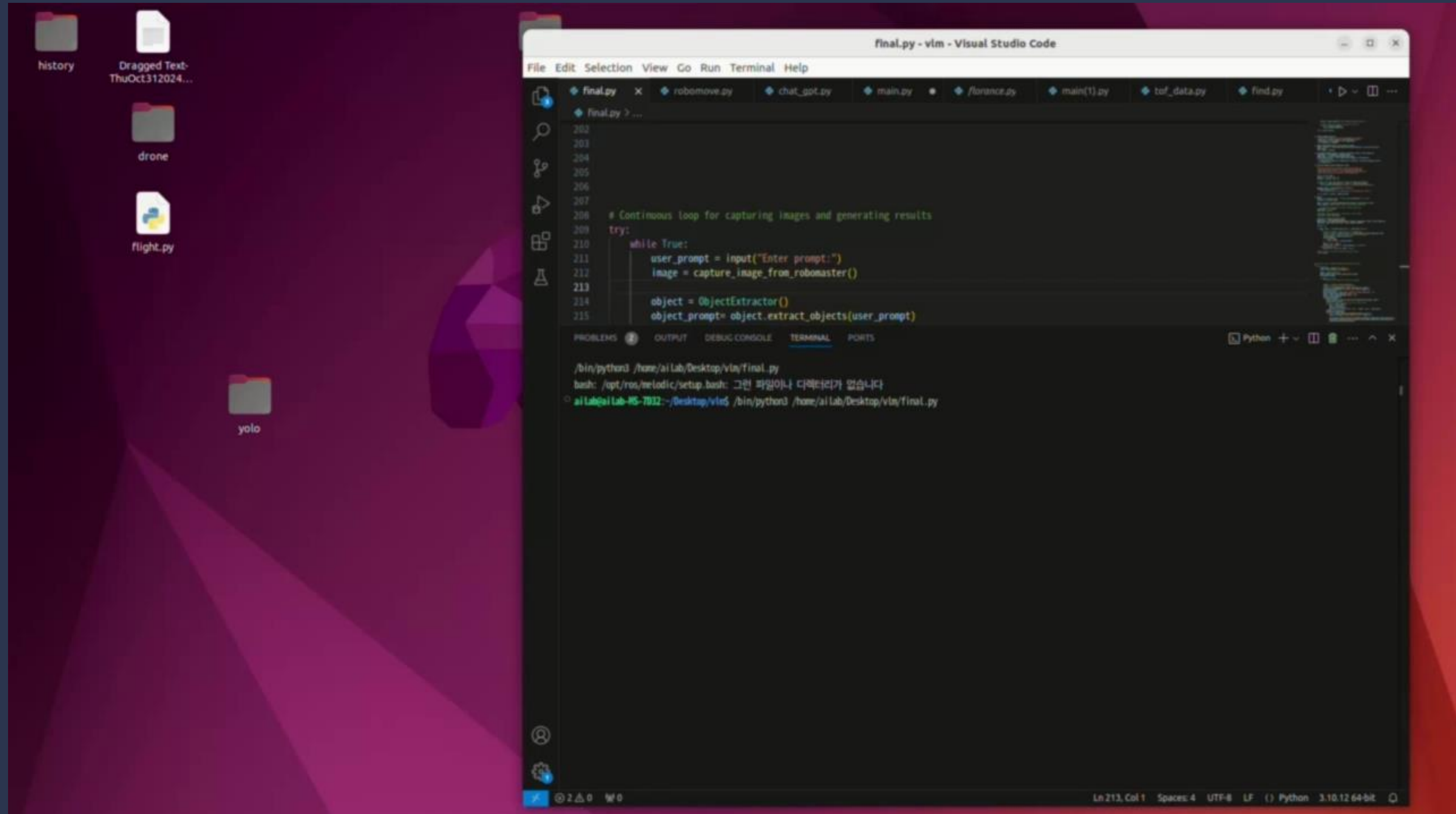
final

- ◆ for 문을 이용한 제어
- ◆ 필요한 제어만 골라서 출력



일반적인 상황 로봇 제어

08





객체 탐지 및 위험 상황 추론

09



사용자 명령(prompt)

