

2026. 1. 14

자율주행 경진대회

캡스톤 디자인 2 기말발표

LIMO Pro

문정석, 박건우

CONTENTS

01.

목표 및 현황 분석

문제점
개발 목표

02.

코스 별 개선점

개선 내용
결과 분석

03.

전체 주행 평가

완주 결과 분석

04.

랩타임 단축

실험 설계
결과 분석

05.

결론

한계점 및 향후 과제
역할 분담

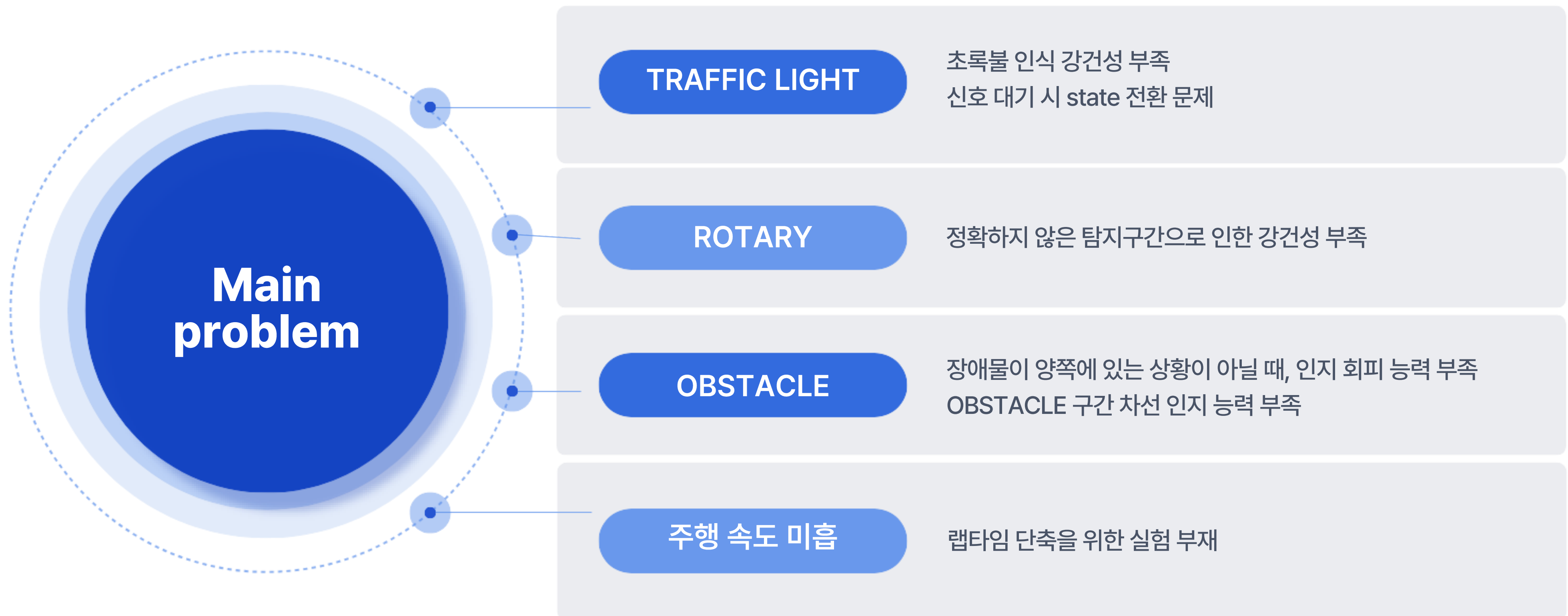
01.

목표 및 현황 분석

▶ 문제점

개발 목표

주행 강건성 부족 및 주행 속도 미흡



01.

목표 및 현황 분석

문제점

▶ 개발 목표

전체 완주 강건성 향상 및 랩타임 단축



02.

코스 별 개선점

▶ 개선 내용

결과 분석

개선 내용

☑️ 실험 setting

- ☑️ Trial 횟수: 각 코스 당 10회
- ☑️ 동일 조건 초기화

☑️ 성공 기준 선정

Node	성공 기준
Line_Tracing	차량의 4바퀴 중 하나라도 나가지 않거나 차선 검출을 실패하지 않는 경우
Traffic	빨간불일 경우 stop이 풀리는 오검출이 0회
Pedestrian	보행자가 지나갈 때 멈추고 다 지나간 후에 정상적인 주행 복귀
Rotary	앞의 차량과 부딪히지 않으면서 안정적인 주행
Obstacle_Avoidance	장애물과 충돌 0회 / 회피 후 차선 복귀 및 유지
Parking	주차 시퀀스 수행 후 모든 바퀴가 주차 구간 내 위치

02.

코스 별 개선점

▶ 개선 내용

결과 분석

개선 내용

☑ 측정 결과

Node	성공률(%)
Line_Tracing	60%
Traffic	60%
Pedestrian	100%
Rotary	70%
Obastacle_Avoidance	40%
Parking	70%

☑ 실패 원인 분석

Node	주요 원인
Line_Tracing	여러 차선이 있는 경우에 대한 차선 검출 실패
Traffic	조도 변화/빛 반사로 인한 HSV 변화
Pedestrian	X
Rotary	정확하지 못한 탐지 구역 범위 설정
Obastacle_Avoidance	Safe distance, scan angle과 같은 parameter 튜닝 실패
Parking	하드코딩으로 인한 실험 재현성 부족

02.

