

모바일로봇프로그래밍 프로젝트 결과보고서

과제명	도로 위 로봇 자율주행 시뮬레이션
참여인원	(총 05명) 학부과정 05명(안준영, 우정우, 이지원, 전우혁, 차수진)
추진배경	<p>○ 모바일로봇의 경로계획, 센서정보처리, 위치인식 등의 모바일 로봇의 지능제어를 적용하고 응용하고자 함</p> <p>○ 자율주행은 보다 더 편리하고 안전한 차량을 추구하는 인간의 지속적인 기대와 함께 고령화 시대가 가속화되면서 더욱 더 커지고 있다. 최근 자동차업계에서 자율주행이 주요 트렌드로 자리매김한 배경이며 이러한 트렌드를 따라 모바일 로봇을 통해 자율주행을 구현하여 모바일 로봇의 지능제어와 최신 자율주행 트렌드를 적용해보고자 한다.</p>
목표 및 내용	<p>○ 모바일로봇의 경로계획, 센서 정보처리를 위해 고속 도로 위 차량의 상황을 시뮬레이션 한다.</p> <ul style="list-style-type: none"> - 차선을 제시하고 해당 차선을 따라 모바일 로봇이 자율 주행하도록 한다. - 차선 변경 기호 등을 인식하고 기호에서 제시한 행동에 따라 모바일 로봇의 이동을 구현한다. - 모바일 로봇의 전방에 장애물이 출현하거나 존재가 확인될 경우의 상황에 대해 행동을 제시한다. - 고속 도로 위 톨게이트 등의 상황을 인식하고 적절한 행동을 제시한다.
수행결과	<p>○ 자율주행</p> <ul style="list-style-type: none"> - 로봇이 두 차선을 모두 인식하고, 해당 차선의 중앙값이 설정한 범위를 벗어나지 않으면 직진을 한다. - 설정한 범위를 벗어날 경우, 차선의 중앙보다 왼쪽으로 벗어났으면 우회전을 하고, 오른쪽으로 벗어났다면 좌회전을 한다. <p>○ 화살표 인식</p> <ul style="list-style-type: none"> - 화살표 방향을 인식 후 그에 따른 로봇의 행동을 제시한다. 좌회전이나 우회전 표지판을 인식할 경우, 로봇에게 현재 속도에 따른 적절한 회전 값을 부여하고, 잠시 차선 인식을 멈춘다. <p>○ 전방 물체 인식</p> <ul style="list-style-type: none"> - 로봇이 바라보는 방향에서 (-20도 ~ 20도) 사이의 장애물이 인식되고 그 장애물까지의 거리가 20cm 이하일 경우 로봇의 움직임을 멈춘다. 그리고 장애물이 인식되지 않으면 다시 로봇이 자율 주행하도록 한다. <p>○ 요금 정산소 인식</p> <ul style="list-style-type: none"> - 로봇이 바라보는 방향을 기준으로 오른쪽, 왼쪽에 물체가 존재가 인식되고 거리가 35cm 내 이면 요금 정산소로 인식한다. 그 후, 정산소를 지나가기 까지 로봇의 속도를 낮추어 천천히 주행하도록 한다.

1. 과제 수행 배경

국제전자제품박람회(CES)에서는 도요타, 벤츠, 포드, 현대자동차 등 자동차 제조업체와 보쉬, 덴소 등 부품업체의 대거 참여로 자율주행자동차가 행사의 주요 테마로 자리매김하였다. 구글, 애플, 인텔, 엔비디아 등 ICT대표 기업들도 자율주행 제품 소개 및 시연에 적극 참여하며 자율주행 기술 개발 대열 합류를 알렸다. 이는 자율주행 기술이 자동차를 단순 이동수단에서 이동성을 확보한 생활공간으로 자동차의 근본적인 개념을 변화시켜 새로운 산업 패러다임적 가치뿐 아니라 사회, 경제적 변혁을 예고하고 있기 때문일 것이다.

이러한 배경에 힘입어 자율주행 시장은 보다 더 편리하고 안전한 차량을 추구하는 인간의 지속적인 기대와 함께 고령화 시대가 가속화되면서 더욱 더 커지고 있다. 최근 자동차업계에서 자율주행이 주요 트렌드로 자리매김한 배경이며 이러한 트렌드를 따라 모바일 로봇을 통해 자율주행을 구현하여 모바일 로봇의 지능제어와 최신 자율주행 트렌드를 적용해보고자 한다.

이러한 배경을 통해 모바일로봇프로그래밍 강의에서 제시하는 모바일로봇의 경로계획, 센서 정보처리, 위치인식 등의 모바일 로봇의 지능제어를 적용하고 응용하고자 한다.