Put the coca cola can right next the starbucks cup more closely



상황 분석(Florence-2)

The image shows a black office chair left side a white radiator, in the center of the image white paper, Starbucks logo on it, can of Coca-Cola on the floor



객체 위치 감지(Florence-2)

the coca cola - [490, 204, 139, 292]
the starbucks cup - [274, 139, 313, 241]



위험 상황 추론(chat gpt)

No, there is no indication of aggressive behavior or a disaster situation present



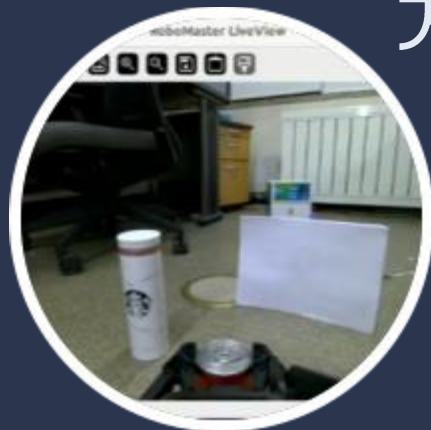
코카 콜라로 오른쪽 회전

1. **Rotate right 33.02 degrees**
the center of the Coca Cola can
 $((490.56 + 539.20) / 2) = 514.88$
using the formula $((514.88 - 320) / 5.9)$



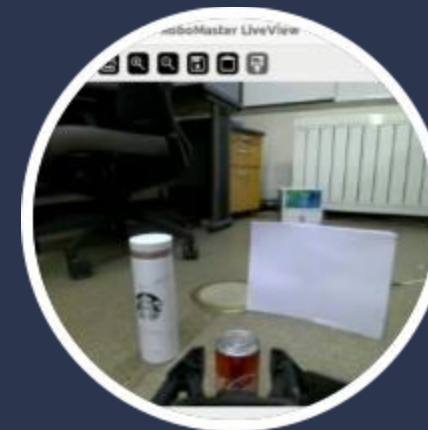
코카 콜라로 접근 후 잡기

2. **Move Forward** Approach the Coca Cola
3. **Grip** Grip the Coca Cola can
Coordinates $[490.56, 204.66, 539.20, 292.5]$



거리 조정 및 스타벅스 컵으로 왼쪽 회전

4. **Calculate distance difference**
 $\text{abs}(0.4029 - 0.6363) = 58.35 \rightarrow$
 $514.88 + 58.35 = 572.23$
5. **Rotate left 47.25**, using the formula
 $((294.4 - 572.23) / 5.9)$



