

# 국민대학교 FOSCAR 개발계획서

이태훈, 이주형, 배한율, 박지호, 홍성표

## 목차

<b>1. 개요</b>	<b>3</b>
1.1 작품개요	3
1.2 필요성	3
1.3 개발목표	3
<b>2. 운행부분 개발계획</b>	<b>4</b>
2.1 주차알고리즘	4
2.2 주차공간판단	4
2.3 수평주차	4
2.4 수직주차	5
2.5 회전교차로	5
2.6 회전교차로에서의 장애물 감지	6
<b>3. 차선처리 개발계획</b>	<b>6</b>
3.1 차선인식 알고리즘	6
3.2 ROI 설정	7
3.3 이진화	8
3.4 Edge 검출	8
3.5 차선 후보군 검출	9
3.6 대표차선	9
3.7 곡선주행	10

<b>4. 그 외 영상처리 개발계획</b>	10
4.1 신호등	10
4.2 우선정지장애물	11
<b>5. 터널구간 계획개발</b>	11
5.1 터널구간 알고리즘 설계도	11
5.2 터널구간 진입 판단	12
<b>6. 추월차로 개발계획</b>	12
6.1 추월차로 인식	12
6.2 추월차선 ROI 값	13
<b>7. 예상되어지는 장애요인 및 해결방안</b>	13
<b>8. 기간별 개발계획</b>	15
<b>9. 팀원소개 및 담당업무</b>	15

# 1. 개요

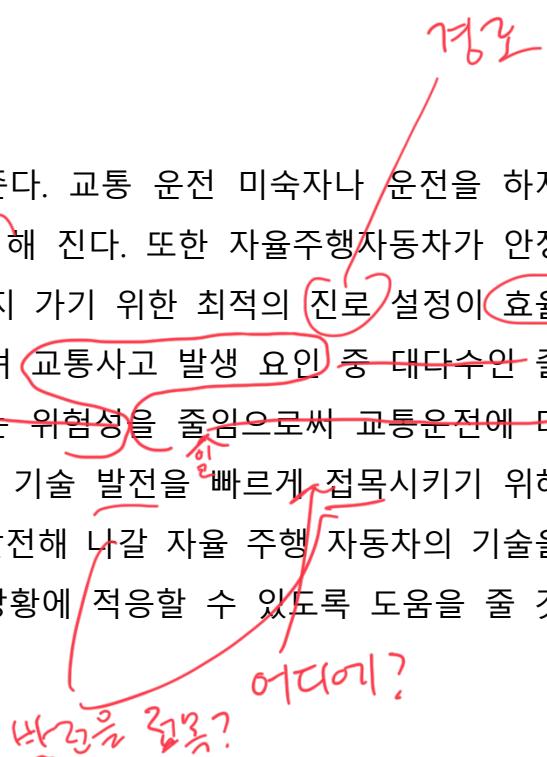
## 1.1 작품 개요

이 작품은 자율 주행 자동차가 영상처리를 바탕으로 주행하는 도중 발생하는 다양한 상황을 설계된 알고리즘을 통해 해결해 나간다. 차선 선택과 변경, 수직 및 평행 주차, 장애물 인식, 신호등과 표지판 인식을 원활하게 수행 할 수 있도록 소프트웨어를 개발한다. 이는, 실제 도로에서 발생할 수 있는 문제들에 접목되어 문제들을 판단하고 대처할 수 있다.



## 1.2 필요성

자율주행자동차는 운전에 대한 접근성을 용이하게 해준다. 교통 운전 미숙자나 운전을 하지 못하는 운전 미숙자들에게도 운전에 대한 접근이 수월 해 진다. 또한 자율주행자동차가 안정화 및 상용화가 되면 도로 상황에 대한 정보, 목적지까지 가기 위한 최적의 진로 설정이 효율적으로 가능하기 때문에 교통정체도 많이 감소하게 되며 교통사고 발생 요인 중 대다수인 출음운전, 음주운전, 과속, 신호위반 등으로 발생하게 되는 위험성을 줄임으로써 교통운전에 대한 안전성을 확보할 수 있다. 따라서 자율주행자동차의 기술 발전을 빠르게 접목시키기 위해 트랙에서의 시뮬레이션을 통해 앞으로 더욱더 빠르게 발전해 나갈 자율 주행 자동차의 기술을 재현 가능하며, 실제 도로에서의 차량이 여러 환경과 상황에 적응할 수 있도록 도움을 줄 것으로 기대된다.



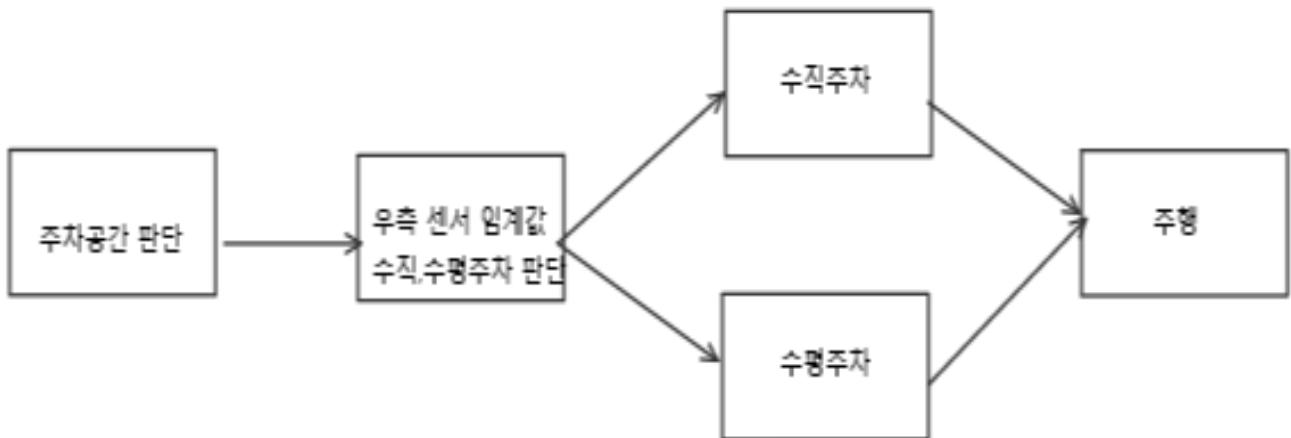
## 1.3 개발목표

자율 주행 모형자동차 부분은 실제 자동차가 주행하는 것과 비슷하게 모형차로 시뮬레이션 하는 것이 목표이다. 실제 자동차에 적용될 수 있도록 실시간성을 갖추어야 하기에 모든 연산은 간결화 되어야 한다. 또한 어느 환경에서도 동일한 주행을 해야 하기에 환경 변화에 따라 다른 결과가 나오지 않도록 소프트웨어를 구성하여 항상 동일한 주행을 할 수 있는 능력을 갖춘 자동차를 만들어야 한다.