2. 과제 목표

모바일 로봇이 경로계획, 센서 정보처리 및 위치 인식을 통해 자율주행을 하도록 고속 도로 위의 상황을 시뮬레이션 한다.

- 카메라 센서를 이용하여 차선을 인식한 후 자율 주행하도록 한다.



그림 1. 차선을 벗어나지 않는 경우



그림 2. 차선을 벗어날 경우

- 카메라 센서를 이용하여 화살표가 그려져 있는 표지판을 인식한 후 상황에 따른 행동을 제시한다.
- LIDAR 센서를 이용하여 전방에 있는 물체를 인식한 후 상황에 따른 행동을 제시한다.
- LIDAR 센서를 이용하여 요금 정산소를 인식한 후 상황에 따른 행동을 제시한다.



그림 3. 요금 정산소 (하이패스)

- 도로의 차선을 제시하고 해당 차선을 인식하여 모바일 로봇이 자동으로 주행하도록 하게한다.
- 모바일 로봇은 양쪽 차선을 모두 인식하여 주행해야하며 차선을 벗어났을 경우 자동으로 차선을 찾아 주행하게 해야 한다.

3. 과제 수행 결과

- 자율 주행

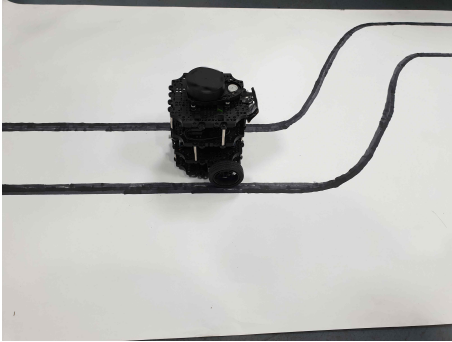


그림 4. 직진 상황 Robot

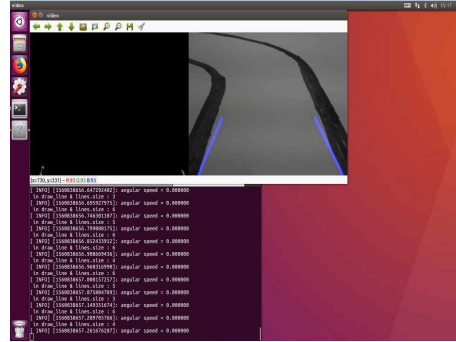


그림 5. 직진일 때 Remote PC

로봇이 두 차선을 모두 인식하고, 해당 차선의 중앙값이 설정한 범위를 벗어나지 않으면 직진을 한다. 하지만 설정한 범위를 벗어날 경우, 차선의 중앙보다 왼쪽으로 벗어났으면 우회전을 하고, 오른쪽으로 벗어났다면 좌회전을 한다. 따라서 차선의 중앙을 맞추어 로봇이 주행하도록 할 수 있다.



그림 6. 커브 상황 Robot

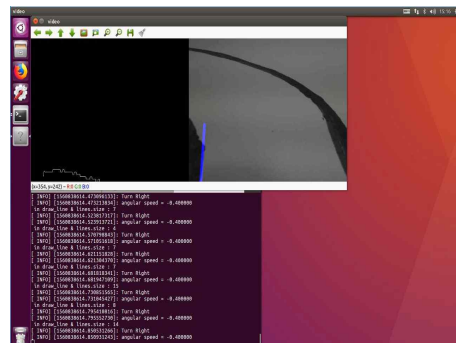


그림 7. 커브일 때 Remote PC

보통 커브일 경우 차선을 하나밖에 인식하지 못한다. 따라서 차선 하나만으로 회전을 하며 주행해야 하므로, 인식된 차선의 기울기에 따라 회전하는 정도를 설정하여 주행하도록 한다. 또한 카메라 환경에 따라 차선이 있음에도 불구하고, 차선을 인식하지 못하는 경우가 있다. 이는 Single Camera에 따른 제약사항에 해당되는데, 그럴 경우에는 인식하지 못하기 전의 주행속도와 회전 값으로 계속 주행하도록 하여 차선을 발견하는 즉시 다시 자율주행 하도록 한다.

차선의 폭이 넓어 Camera Frame안에 도로가 나타나지 않으면 두 차선을 적절하게 인식을 하지 못하지만 로봇이 차선에 맞춰서 주행하는 모습을 보여주기 위하여 이렇게 차선을 설정하였다. 또한 Single Camera로는 전방 차선과 현재 로봇이 지나가고 있는 차선을 동시에 인식하는 것을 구현하는데 있어서 어려움이 따른다. 이는 추후에 Dual Camera를 이용한다면 해결될 것으로 생각된다.

