

제어지능SW개발  
개인연구일지

## <차선 인식 프로그램>

2021년 6월 16일

연구자 정보			
학년	3	반	A
학번	201708015	이름	이한결

인덕대학교  
컴퓨터전자공학과

※ 작성 규칙

- 폰트 : 나눔 고딕 (선택사항)
- 본문 등 기본 폰트 크기 10
- 제목 폰트 크기 12
- 줄간격 : 배수 1.1

1. 개발 시스템(서비스) 제목

차선 감지 프로그램

2. 개발 시스템 개요 및 필요성

2.1 개발 시스템 개요

자율 주행 서비스 중 기초적인 기능으로 자율 주행 시 차선 이탈을 방지하기 위해 전방 시야 영상의 차선을 인식하고 판단해야 한다.

2.2 개발 필요성

자율 주행 프로그램 중 차선 감지 기능은 운전자에게 이동 시간에 다른 일을 할 수 있는 편의를 제공하고, 도로 위에서 발생하는 각종 사고를 예방할 수 있도록 돕기 위한 서비스의 기초적인 기능이다. 주행 중인 자동차가 현재 차선을 벗어나는 것을 방지하기 위해 현재 위치한 차선을 인식하고 파악해야 한다. 이로써 현재 내 위치, 전후좌우의 차량 위치를 파악하기 위한 기초 기능이다.

2.3 관련된 시장 분석 및 관련 기술 분석

〈시장〉

자율주행자동차는 기존 자동차 대비 교통혼잡도, 연료사용량, 오염배출량 감소효과를 가져오며, 운전자의 운전 부담을 줄여 차내 생산성 내지 여가시간이 증대되어 개인의 삶의 질 개선에 크게 기여함은 물론, 이로 인하여 차내 콘텐츠, 광고 등과 같은 서비스 산업이 성장하고, ICT 기술 접목에 따라 반도체, 소프트웨어와 같은 핵심 부품에 필요한 기술부분에서 새로운 사업영역을 파생시키는 효과가 있는 바, 현재 글로벌 자동차 업체, ICT 업체는 기업의 사업영역을 불문하고 자율주행관련 프로그램에 공동참여하거나 기술 제휴, 인수 합병 등을 통해 자율주행자동차 시장에 뛰어들고 있는 추세이다. 향후 자율주행자동차 시장규모에 대해서는 여러 컨설팅 기업 및 리서치 기업들이 각기 다른 예상을 하고 있는데, 그 중 럭스 리서치(Lux Research)에 의하면 2030년까지 자율주행자동차 시장규모가 880억 달러에 이를 것으로 예상하였고, 2030년 시장 점유율의 92%는 2단계 자율주행자동차가, 나머지 8%는 3단계 자율주행자동차가 차지할 것으로 전망하고 있다. 한편 글로벌 시장조사 업체 AMR(Allied Market Research)에 의하면 자율주행자동차 시장규모는 2019년 542억달러에서 2026년에는 5560억달러규모로, 연평균 39.47%의 성장율을 보이며 성장할 것으로 예상되었다.

미래차 분야는 급격한 기술진화와 산업 패러다임의 변화를 바탕으로 기존 완성차 차량의 저성장 기조 속에서도 급속한 팽창이 예상되며, 2030년 미래차 시장은 전기·수소차, 자율주행자동차, 이동서비스 산업이 주도할 것으로 전망된다.