SW?

## 2. 운행부분 개발계획

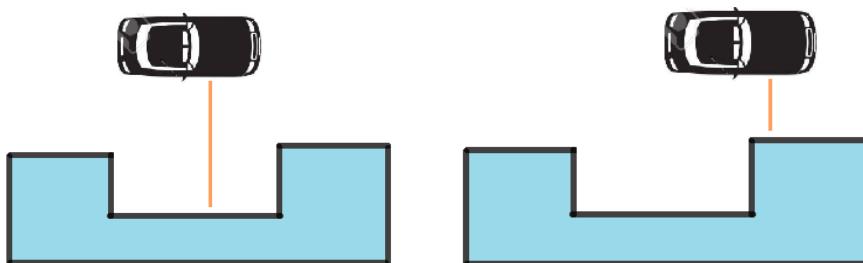
### 2.1 주차 알고리즘



### 2.2 주차공간판단

주행 중 주차 공간이 들어오는 방향을 판단하기 위해 양 옆 센서 값을 모두 받아들여 일정 구간 이상의 센서 값의 차이가 발생하면 주차공간으로 인식한다. 예를 들어 장애물까지의 거리가 길어지기 시작하는 구간부터 다시 짧아질 때까지의 거리를 측정하여 이 거리가 일정값 보다 작으면 수직주차로, 그렇지 않으면 수평주차로 인식하여 주차 공간을 판단한다. 하지만 값이 너무 크거나 작으면 값을 채택하지 않고 최대 임계값과 최소 임계값을 설정하여 주차 공간을 판단한다.

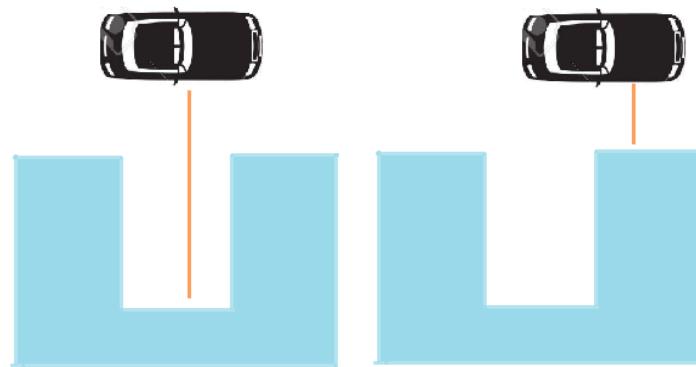
### 2.3 수평주차



#### 수평주차 공간 탐색

주차 공간이 오른쪽이라 판단되면 오른쪽 센서를 통해 주차공간의 너비와 폭을 측정하고, 공간의 끝을 탐색하면 정지한다. 수평주차는 우향 스트리어링 값을 결정하여 장애물까지의 거리가 기준으로 정해놓은 threshold가 될 때까지 후진하고, 다시 좌향 스트리어링 값을 결정하여 1차 장애물까지의 거리가 threshold가 될 때까지 후진하여 수평주차를 완료한다.

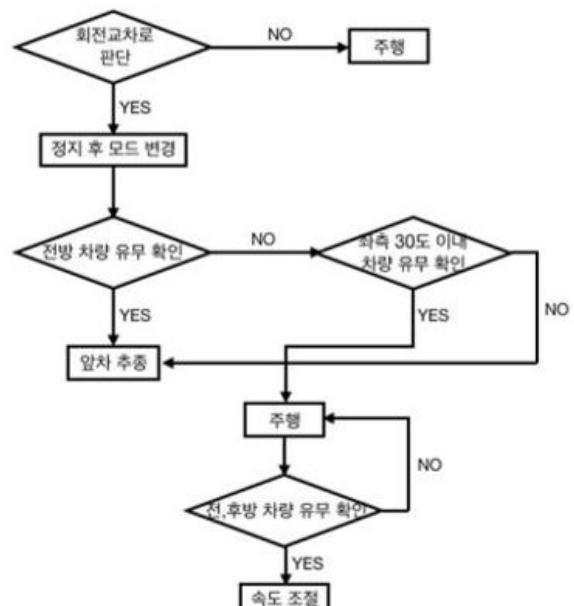
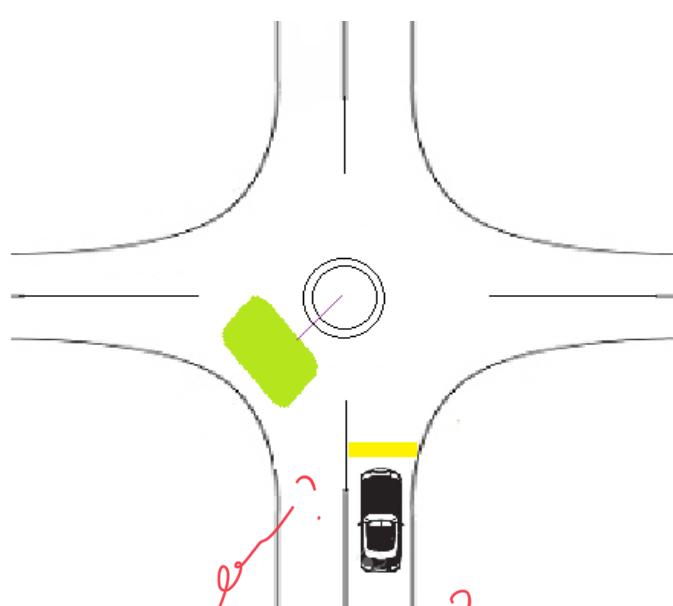
## 2.4 수직주차



### 수직주차 공간 탐색

주행 중 주차 공간이 들어오는 방향을 판단하기 위해 양 옆 센서 값을 모두 받아 들여 일정 구간 이상의 센서 값의 차이가 발생하면 주차 공간이라고 인식한다. 수직주차는 2차장애물의 인식이 끝나는 점과 주차가 완료된 시점의 차를 기준으로 원을 그려, 우향 스트어링 값을 결정하고 벽과의 거리가 threshold가 될 때까지 후진한다. 후진 중에 센서에 들어오는 값들을 비교하여 수직으로 들어가고 있는지를 판별하고 벽에 충돌 할 만큼 가깝다고 판단되면 그 차이 만큼 앞으로 전진해 보정한다.