- 화살표 표지판 인식

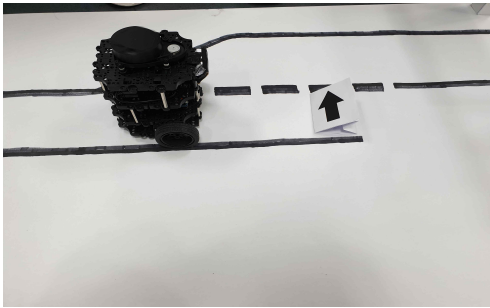


그림 8. 화살표 왼쪽(좌회전)

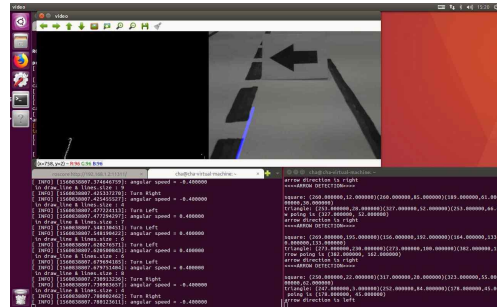


그림 9. 좌회전 Remote PC

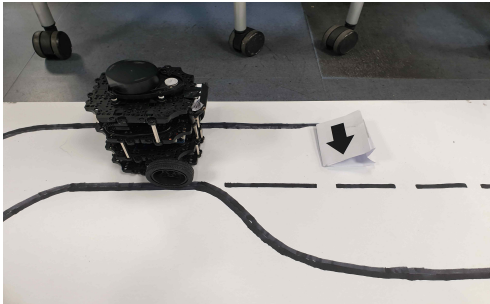


그림 10. 화살표 오른쪽(우회전)

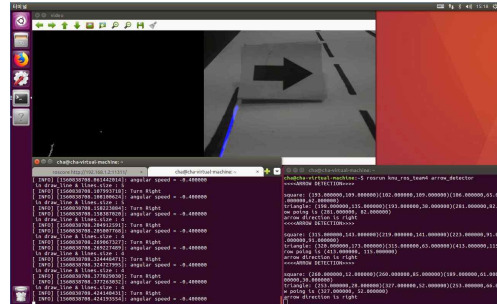


그림 11. 우회전 Remote PC

고속도로를 주행하다 표지판을 봤을 경우를 시뮬레이션 한다. openCV를 통해 카메라로 받아온 영상을 분석하여 외곽선을 가진 다각형을 받아온다. 그 후 꼭짓점이 7개인 도형만을 받아와 각 꼭짓점의 각도를 계산한다. 화살표에서 사각형인 부분과 삼각형인 부분을 나누어 판단한다. 사각형인 부분은 90도로 계산되는 두 선분이 만나는 점을, 삼각형인 부분은 60도로 계산되는 두 선분이 만나는 점을 지정한다. 화살표가 인지되면 삼각형 부분에서 가장 오른쪽에 있거나 왼쪽에 있는 점을 선택한 후 사각형의 점들 중 임의의 점 하나와 비교하여 삼각형의 꼭짓점이 더 크다면 오른쪽, 작다면 왼쪽으로 인식한다. 이렇게 인식한 방향은 메시지를 퍼블리시 하여 자율주행 노드에 전달한다.

따라서, 화살표 방향을 인식 후 그에 따른 로봇의 행동을 제시한다. 좌회전이나 우회전 표지판을 인식할 경우, 로봇에게 현재 속도에 따른 적절한 회전 값을 부여하고, 잠시 차선 인식을 멈춘다. 그리고 새로운 차선이 인식되면 다시 차선 인식을 이용해 자율 주행으로 동작하도록 한다.

- 전방 물체 인식

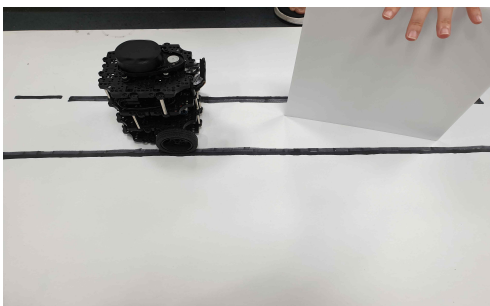


그림 12. 전방 장애물 인식

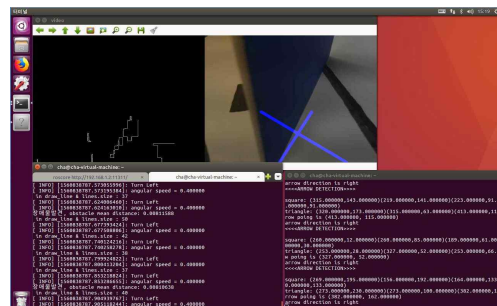


그림 13. 장애물 인식 Remote PC

고속도로를 주행하다 전방에 장애물이 나타날 경우에 대한 상황을 시뮬레이션 한다. 장애물

인식을 위해 물체 스캔에 대한 수식은 아래와 같다. 물체와 로봇 사이의 거리는 유클리드 거리 계산에 따른다.