코스 별 개선점

▶ 개선 내용

결과 분석

LANE TRACING 개선



< 수정된 알고리즘 >

Before

- 기존 lane에 fitting된 곡선의 곡률에 따라 왼쪽차선인지 오른쪽차선인지 구분
- 곡률이 0에 가까울 때 잘못 잡는 경우가 종종 생김

After

- 단순히 fitting된 라인의 시작 위치를 보고 결정

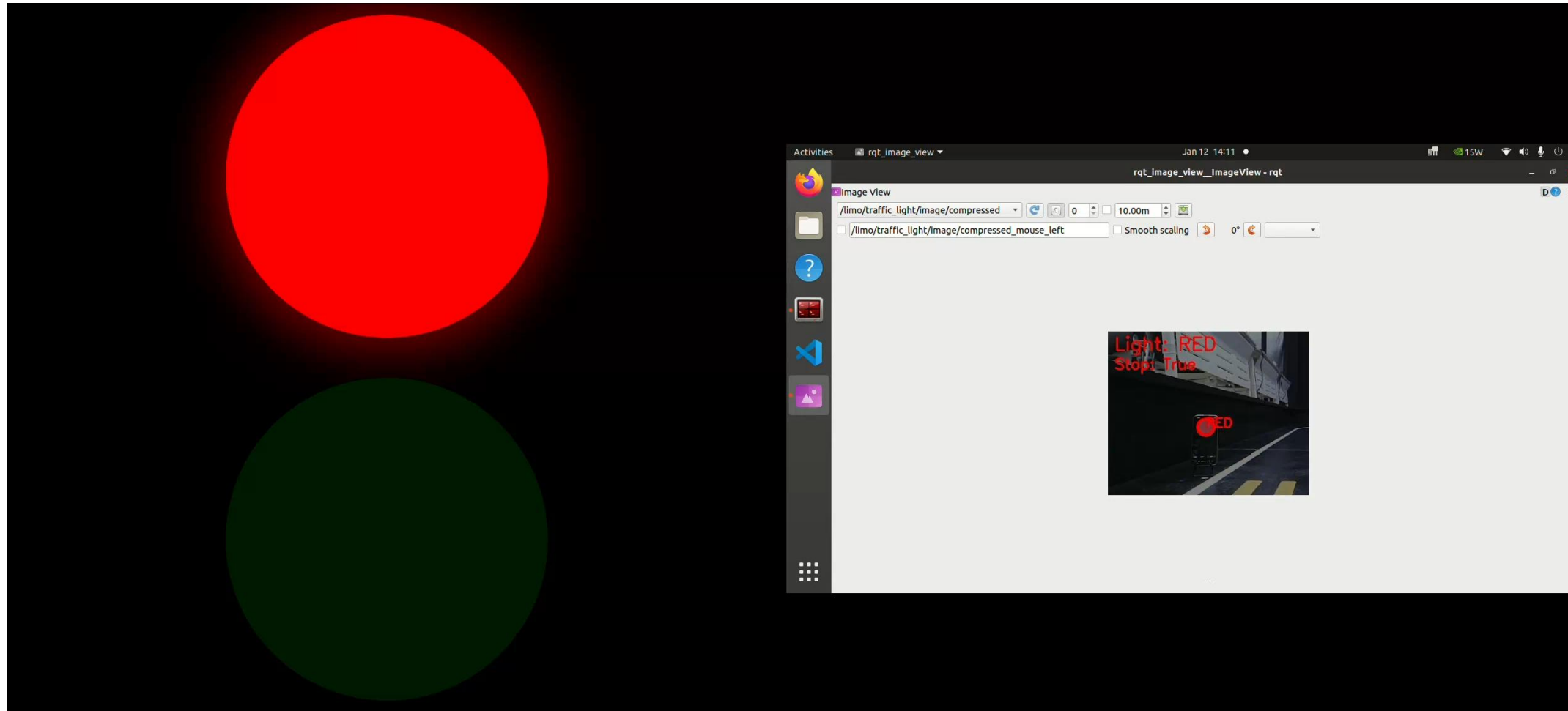
02.

코스 별 개선점

▶ 개선 내용

결과 분석

TRAFFIC LIGHT 개선



< 수정된 알고리즘 >

Before

- 정확하지 않은 HSV값 범위로 인한 신호등 인지 강건성 부족
- 신호대기시 노란선 탐지로 인한 잘못된 State 전환

After

- 밝은 화면을 인지하기 위해 Green S값 범위 수정
- 신호대기시 State 전환 X, 초록불 탐지시 State 3초간 고정

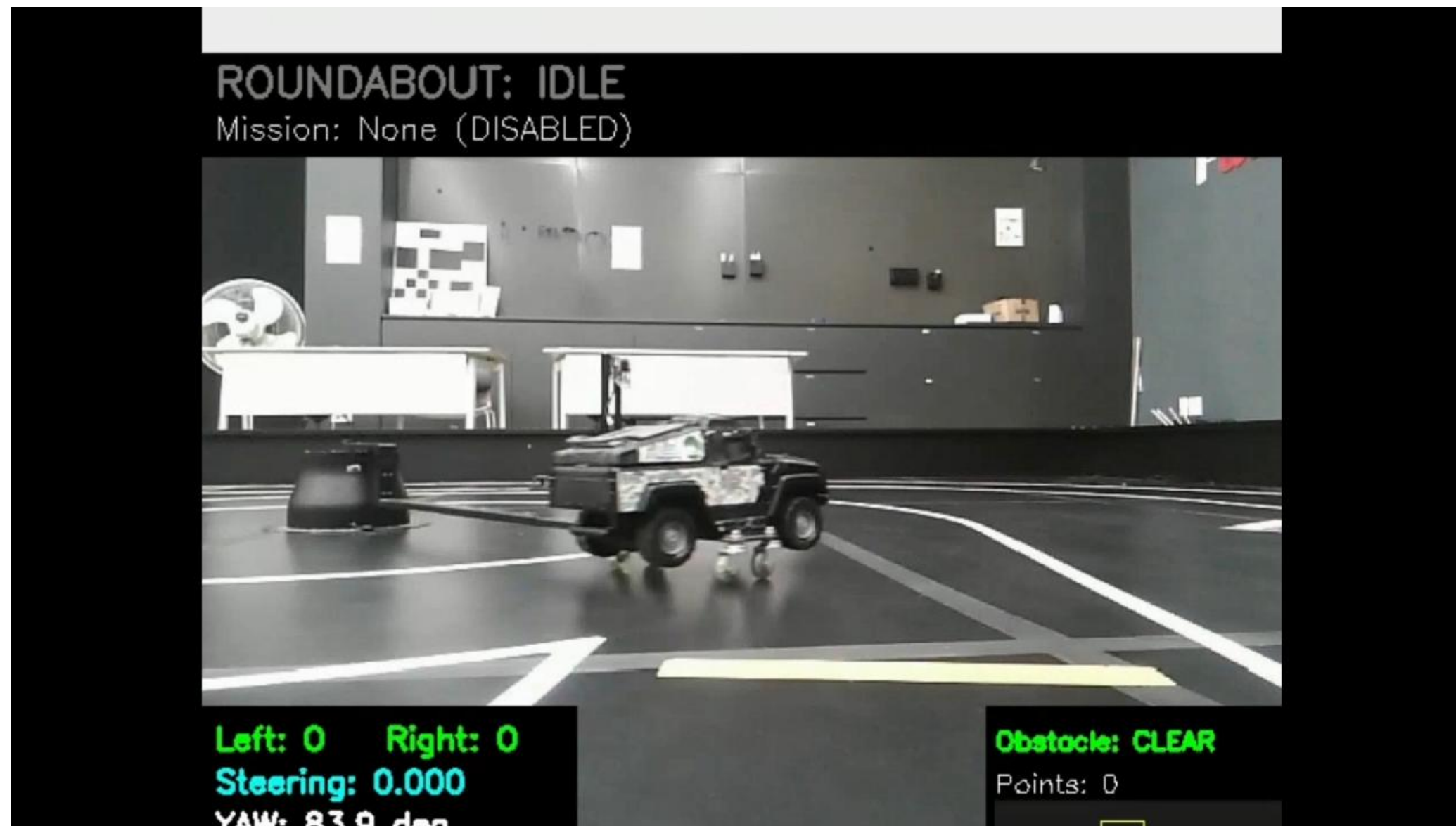
02.