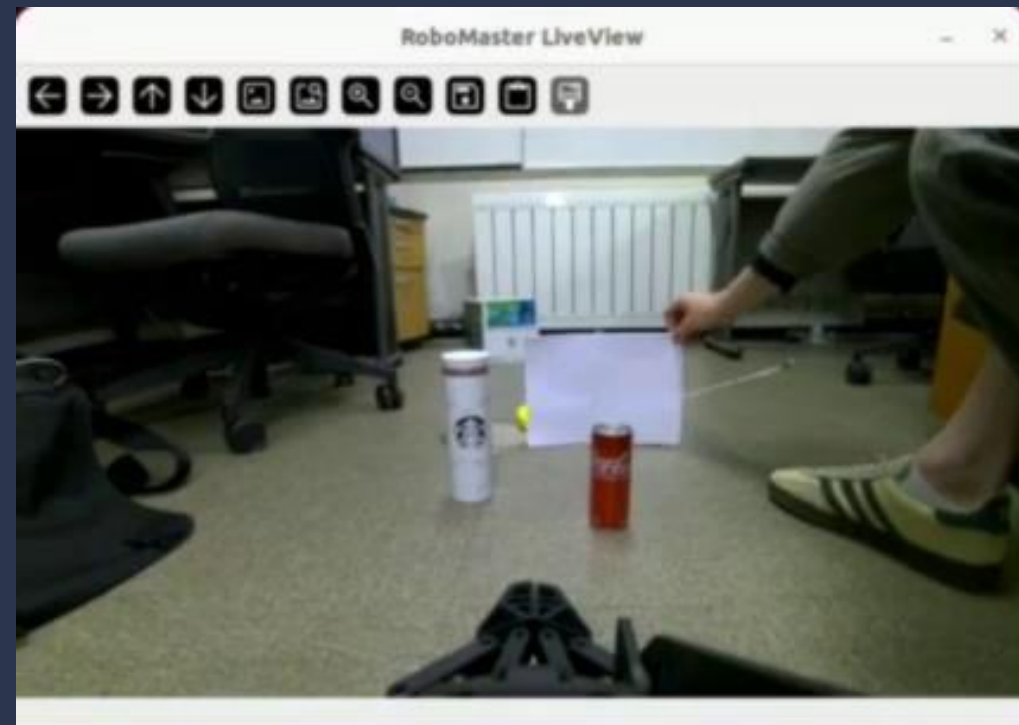
코카 콜라 안전하게 내려놓기

6. **Move Forward** Approach the starbucks cup
7. **Release** Place the Coca Cola can next to the StarBucks cup



재난 상황 가정 로봇 제어

11



```
final.py - vim - Visual Studio Code
File Edit Selection View Go Run Terminal Help

final.py > ...
202
203
204
205
206
207
208 * Continuous loop for capturing images and generating results
209 try:
210     while True:
211         user_prompt = input("Enter prompt:")
212         image = capture_image_from_robomaster()
213
214         object = ObjectExtractor()
215         object_prompt= object.extract_objects(user_prompt)

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Python + - - - - - ^ - X

[{'commands': [{'command': 'turn_left', 'parameters': {'turning_angle': 47.36}}]}]
Executing commands: {'commands': [{'command': 'turn_left', 'parameters': {'turning_angle': 47.36}}]}
Executing turn_left with parameters {'turning_angle': 47.36}
True
Successfully executed turn_left.
No grip details found.
No release details found.
No backward movement detected.
No rotation details found.
Moving forward
Distance: 88
Obstacle detected within 100 cm. Stopping the robot.
No grip details found.
No release details found.
No backward movement detected.
No rotation details found.
No grip details found.
Object: the coca cola, Coordinates: [490.56, 204.66, 539.20, 292.5]
[{'commands': [{'command': 'release_object', 'parameters': {'release_power': 50}}]}]
Executing commands: {'commands': [{'command': 'release_object', 'parameters': {'release_power': 50}}]}
Executing release_object with parameters {'release_power': 50}
None
Failed to execute release_object.
No backward movement detected.
None
Enter prompt:
```



객체 탐지 및 위험 상황 추론

12



사용자 명령(prompt)

Put the coca cola can right next the starbucks cup more closely



상황 분석(Florence-2)

there is a can of Coca-Cola, a White Starbucks bottle, a small TV screen with a fire burning on it



객체 위치 감지(Florence-2)

the coca cola - [361, 184, 393, 258]
the starbucks cup - [267, 139, 306, 241]



위험 상황 추론(chat gpt)

Yes, The fire is depicted on screen and burning is considered a fire hazard



Gpt 4o 로봇 제어

13



코카 콜라로 오른쪽 회전

1. **Rotate right 9.71 degrees** the center of the Coca Cola can $((361.28 + 393.28) / 2) = 377.28$. using the formula $((377.28 - 320) / 5.9)$



코카 콜라로 접근 후 잡기

2. **Move Forward** Approach the Coca Cola
3. **Grip** Grip the Coca Cola can, Coordinates : [361.28, 184.86, 393.28, 258.30]



화재 지역에서 코카 콜라 꺼내기

4. **Move backward** Move to a safe zone with the Coca Cola
5. **Release** Release the Coca Cola in the safe zone
6. **Rotate** Calculate the angle to the Starbucks Cup