$$dAngle = \tan^{-1}\left(\frac{laserY - robotY}{laserX - robotX}\right)$$

수식 1. 물체와 로봇이 이루는 각도

$$distance = \sqrt{(laserX - robotX)^2 + (laserY - robotY)^2}$$

수식 2. 물체와 로봇 사이의 거리

따라서, 로봇이 바라보는 방향에서 (-20도 ~ 20도) 사이의 장애물이 인식되고 그 장애물까지의 거리가 20cm 이하일 경우 로봇의 움직임을 멈춘다. 그리고 장애물이 인식되지 않으면 다시 로봇이 자율 주행하도록 한다.

- 요금 정산소 인식

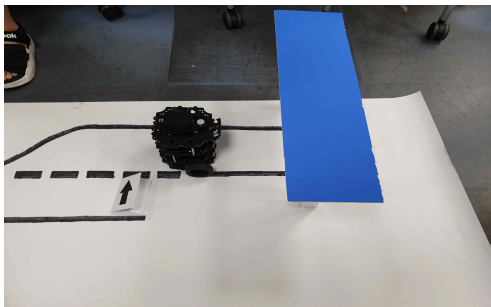


그림 14. 요금 정산소 인식

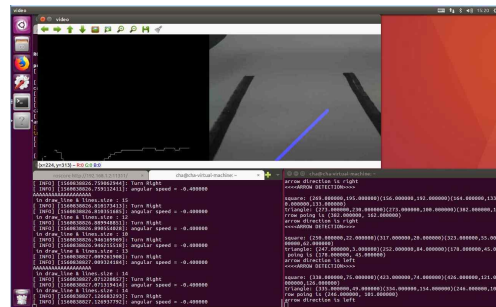


그림 15. 요금소 인식 Remote PC

고속도로에서 나가기 위해서는 요금 정산소(하이패스)를 꼭 지나야 한다. 위 수식 1. 와 수식 2. 를 활용하여 로봇이 바라보는 방향을 기준으로 오른쪽, 왼쪽에 물체가 존재가 인식되고 거리가 35cm 내 이면 요금 정산소로 인식한다.

그 후, 정산소를 지나가기 까지 로봇의 속도를 낮추어 천천히 주행하도록 한다.

4. 기대 효과 및 활용 방안

1. 장애물과의 충돌 상황 대처

우리는 로봇 앞에 어떤 장애물이 나타날 경우 그 장애물까지의 거리를 스스로 판단하여 로봇의 주행속도를 줄일 수 있게 하였다. 고속도로에서는 자연재해로 인해 장애물이 생기거나 앞에서 졸음운전을 하는 차량과 거리가 가까워지는 등 돌발 상황이 많은데, 이러한 경우 로봇 스스로 판단하여 속도를 줄일 수 있게 된다.

2. 차선인식을 이용한 주행 자동화

차량은 항상 차선 중앙에 맞춰서 주행되어야 한다. 우리는 차선의 중앙값을 계산하여 로봇이 실시간으로 차선의 중앙을 찾아갈 수 있게 하였다. 이는 주행에 있어서 매우 중요한 요소이고, 이는 추후 자율주행 자동차 개발에서 기본적인 알고리즘으로 이용될 수 있을 것이다.

3. 특정 장애물 인식과 상황에 따른 대처

우리는 표지판이나 요금 정산소와 같은 특정 장애물을 인식하여 그에 따른 대처를 할 수 있게 하였다. 표지판의 화살표의 방향에 따라 차선을 변경할 수 있는데, 이는 추후 도로 표지판에 따라 로봇이 어디로 가야하는지 스스로 판단할 수 있는 네비게이션 기능의 발판이 될 수 있다. 또한 고속도로에서 요금 정산소를 인식 후 차량의 속도를 줄일 수 있게 하였는데, 이는 추후 하이패스 기능에 이용할 수 있을 것이다.

5. 참여인력(세부)

과제 참여 학생	소속(학과)	학위과정 (성별)	학번	성명	담당업무
	컴퓨터학부	학사과정 (남)	2016113705	안준영	차선 인식 및 자율주행 구현
		학사과정 (남)	2016111829	우정우	장애물 인식 및 행동 구현
		학사과정 (남)	2014105075	이지원	도로 위 특정 상황 인식 및 행동 구현
		학사과정 (남)	2014105081	전우혁	차선 인식 및 자율주행 구현
		학사과정 (여)	2016111813	차수진	기호 인식 및 행동 구현