코스 별 개선점

▶ 개선 내용

결과 분석

ROTARY 개선



< 수정된 알고리즘 >

Before

- 잘못된 탐지구역으로 인한 불필요한 정지 발생

After

- 탐지구역 범위 개선

02.

코스 별 개선점

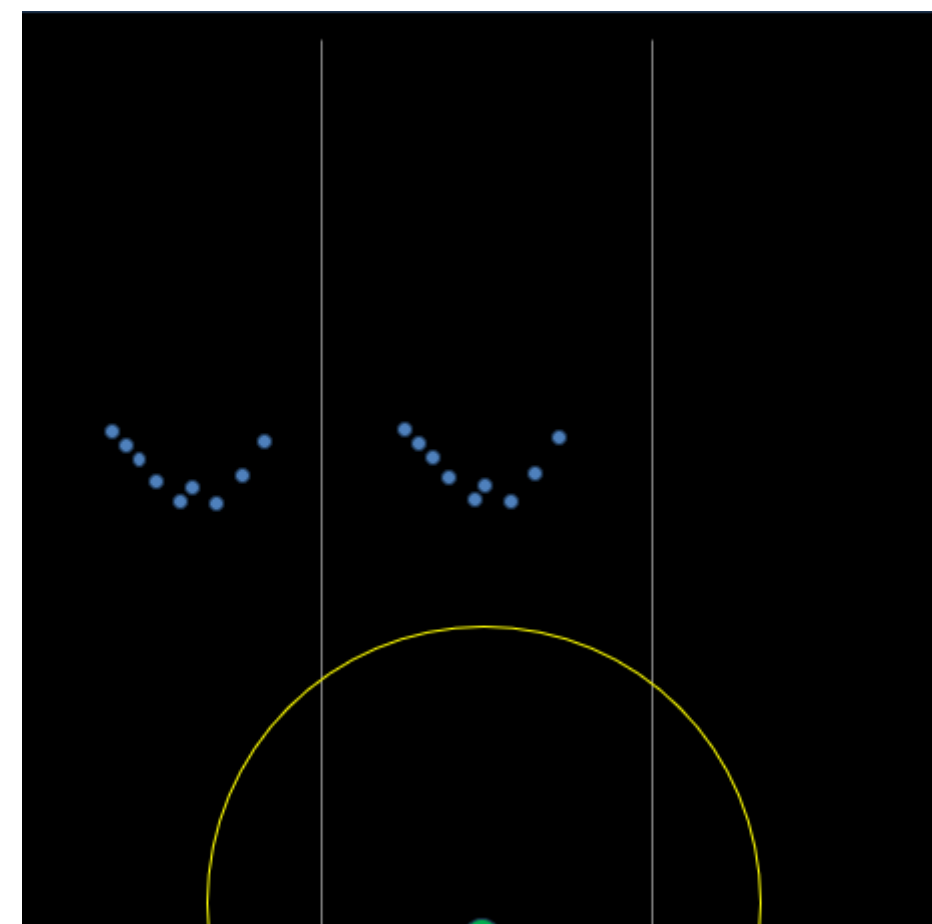
▶ 개선 내용

결과 분석

OBSTACLE AVOIDANCE 개선



< 기존 알고리즘 >



< 수정된 알고리즘 >

Before

- 기존 알고리즘에서는 장애물 사이의 Gap 또한 갈 수 있는 길로 인식하는 문제가 있었음

After

- 기존 알고리즘을 수정하여 섹터를 3개로 나누고 DBSCAN을 통해 장애물이 있는 섹터는 점유된 상태로 두어, 장애물이 없는 섹터를 골라 주행을 할 수 있도록 변경함