## 2.5. 회전교차로 알고리즘



회전교차로 판단 진입 판단: 라인 감지용 센서가 자동차 아래 흰색 라인을 인식하면 정지하고 회전 교차로로 진입으로 판단한다.

## 2.6 회전 교차로에서의 장애물 감지

자동차 앞과 뒤에 있는 센서를 이용하여 장애물이 일정한 거리에 인식되면 차량을 출발시킨다.  
회전교차로에서는 주행 중인 다른 차량을 감지할 때는 그림자를 이용한다. 해석!

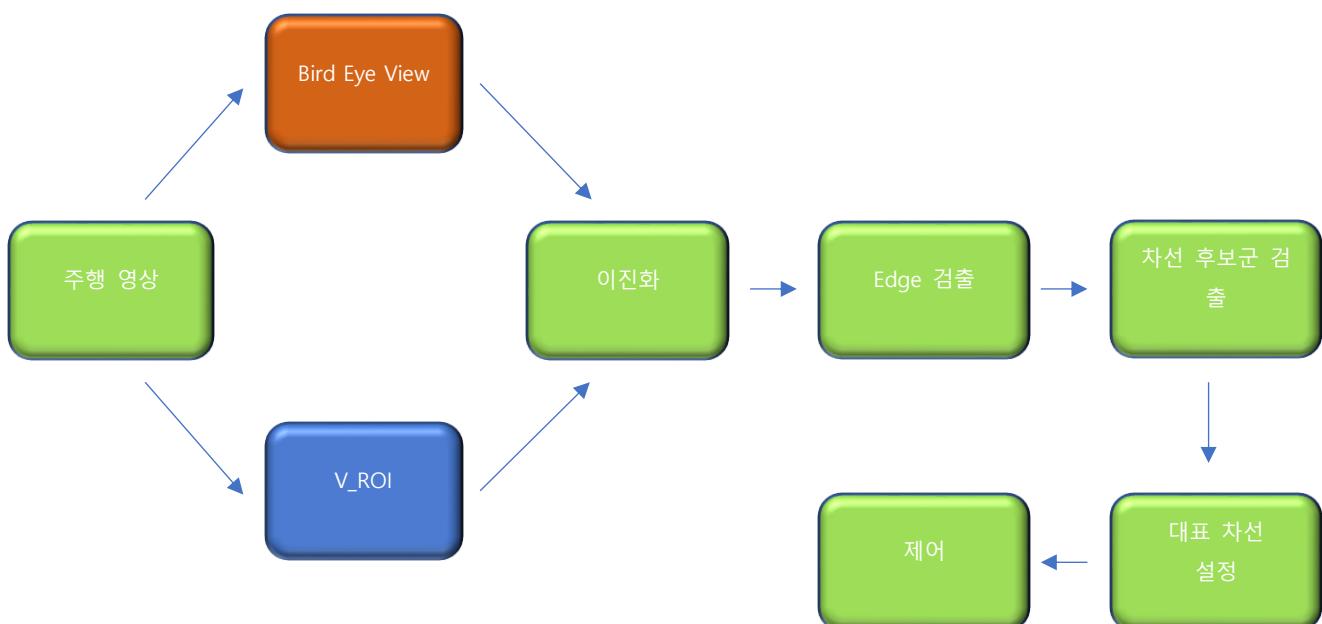
1) 장애물 판단 : YUV모델을 이용하여 ROI 내 각 픽셀의 밝기 값을 추출하고 ROI 영역과 다른 영역과의 Y값 차이가 기준값 이상이면 그 영역에 장애물이 있는 것으로 판단하고 또한 계속해서 거리센서를 이용하여 장애물이 얼마나 가까운지를 판단한다.

2) 충돌 방지 : 차량이 앞쪽 장애물을 감지하고 그 거리가 사용자가 설정한 기준값 보다 가까우면 속도를 줄인다. 후방에서 오는 장애물은 센서를 이용하여 장애물이 뒤쪽에서 가깝게 접근하면 속도를 높여 장애물과 충돌을 방지한다. 동시에 계속해서 주행라인을 감지하여 주행을 계속한다.