〈기술〉

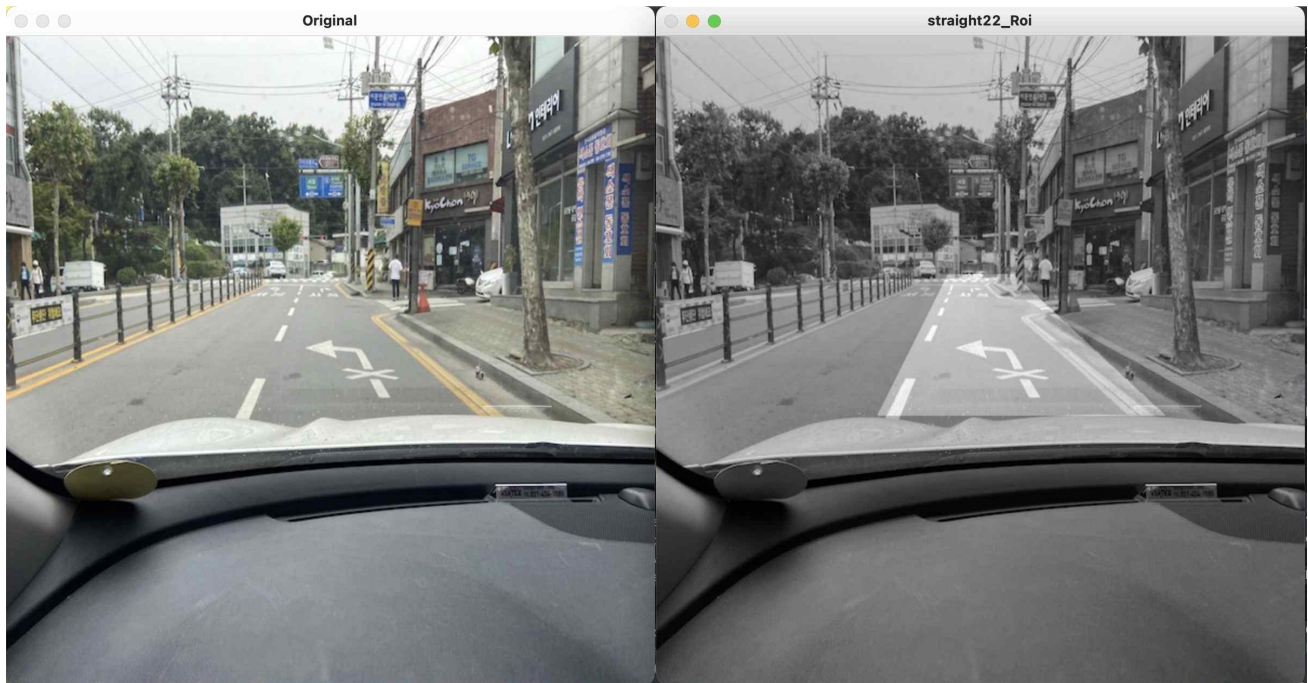
자율주행 자동차의 핵심 분야는 주행환경 인식, 위치 인식 및 맵핑, 판단, 제어, HCI(Human Computer Interaction)로 나눌 수 있다. 특히 레벨 3 이상의 자율주행이 가능하기 위해서는 이 모든 기술 구성요소를 포함하는 데이터 센서, 네트워크, 인공지능 기술의 고도화가 관건이다. 현재 자율주행차 시장에서 가장 비중이 큰 시장은 주변 환경 데이터를 인지하고 판단하는 센서이다. 자율주행차는 센서와 통신을 통해 데이터들을 생성·수집하는 일종의 디바이스이기 때문이다. 이미 카메라, 레이더 분야의 세계적 토종 기업인 인텔 모빌아이, 인피니언 등이 과점하고 있지만, 최근에는 스타트업의 R&D가 기술력을 집중시키고 있다. 우리나라도 스트라드 비전, 에스오에스랩 등 스타트업이 기술력을 차차 인정받는 추세다. 앞으로 카메라, 레이더(Radar), 라이다(LiDAR) 등 3대 분야가 고르게 개발된다면 세계적인 경쟁력을 갖추게 될 것으로 예상된다.

3.2 소프트웨어 개발 내용

※ 목표로 하는 소프트웨어 구성, 기능 등을 구체적으로 작성함.