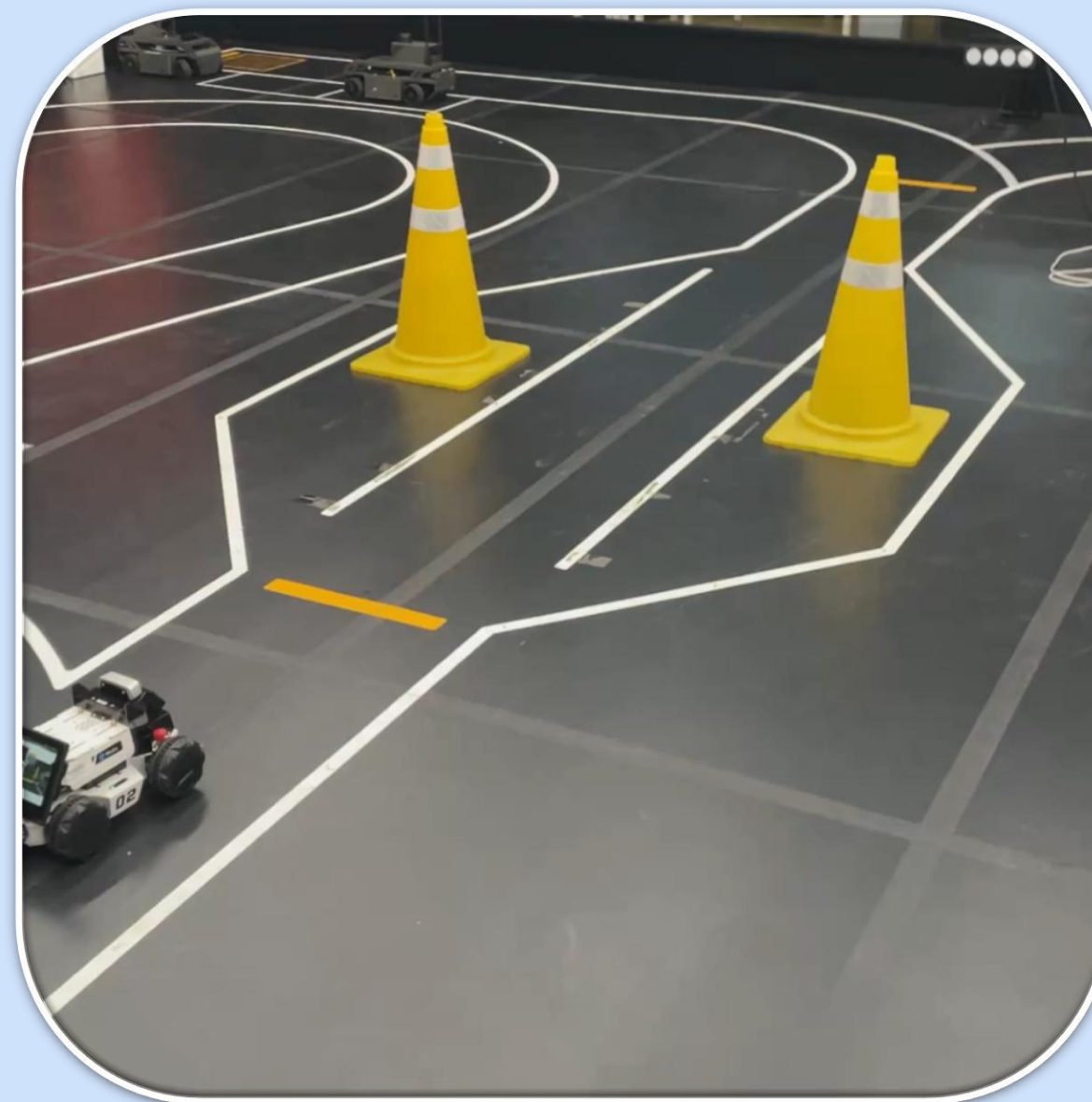
02.

코스 별 개선점

▶ 개선 내용

결과 분석

OBSTACLE AVOIDANCE 개선



< 수정된 알고리즘 >

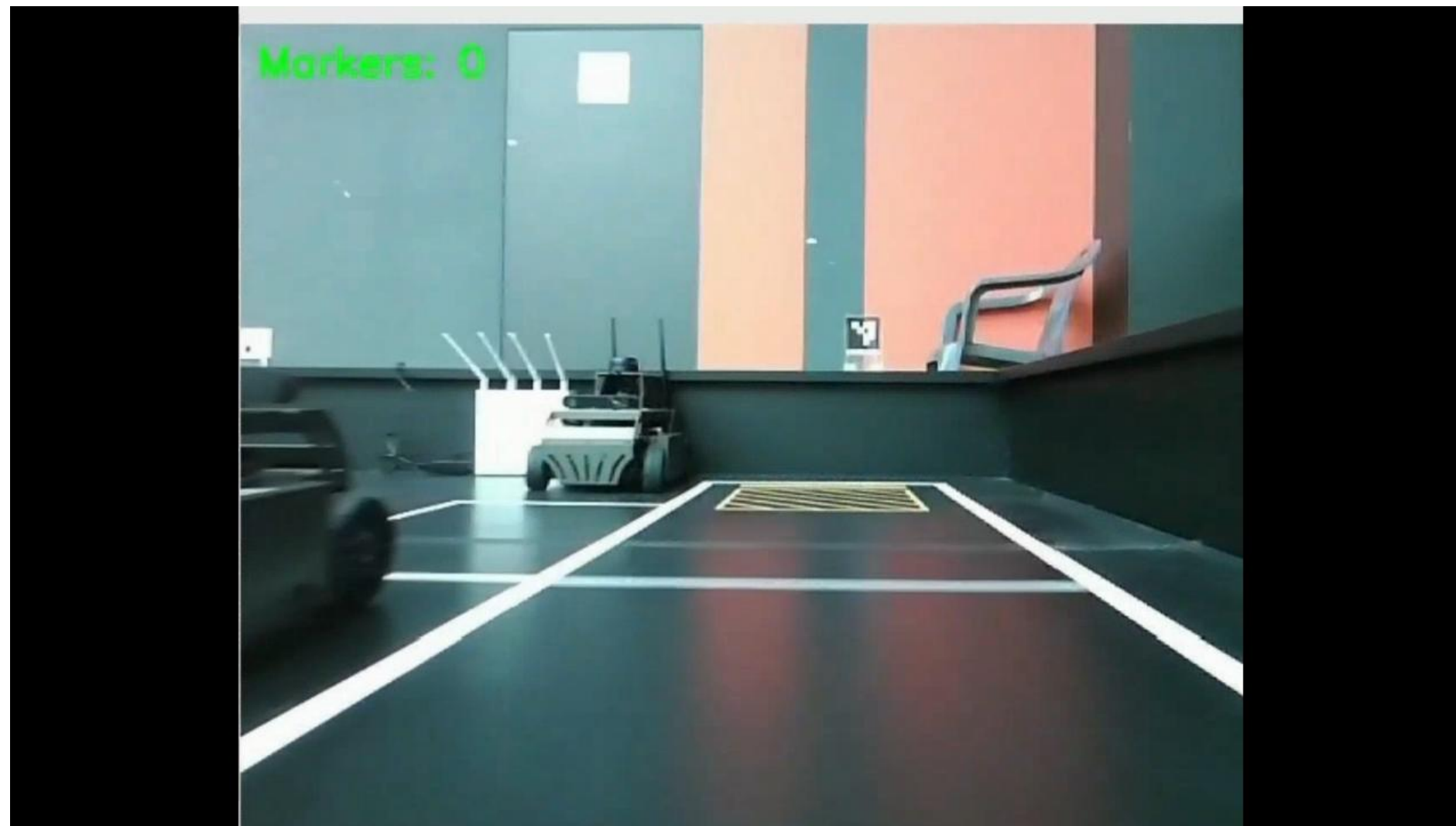
02.

코스 별 개선점

▶ 개선 내용

결과 분석

PARKING 개선



< 수정된 알고리즘 >

Before

- Hard coding된 parameter값 강건성 부족
- 주차동작시, Lane tracing node와 cmd_vel 발행 충돌

After

- Parameter값 수정
- 아루코마커 인식시, Lane tracing node는 cmd_vel 발행 종료

02.