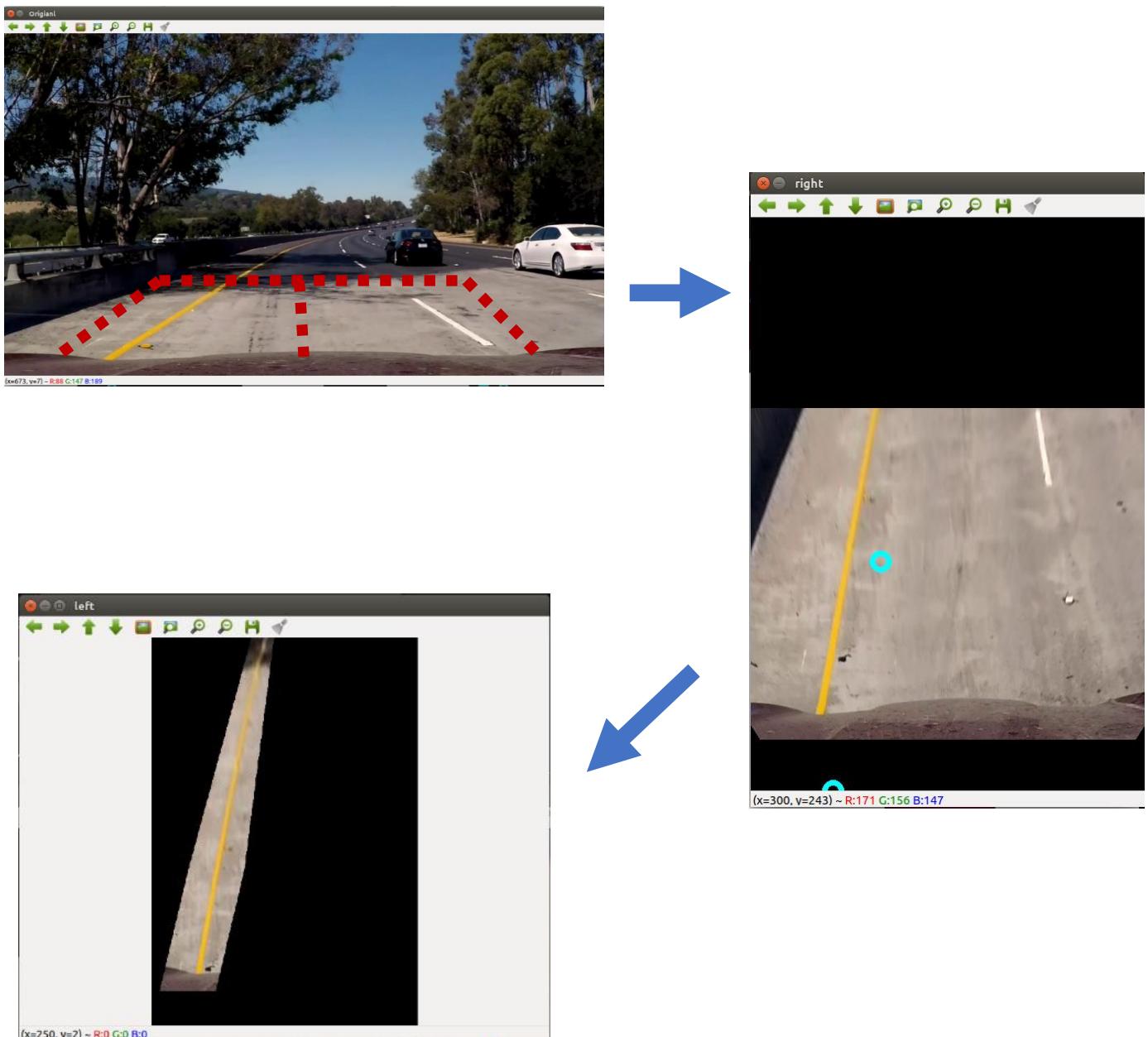
3) 회전교차로 탈출 : 일정 시간 동안 거리 센서에 장애물이 인식되지 않고 더불어 그림자를 통한 장애물 인식이 되어지지 않을 경우 회전교차로를 탈출했다고 판단하고 주행라인 감지하여 다음 미션을 수행한다. 인식되지 는

## 3. 차선 처리 개발계획

### 3.1 차선 인식 알고리즘 설계도



### 3.2) ROI 설정 :



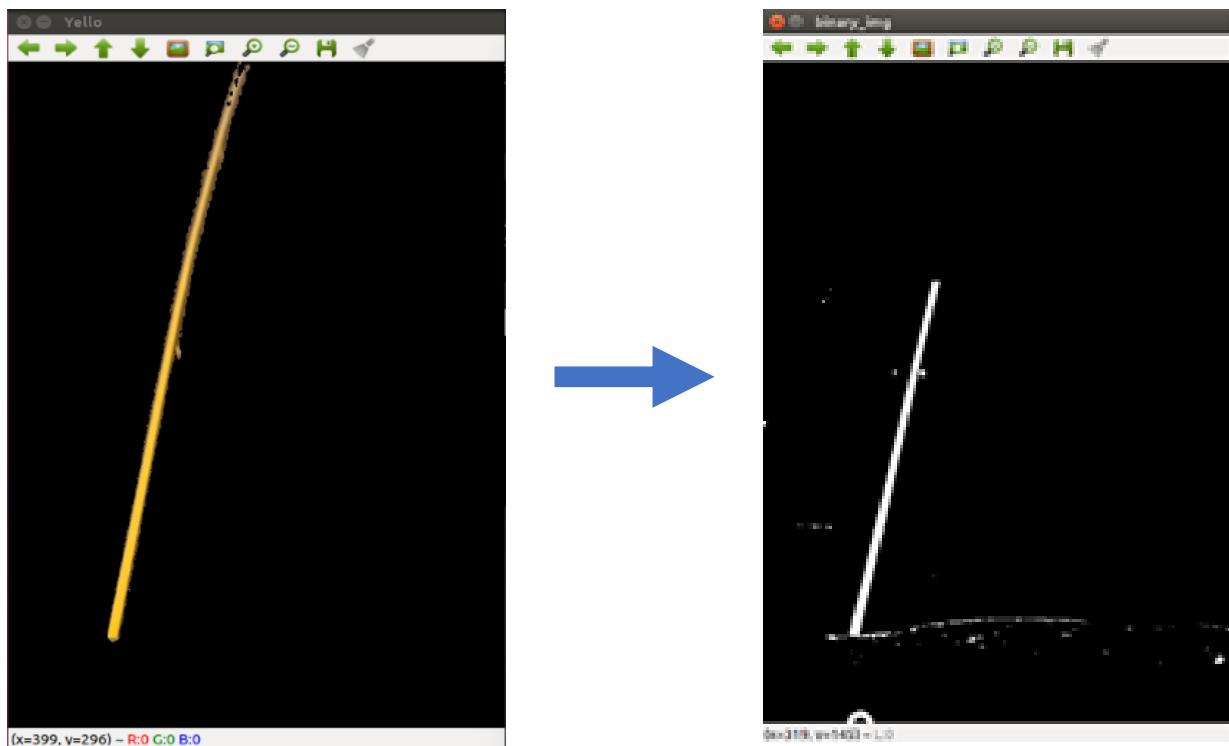
기존 화면에서는 두 차선이 평행하지 않게 영상 처리가 되어 교차하는 지점이 생긴다. 그래서 다음과 같이 Bird Eye's View를 이용해 두 차선을 평행하게 처리하여 효과적으로 차선 인식을 설정한다. 인식이 되면, 인식된 차선을 기준으로 다시 ROI를 지정하여 차선만 연산할 수 있게(V\_ROI) 연산 속도를 향상시킨다.