화재 지역에서 스타벅스 컵 꺼내기

7. **Move Forward** Approach the starbucks cup
8. **Grip** Grip the Starbucks Cup at Coordinates : [267.34, 139.14, 306.24, 240.3]
9. **Move Backward** Move to a safe zone with the Starbucks cup

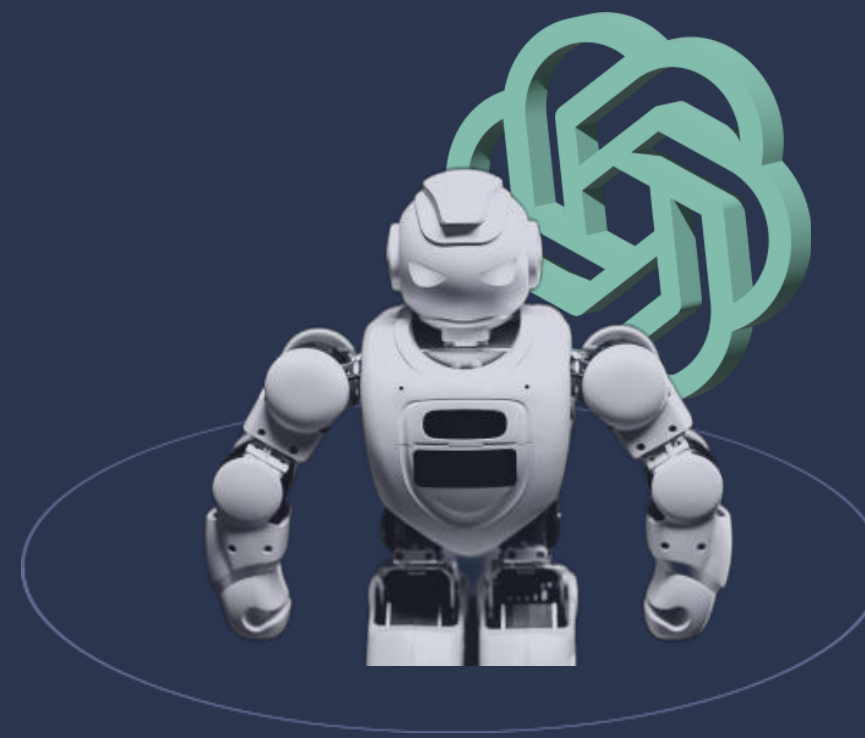


결론 및 기대효과

14



기존 로봇 제어



LLM 로봇 제어

| | | | | |
|-----------------------------|--|-------------|--|----------------------------|
| 낮음(제한적) | | 자연어 처리 능력 | | 높음(향상됨) |
| 낮음(정해진 작업에 한정) | | 상황 인식 및 적응성 | | 높음(유연하고 다양한 상황 대응) |
| 중간(주어진 상황에만 가능, 정확도 비교적 높음) | | 복잡한 문제 해결 | | 중간(다양한 상황 효과적, 정확도 비교적 낮음) |
| 낮음(특정 작업에 최적화) | | 다양한 작업 수행 | | 높음(다목적 활용 가능) |



결론 및 기대효과

15

STRENGTHS

Zero-shot 방식, 프롬프트 엔지니어링 활용
학습되지 않은 환경에서도 유연한 대응 가능
객체 유사도 검정과
위험 상황 인지를 통해
안전하고 신뢰성 높은 시스템을 구축



WEAKNESSES

상대적으로 낮은 정확도 문제
고성능 컴퓨팅 기술 필요
연산 처리 지연



OPPORTUNITIES

재난 상황과 같은 빠르게
변하는 환경에서 효과적인
대응과 복구 작업 가능



THREATS

데이터 처리 과정에서
발생할 수 있는 보안 위협
및 데이터 유출 위험



THANK YOU!



거대 언어모델(LLM) 기반
로봇 제어 자율 주행 시스템 개발

AI-ROBOTIC