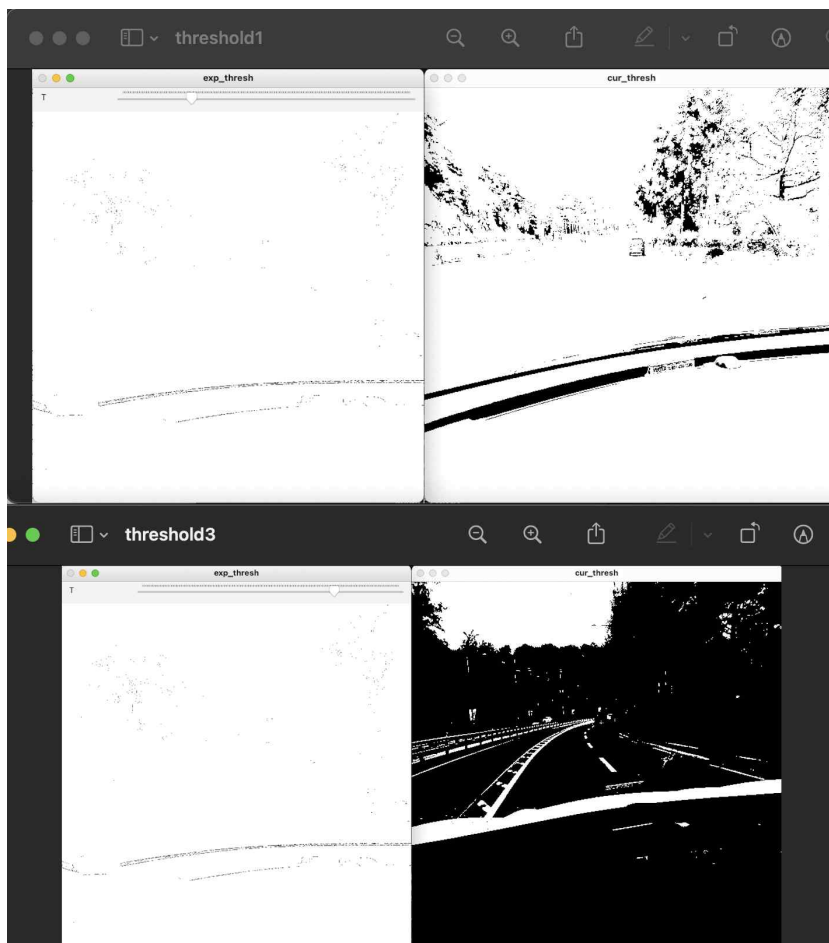
※ 자유롭게 작성함.

(5월 29일)관심 영역만 처리 - ROI(Region Of Interest)배열 사용 - 관심 픽셀 : 1, 비 관심 픽셀 : 0  
으로 지정하여 구별



ROI로 관심영역을 이차배열형태로 왼쪽위, 왼쪽아래, 오른쪽아래, 오른쪽위 순서로 지정하여 차선크기에 맞추어 설정을 해준다. 차선 밖의 비관심영역(나무, 건물, 중앙분리대 등)을 취급영역에서 제외시킴으로 노이즈를 크게 감소시킬 수 있다.

**threshold** - 임계값을 설정하여 설정된 임계값을 기준으로 흑, 백으로만 채색을 하여 픽셀을 구별한다. 흑백으로만 구별하여 차선과 도로를 확실히 구별하여 차선을 더 뚜렷하게 보정한다.



Histogram Equalize를 통해 영상을 더 선명하게 만든다.