코스 별 개선점

개선 내용

▶ 결과분석

결과 분석

중간 발표 대비 결과 분석

☑ 개선 전 측정 결과

Node	성공률(%)
Line_Tracing	60%
Traffic	60%
Pedestrian	100%
Rotary	70%
Obastacle_Avoidance	40%
Parking	70%

☑ 개선 후 측정 결과

Node	성공률(%)
Line_Tracing	80%
Traffic	80%
Pedestrian	100%
Rotary	100%
Obastacle_Avoidance	90%
Parking	90%

☑ 전 구간 성공률 개선

☑ 주행 강건성 확보

03.

코스 별 개선점

▶ 완주 결과 분석

완주 결과 분석

중간 발표 대비 완주 결과 분석

☑ 개선 전 측정 결과

Obstacle	성공률(%)
왼쪽	0%
가운데	80%
오른쪽	0%

완주 시간
3분 7초

☑ 개선 후 측정 결과

Obstacle	성공률(%)
왼쪽	100%
가운데	90%
오른쪽	100%

완주 시간
2분 57초

☑ 전 구간 성공률 개선

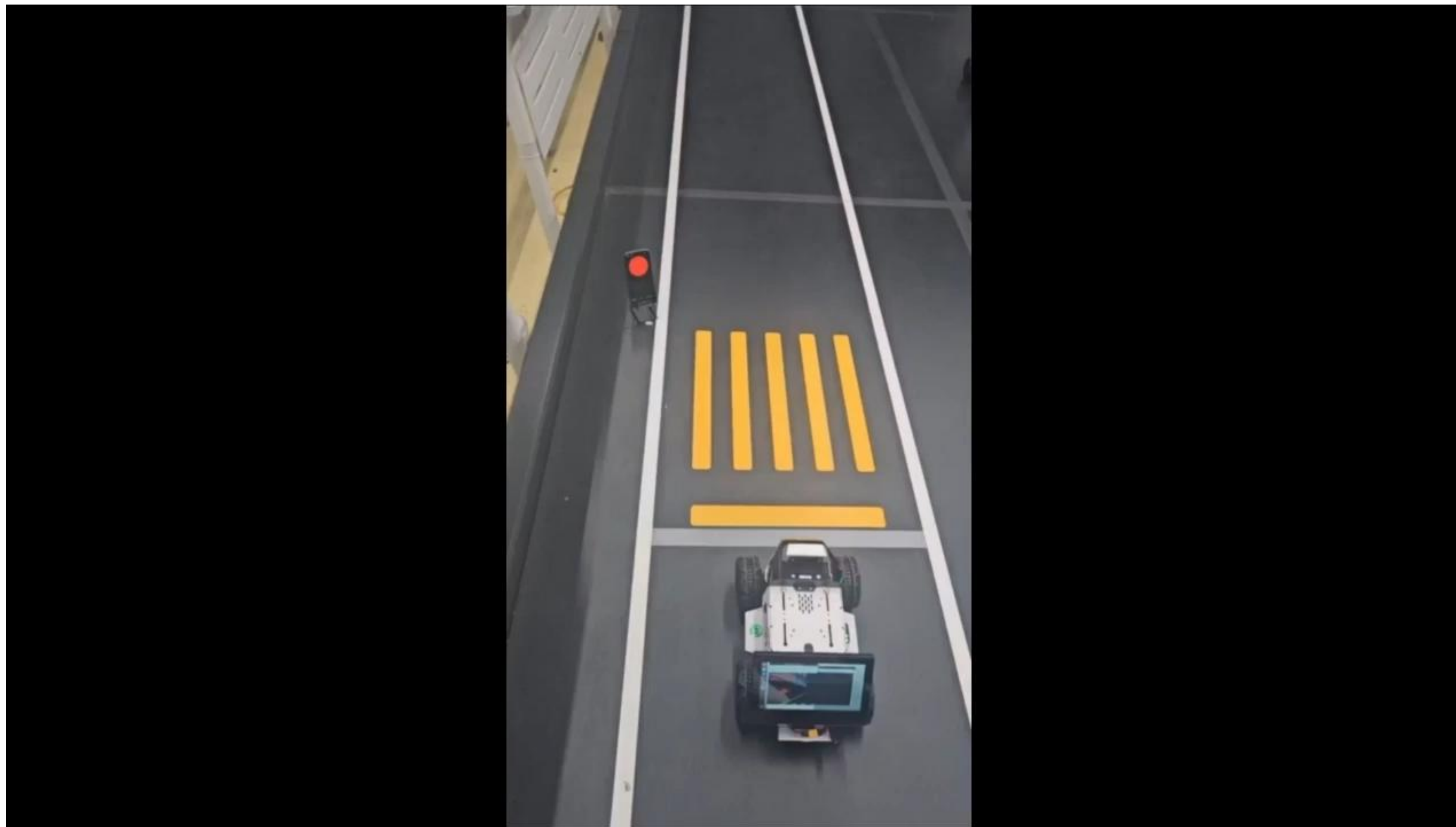
☑ 주행 강건성 확보

03.

코스 별 개선점

▶ 완주 결과 분석

주행 영상



< 4배속 완주 영상 >

04.