도록

뒤에 나오는 부분과 표기 통일

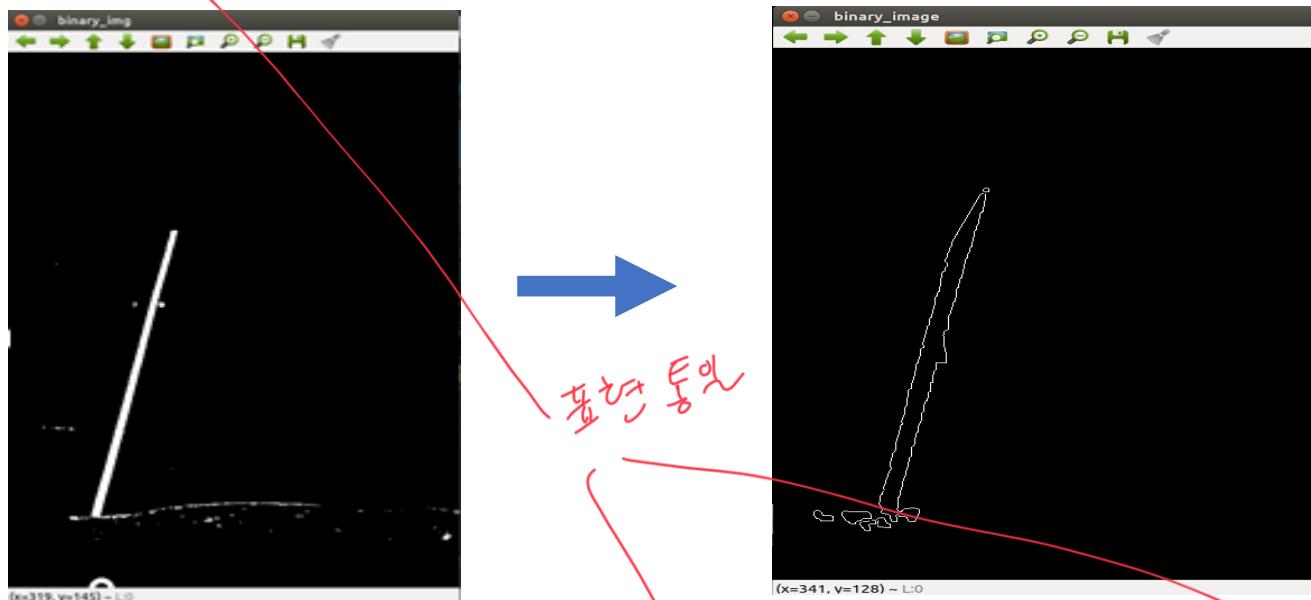
15

### 3.3) 이진화



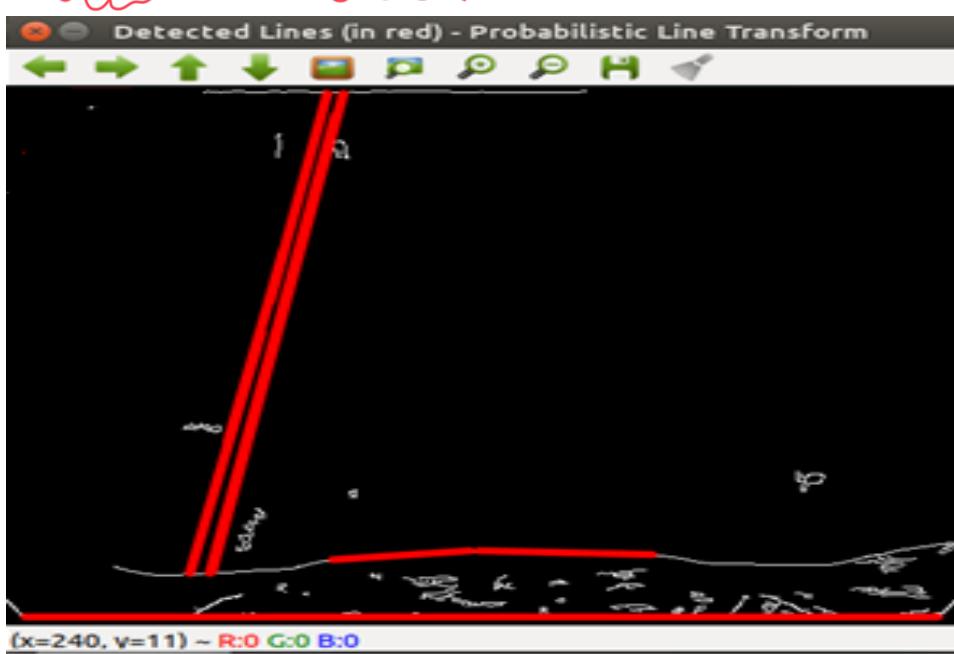
인식된 차선이 노란색이라는 것을 판단하기 위해 HSV 색공간 변환을 이용해 노란색 필터를 적용한다. 필터가 적용된 이미지를 이진화하면 보다 효과적으로 차선을 이진화할 수 있다.