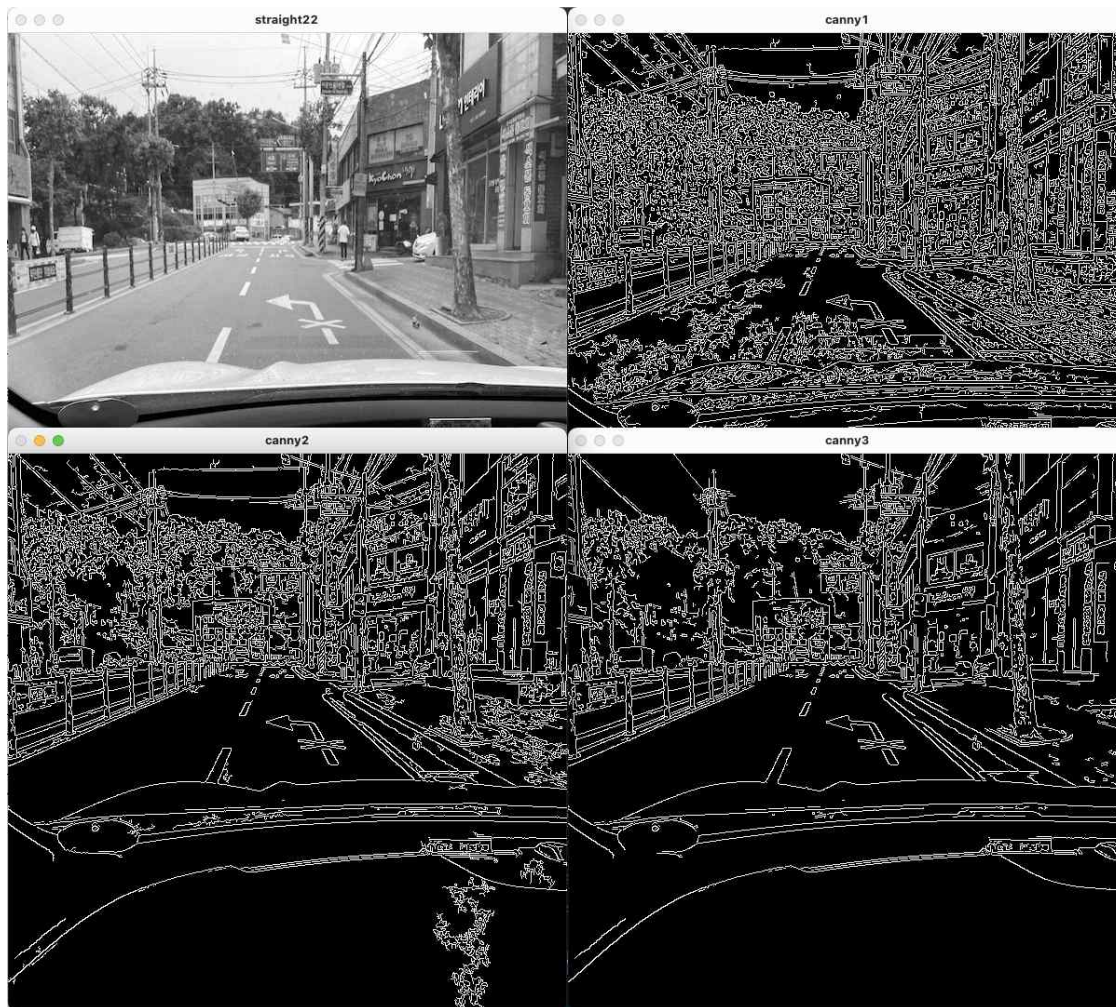
이미지를 전처리하여 픽셀이 0에서 255까지 고르게 분포하도록 하여 선명도를 높인다. 이는 선을 더 분명하게 하여 차선을 더욱 확실히 검출가능하게 도와주는 역할을 한다. 하지만 위 사진처럼 차선이 없는 아파트단지내 혹은 비 포장도로에서는 오히려 노이즈를 증가시킬 가능성이 있다. 이러한 경우를 해결하기 위해서는 어떤 방법을 구안해야되는지 아직 잘 모르겠다.

(6월3일)canny - canny는 흑백사진을 경계선이 확실한 픽셀을 특정 임계값을 기준으로 0, 1(흑,백)으로



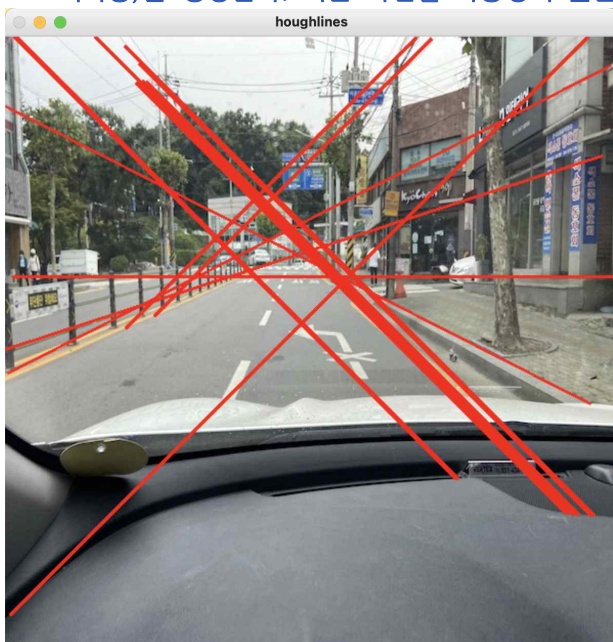


구분한다.