랩타임 단축

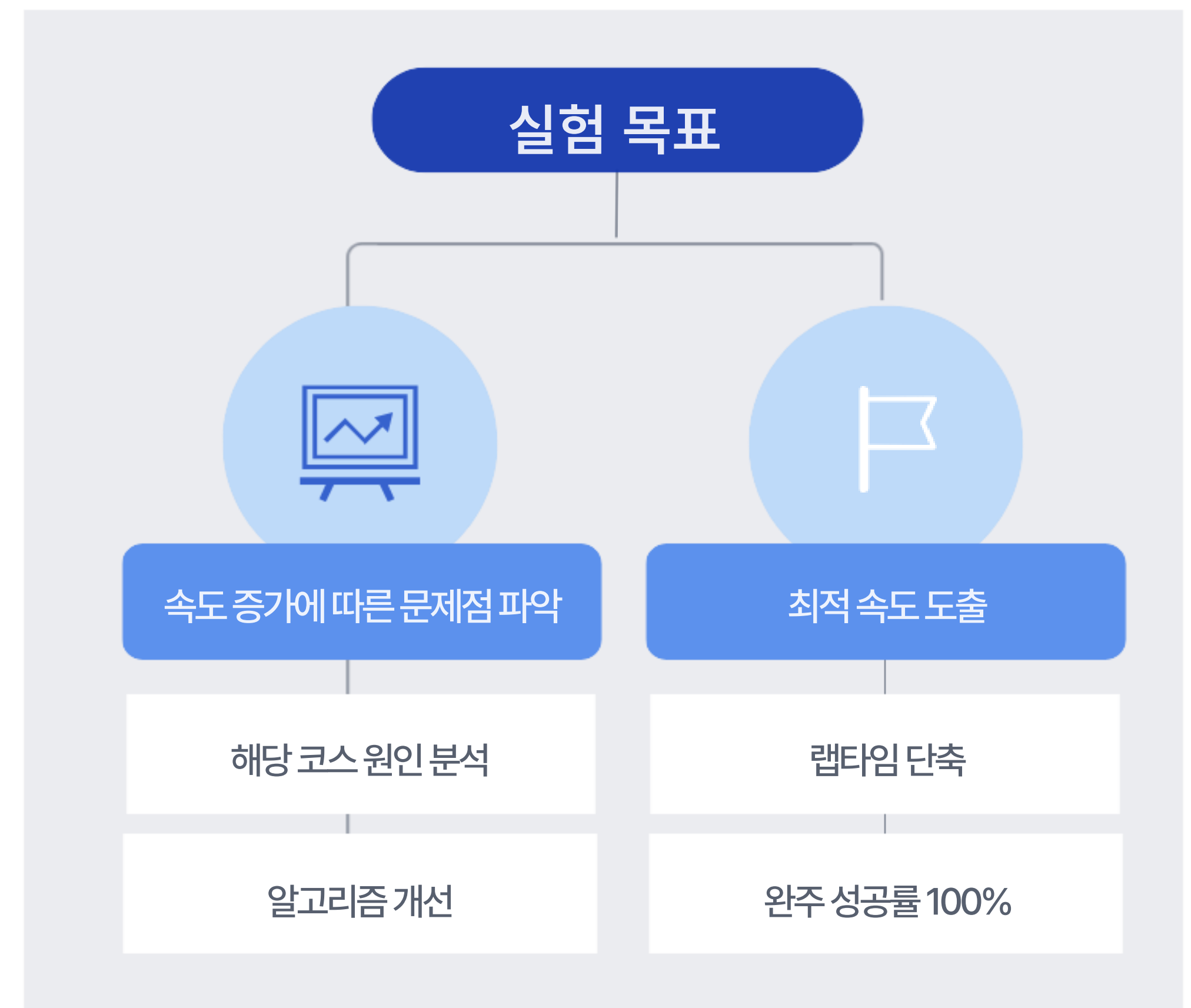
▶ 실험 설계

결과 분석

랩타임 단축을 위한 실험 설계

주행 속도를 제외한 동일 조건에서 실험 진행

주행 속도	주행 속도	주행 속도
0.3	0.5	0.8

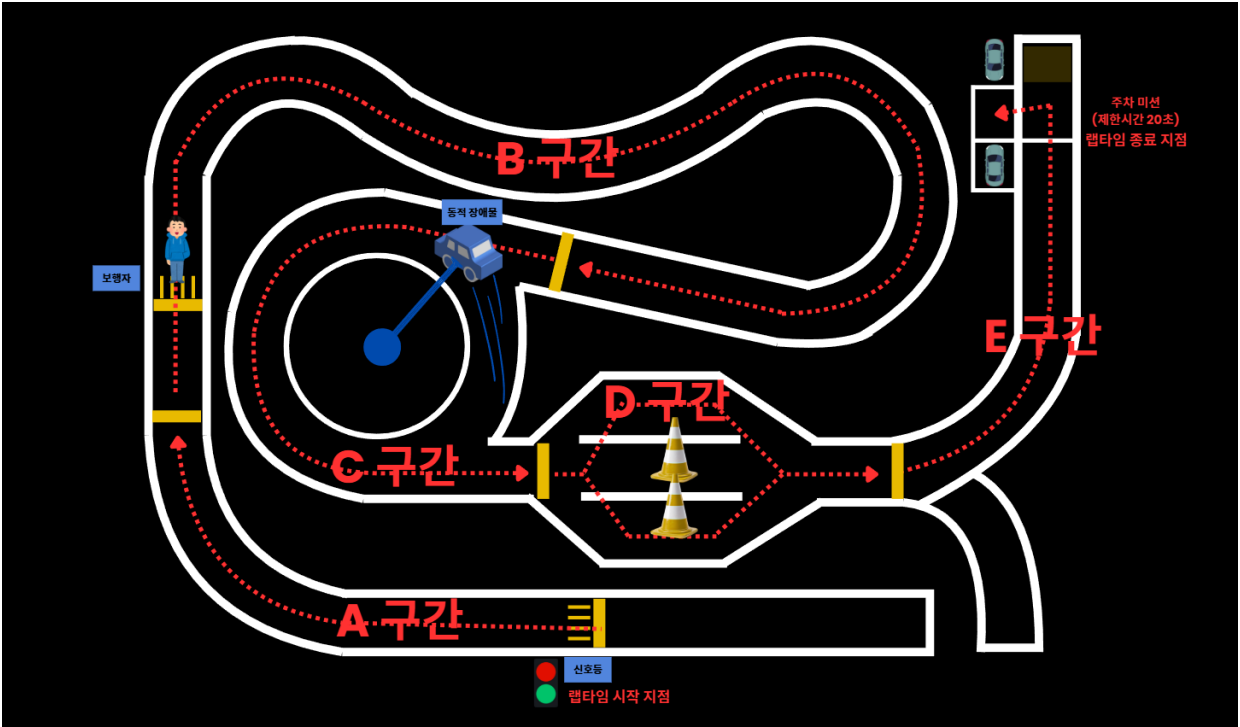


결과 분석

결과 분석

✓ 측정 결과

주행 속도	0.3	0.5	0.8