### 3.4) Edge 검출



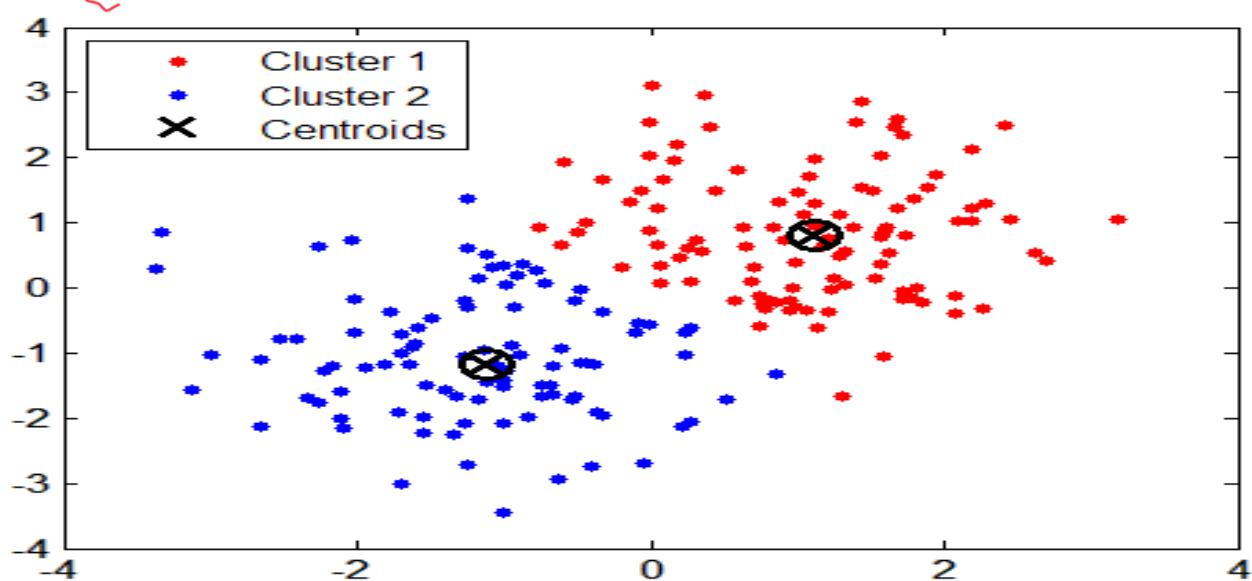
Canny Edge 검출 알고리즘의 최솟값과 최댓값 변수를 적절히 조정하여 가장 효과적으로 엣지 를 뽑아내는 것을 선택해 이진화 된 영상에서 엣지만 뽑아낸다.

### 3.5) 차선 후보군 검출 :



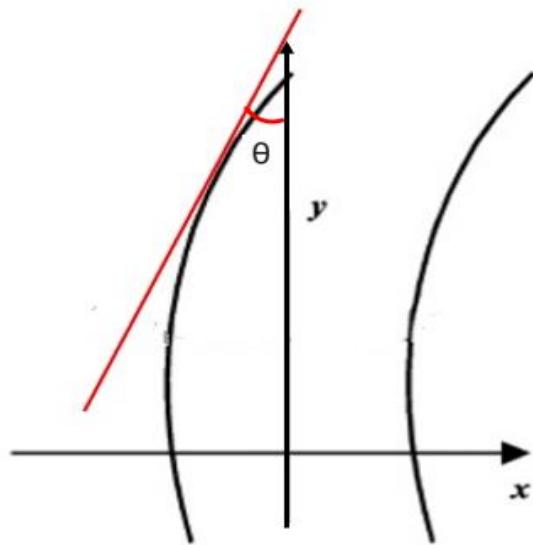
Edge 검출이 완료되면, Hough Transform 알고리즘을 통해 Edge들을 연결한 직선들의 집합, 즉 차선의 후보군들을 검출한다.

### 3.6) 대표 차선 설정 :



Hough Transform 알고리즘으로 도출해낸 차선의 후보군에서 K-means 알고리즘을 통해 후보군을 군집화 한 후, 그 영역의 중심에서 최적의 차선을 구한다.

### 3.7) 곡선 주행 알고리즘

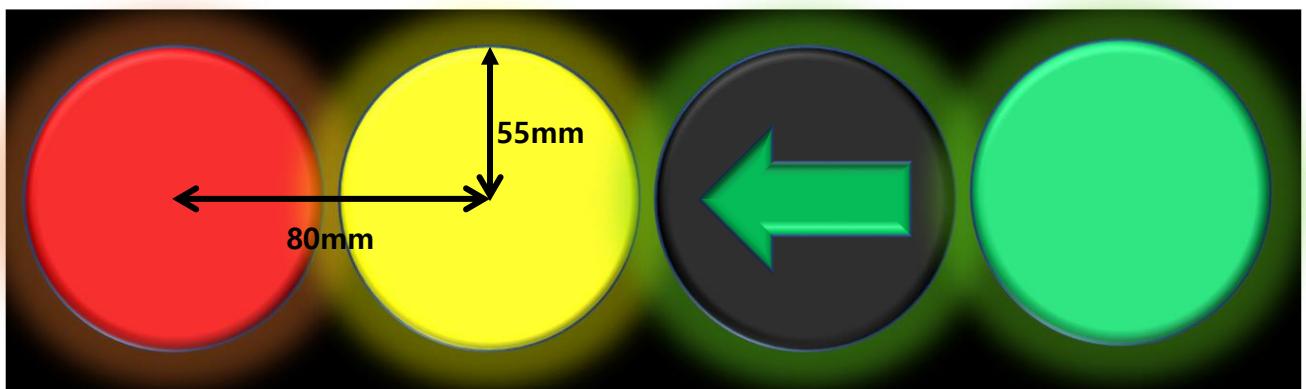


차량을 기준으로 주행중인 방향에 대한 직선을 설정하여 현재 인식중인 차선과의 각도를 구해 해당 값이 임계값 이상으로 커지면 곡선도로에 진입하고 있음을 인식하고, 차체의 속도를 각도에 따라 조절하여 주행한다.

## 4. 그 외 영상처리

### 4.1) 신호등

정지선을 인식하게 ROI를 설정해, 인식된 화면을 이진화해 흰색 픽셀 수를 계산한다. 계산된 픽셀의 값이 미리 설정한 임계값과 비교해, 그 값이 클 경우 정지선으로 판단하고 정지시킨다.

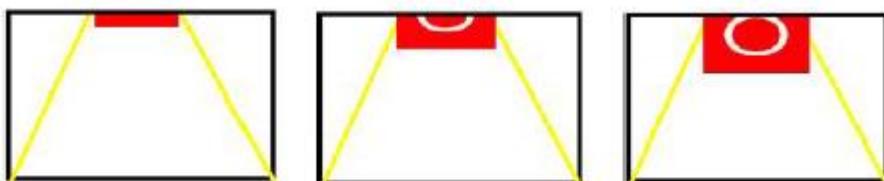


다음과 같이 신호등은 4개의 원이 일정한 거리를 유지하며 위치해 있다. 앞서 판단한 정지선

에서 잡음을 제거한 ROI를 통해 신호등을 검출한다. 검출된 각 원들의 좌표를 저장한다. 왼쪽 두 개의 원의 픽셀이 RED, YELLOW로 검출될 경우 정지를 유지하다가, 우측 두 개의 원에서 녹색 픽셀이 검출되는 순간 출발한다. 단, 만약 우측 두 개 중 왼쪽 원에서 검출될 경우 좌회전을 하고 우측 원에서 검출될 경우 우회전을 한다.