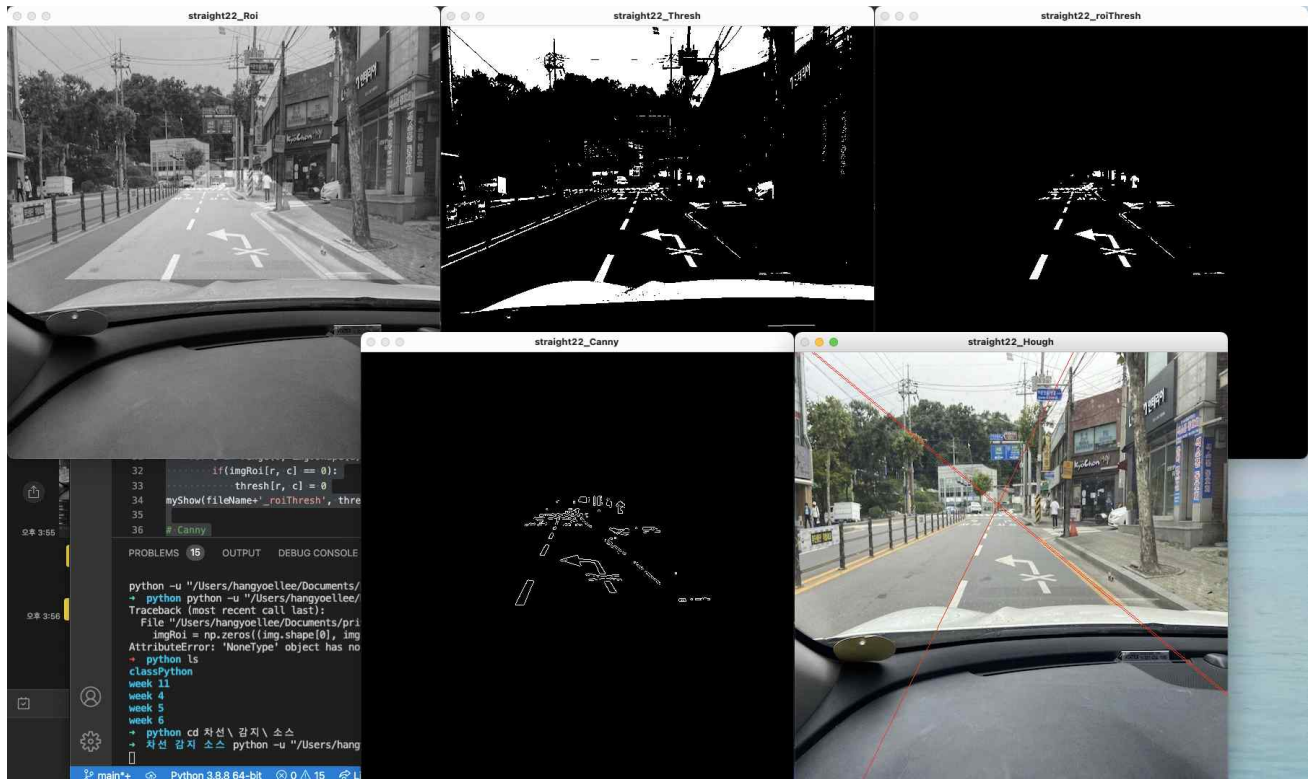


위 사진은 **edge**함수이다. **canny**는 edge보다 더 선명하게 검출하기 때문에 edge보다는 canny함수를 사용하는 것으로 선택한다.

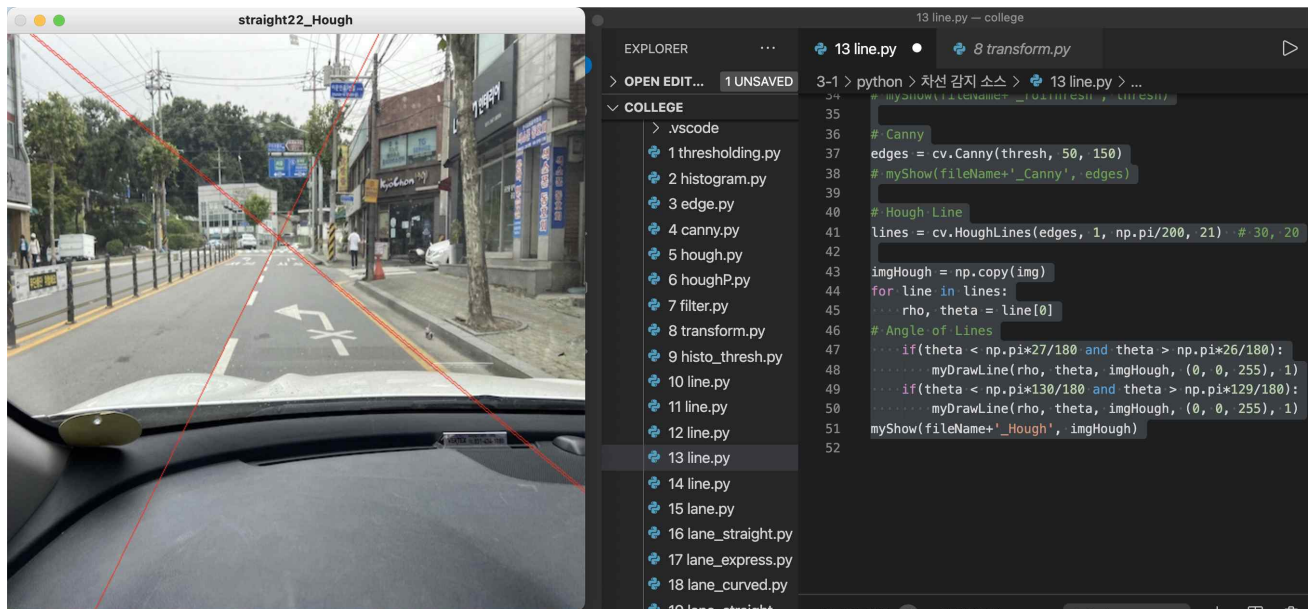
( )월9일6**hough**는 위와같이 검출된 선(직선X, 짧은선)을 이미지 전체에 직선으로 늘려 선명한 선(특정 BGR색가능)을 생성한다. 이는 차선일 가능성이 높은 선의 길이가 긴 선들을 검출한다.