성공률(%)	90%	80%	30%



실패

- 특정 구간에서 라인 팔로잉을 실패함
- 고속 주행시, 보행자 충돌

원인

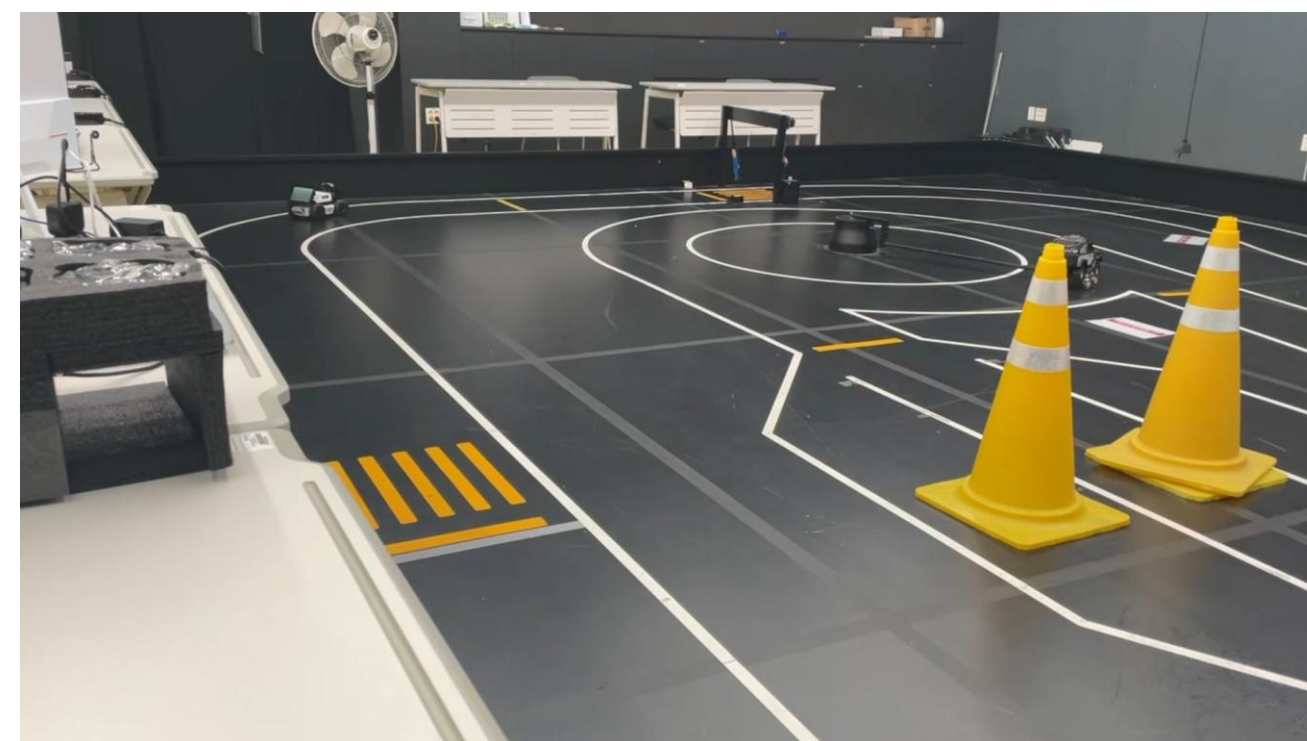
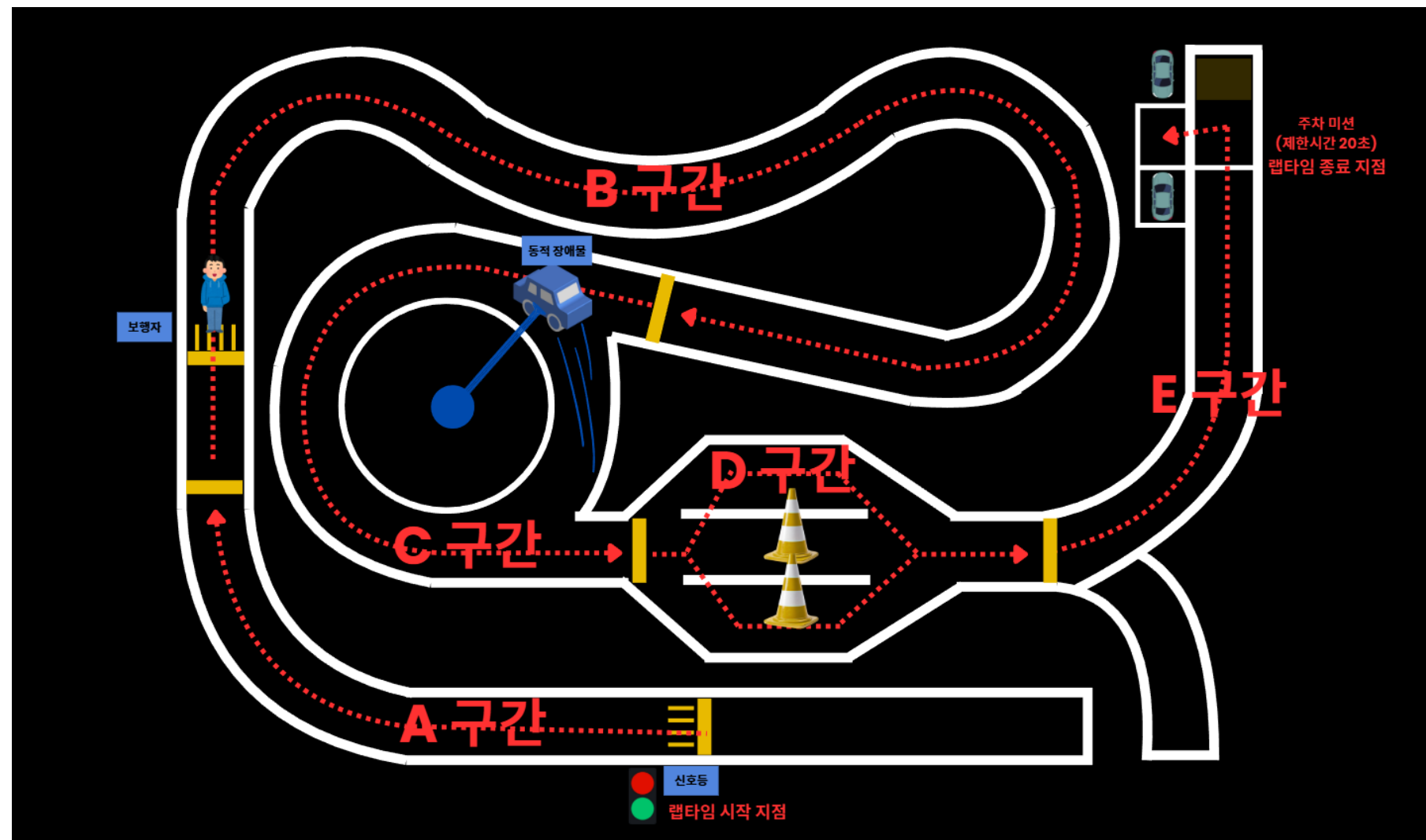
- 여러 차선이 보이는 경우 잘못된 라인 판별로 주행경로가 적절하게 생성되지 않음
- 고속으로 주행시 보행자가 라이다 범위 밖에 위치