YUV 변환을 모든 픽셀에서 수행한다면 연산 속도가 느려지기 때문에, 아까 미리 저장한 4개의 원에서만 YUV 변환을 실시하고, RED, YELLOW, GREEN의 픽셀만을 이용하기 때문에 해당 색상들의 YUV값과 미리 설정한 임계값으로 판단을 한다.

## 4.2) 우선 정지 장애물



주행도중 적색 픽셀 분포를 확인하여 적색 픽셀 분포가 급증하면 감속 주행을 하고, 적색 픽셀이 임계값 이상으로 감지되면 돌발 장애물 표지판임을 인지하고 완전히 정지한다.

## 5. 터널구간 개발 계획

### 5.1) 터널 구간 알고리즘 설계도



#### 1) 터널 구간 진입

터널 구간 진입 시에는 밝기의 변화가 있다. 순간적으로 밝기의 변화가 어두워질 경우 터널 구간 진입 여부를 알 수 있다. 터널 구간 진입을 한 경우에는 헤드라이트를 켜고 터널 모드로 변경한다. 터널모드는 주 센서로 적외선을, 보조 센서로 카메라를 이용하여 주행한다.

## 2) 터널 주행

터널 내부는 어두워 차선을 인식하기에 까다롭다. 평상시에 사용했던 이진화 값을 통한 차선 인식은 어렵다. 빛이 별로 없을 때 사용하는 이진화 값을 차선을 인식하여 주행한다. 적외선 센서를 통해 좌, 우 비율을 1:6 등과 같이 설정하여 일정하게 유지되도록 주행을 한다.

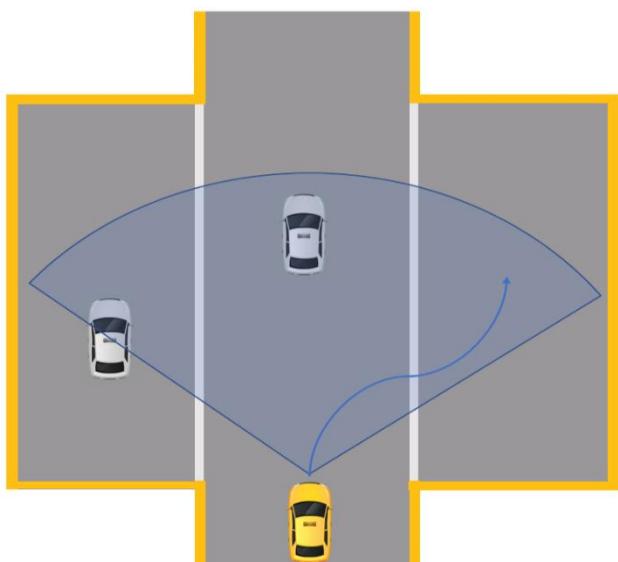
## 3) 터널 구간 종료

터널 구간을 종료 시에 밝기의 변화가 발생한다. 현재 터널 구간 주행 중이고 순간적으로 밝기의 변화가 밝아질 때, 터널 구간 종료를 알 수 있다. 터널 구간 종료 후에는 주 센서를 적외선 센서가 아닌 카메라를 기반으로 영상처리에 임한다.

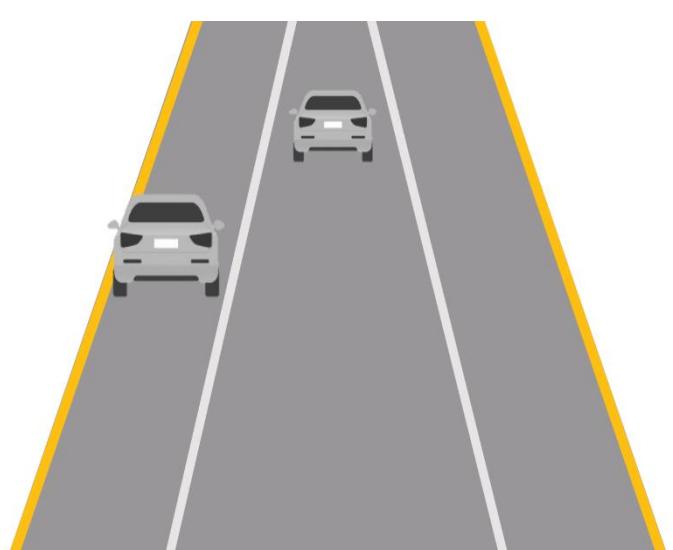
## 5.2) 터널 구간 진입 판별



## 6. 추월차로 개발계획



〈그림1〉



〈그림2〉

## 6.1) 추월구간 인식

추월 구간에 진입하면 총 3개의 차선이 있다. 3개의 차선 중 2개의 차선에만 장애물이 존재한다. 따라서 장애물이 없는 차선으로 추월을 해서 추월 구간을 빠져나가는 것이 해결 과제이다. 추월 차선에 진입 후에는 카메라에서 영상을 받아온다(그림 1). 그 다음 노란색 차선을 가져온다. 장애물이 있는 차선의 경우에는 아래 그림과 같이 노란 차선이 중간에 끊어진 것을 볼 수 있다. 반면에 장애물이 없는 차선의 경우(그림 2), 노란 차선이 중간에 끊김이 없이 연결되는 것을 볼 수 있다. 만약 중앙 차선이 장애물이 없을 경우에는 2개의 노란 차선이 중간에 끊김을 알 수 있다. (또는 전면 적외선 센서에서 장애물을 판단하지 못 할 것이다.) 차선을 변경할 때에는 모터의 회전에 따른 바퀴 회전 수를 구하고 그에 맞게 차선을 변경한다. 양쪽 라인으로 이동할 경우 좌, 우 적외선 센서를 사용하여 장애물을 통과한 것을 인식, 장애물을 통과한 뒤에는 중앙 차선으로 이동하여 추월 구간을 빠져나간다.

예보를

비활성화 시킨다.

## 6.2) 추월차선 ROI 값

차선 추적을 위한

추월차로에서 2대/로

차선을 따라가기 위해서 처리하였던 Bird Eye View 알고리즘을 사용하게 될 경우에는 시야각이 좁아 장애물을 판단하기 어렵다. 위 사진(그림 2)과 같이 장애물을 판단하기 위해서는 시야가 넓어져야 하므로 카메라 시야를 조절해야 한다. 따라서 Bird Eye View를 잠시 OFF한다. 그 후에 영상처리를 통해 노란 차선이 끊기는 부분을 인식, 장애물이 없는 차로를 변수로 저장한다. 다음에는 라인 트래킹을 위해 다시 Bird Eye View를 사용하여 다시 차선을 인식하여 차로를 따라간다.

활용하세요

다음

## 7. 예상되는 장애요인과 해결방안

발생 구간	장애요인	해결방안
도로 주행	<ol style="list-style-type: none"><li><b>내리막길</b> : 카메라가 시야 확보가 안돼 정상적인 차선인식이 제한되고, 속도가 올라가 벽에 부딪힐 가능성 존재한다.</li><li><b>고가도로</b> : 가드레일 그림자에 의해 차선이 보이지 않을 경우 정상적인 차선인식이 제한된다.</li><li><b>곡선차로</b> : 급격하게 차선이 휘는 곡선 차로일 경우 속도가 빨라 차량이 차선을 이탈할 가능성이 있다.</li></ol>	<ol style="list-style-type: none"><li>전면 적외선 센서로 수집한 값이 일정 수준 이하로 감소하면 내리막길로 판단 -&gt; 속도를 대폭 줄여 주행한다.</li><li>모풀로지 연산을 통해 노이즈 제거를 실시 -&gt; 차선을 안정적으로 인식할 수 있게 한다.</li><li>곡선 차로 판단 시, 기존 속도의</li></ol>

		절반으로 줄이고 서서히 늘려가는 방식을 취해 안정적으로 주행한다.
신호등 인식	1. 주변 빛과 그림자로 신호등을 정확하게 인식하지 못해 어떤 동작을 선택할지 판단하지 못할 가능성이 있다.	1. 모풀로지 연산을 통해 노이즈 제거를 실시 -> 신호등을 정확하게 인식할 수 있게 한다.
회전 교차로	1. 장애물이 좌측 30도 일 때 진입을 실시하는데 진입이 늦으면 장애물과 부딪힐 가능성이 있다. 2. 회전교차로 진입 이후 도로주행시 속도가 느리면 장애물이 뒤에서 부딪힐 가능성이 존재한다.	1. 장애물 인식 범위를 넓혀 오차범위를 설정해 해당 범위에 들어오면 진입을 실시한다. 2. 도로 주행시 후면 센서를 통해 지속적으로 장애물의 유무를 인식하고 인식 거리에 따른 속도 조절을 실시한다.
터널 구간	1. 도로가 검정색일 경우 밝기 변화를 인지하는데 어려움이 있을 수 있다. 2. 적외선 센서를 이용하여 Left, Right 거리를 PID 제어를 통해 진행할 경우 차선을 바르게 못 갈 수 있다. 3. 터널 내부 곡선 주행이 있을 경우 Left, Right 거리를 통해 차선 유지에 어려움이 발생할 수 있다.	1. 평시 주행 값을 통해 통계를 내어 차이를 비교한다. 2. 차량의 Width와 도로 폭을 변수로 지정하여 이를 맞게 계산하여 차선을 지키며 주행한다. 3. 전면 적외선 센서를 통해 곡선 주행을 판단, Right를 유지하는 것으로 차량을 유지한다. 또는 헤드라이트를 켜고 영상 처리를 통해 Lane Keeping Assistant를 실행한다.
주차 구간	1. 도로 주행시에 벽에 가까워지면 센서값이 작아져서 주차공간으로 판단할 수 있다. 2. 주차 구간에서 센서값이 매우 크게 튀어 제대로 인식하지 못하는 상황이 발생할 수 있다.	1. 센서값이 너무 작으면 해당 센서값은 채택하지 않는다. 2. 센서값의 오차 범위를 설정해 그 이상으로 넘어가는 센서값은 채택하지 않는다.
추월 차로	1. 중앙에 있는 장애물 외에 양 도로에 있는 장애물의 판단을 노란 차선의 소실로 판단하기 때문에 정확하게 차량이 중앙으로 가지 않으면 정상적으로 판단하지 못한다.	1. 최초 진입시 점선을 차선으로 인식하게 하여 중앙 차선 안을 직선으로 주행하게 한다.

~tant?  
~tance?  
적인것은

알고니증을

## ~~8. 기간별 개발 계획~~

제작 일정

단계	세부사항	6월	7월	8월	9월	10월	11월
준비	계획/개발환경 구축						
설계	영상처리						
	센서 & 제어						
구현	주차 알고리즘						
	회전교차로 알고리즘						
	곡선(S) 주행 알고리즘						
	신호등 분기점 알고리즘						
	터널 구간 알고리즘						
	추월 차로 알고리즘						
	내리막 구간 알고리즘						
	우선정지 장애물 알고리즘						
	도로 주행 알고리즘						
통합	전체 알고리즘 및 통합						
시험	통합 테스트 및 디버깅						

## ~~9. 팀원소개 및 담당업무~~

역할분담

성명	소속	부서	담당업무
이태훈*	국민대학교	컴퓨터공학	진행상황 점검 및 알고리즘 통합 ROI 설정, 대표차선 결정 신호등 알고리즘
이주형	국민대학교	컴퓨터공학	곡선 주행 알고리즘 돌발 장애물 알고리즘
박지호	국민대학교	컴퓨터공학	터널 알고리즘 추월차로 알고리즘
홍성표	국민대학교	소프트웨어전공	주차 알고리즘 구현 개발 환경 구축
배한울	국민대학교	컴퓨터공학	회전 교차로 알고리즘 구현 개발 환경 구축