04. 랩타임 단축

실험 설계

▶ 결과 분석

~NODE 수정



< A~B구간에서의 속도 변화>

Before

- 0.8의 속도로 주행 시, Pedestrian 구간에서 보행자와 충돌
- 0.8의 속도로 주행 시, Rotary에서 보행자기물 인식후 정지
- 0.8의 속도로 주행 시, Obstacle, Parking구간에서 차선 인식 오류

After

- 0.8의 속도로 전 구간을 주행하는 것이 아닌, 구간 별로 속도를 변화
- A구간: 0.8, B구간보행자인식전: 0.3, 인식 후: 0.8, C구간: 0.5, D구간: 0.5, E구간: 0.5
- 각 구간 최적 속도는 Trial and error 방식으로 찾음

04.

랩타임 단축

실험 설계

▶ 결과 분석

주행 영상



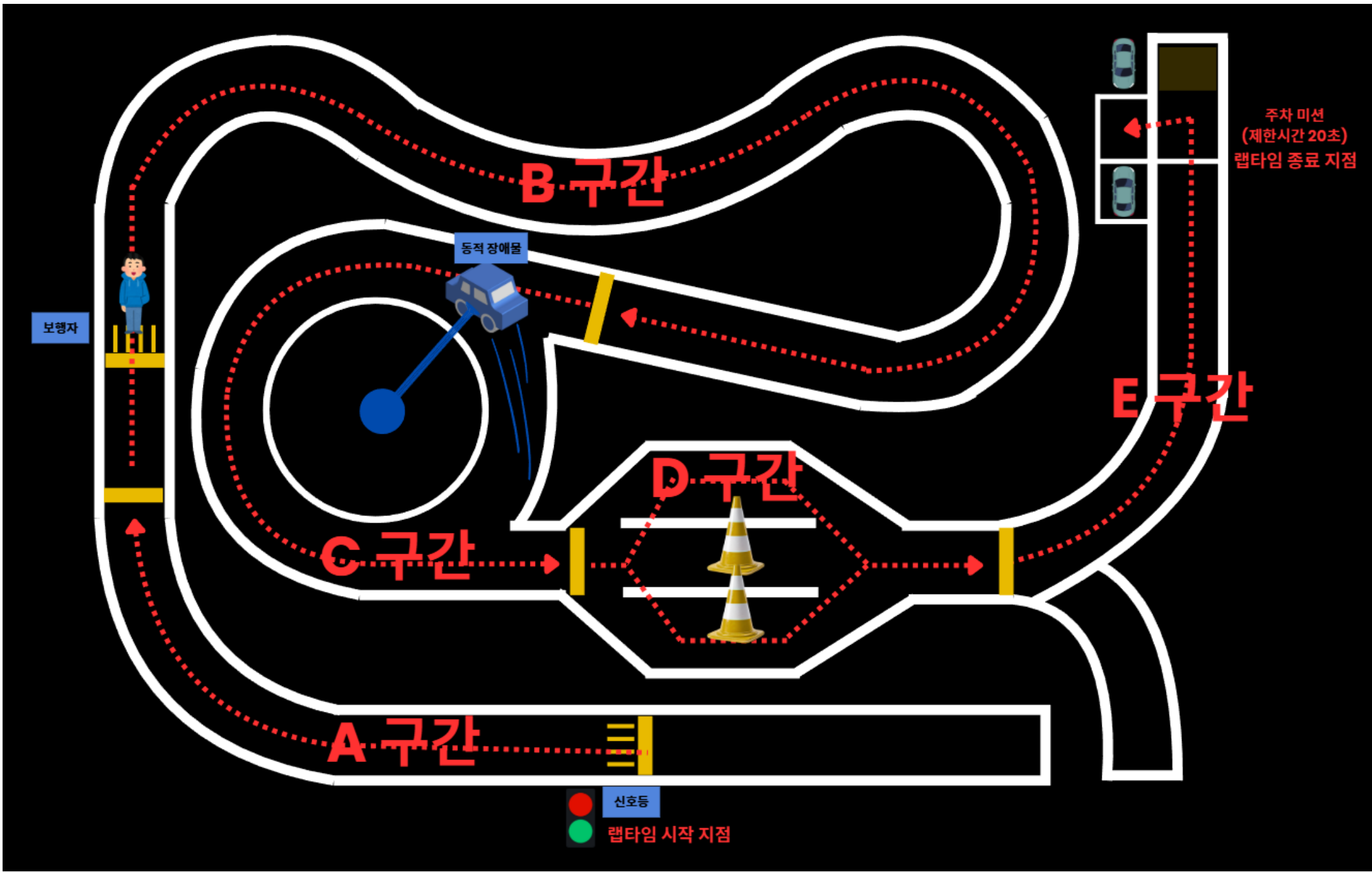
< 랩타임 최적화된 완주 영상 >

04.
랩타임 단축

- 실험 설계
- ▶ 결과 분석

결과 분석

주행 결과



구간	주행 시간
A	9.57 s
B	34.23 s
C	21.74 s
D	22.53 s
E	9.28 s
주차	14.02 s
전체 완주	94.93 s

< 5번 측정 평균 주행 결과 >

05.

결론

▶ 한계점 및 향후 과제

역할 분담

한계점 및 향후 과제

21

Remaining Issues



- ☑ Trial and error 방식으로 인한 global 최적 속도값을 찾지 않았다.
- ☑ 경진대회를 위해 주차를 Hard coding을 수행한다.
- ☑ State 전환 안정화를 위해 특정 구간에서 Hard coding을 수행한다.

Future Growth Work



- ☑ 프로젝트를 결과물로 정리하기 (ex. GitHub)
- ☑ 강화학습으로 연결해보기(상위: FSM, 하위: RL policy)
- ☑ 실패 케이스 중심 분석하기

05.

결론

한계점 및 향후 과제

▶ 역할 분담

역할 분담

22

문정석

- 역할

Traffic light node 작성

Rotary node 수정

Parking node 작성

State 판별 노드 수정

PPT 발표자료 작성

박건우

- 역할

Lane Detection 및 Tracing 노드 작성

Obstacle avoidance 노드 작성

약간의 State 판별 노드 작성

Pedestrian node 작성

PPT 발표자료 작성

Thank you!