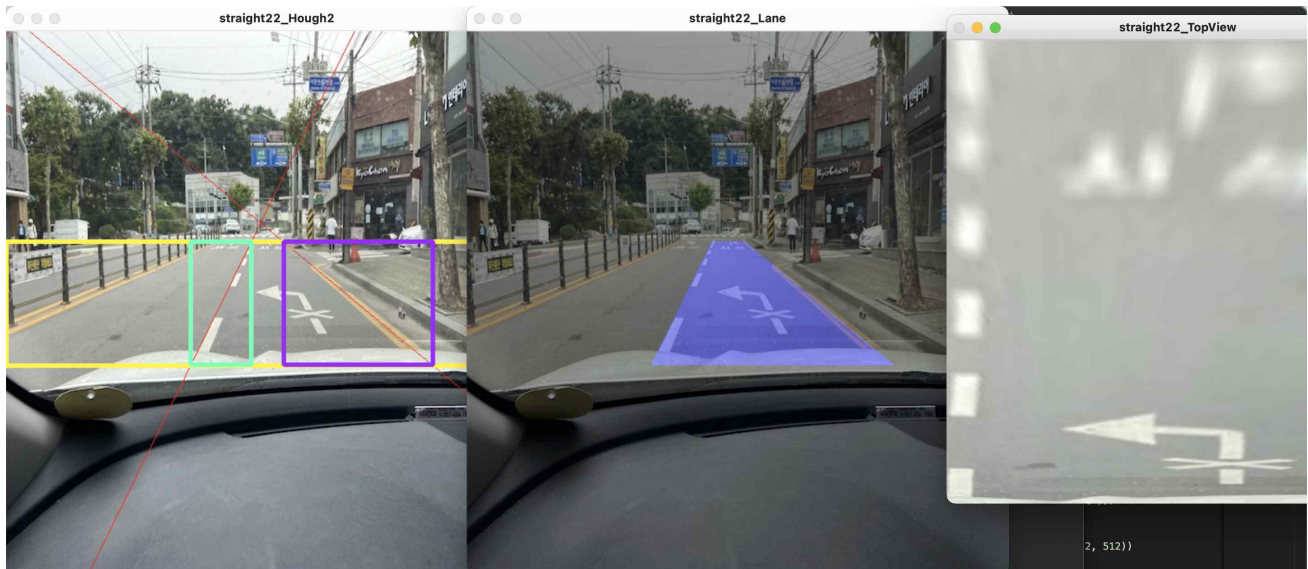
위와 같이 차선만 검출한 것이 아닌 다른 노이즈에 대해서도 선을 생성하기 때문에 앞에 함수들, ROI를 통해 특정 영역(픽셀)을 선택 - threshold - canny - hough 순서로 도로의 차선에 영역에 더 집중할 수 있도록 하여 아래 사진과 같이 차선에만 hough를 적용할 수 있도록 한다. 위 사진과 비교하였을 때 노이즈가 크게 감소하였다.



(6월11일) 위 hough이미지 중 다양한 각도의 line이 생성되어 광범위한 차선이 잡혀있다. 그렇기 때문에 특정 각도를 'theta'값에 저장하여 차선에 맞는 각도를 범위로 지정하여 보다 더 차선 각도에 가까운 hough를 적용할 수 있도록 한다.







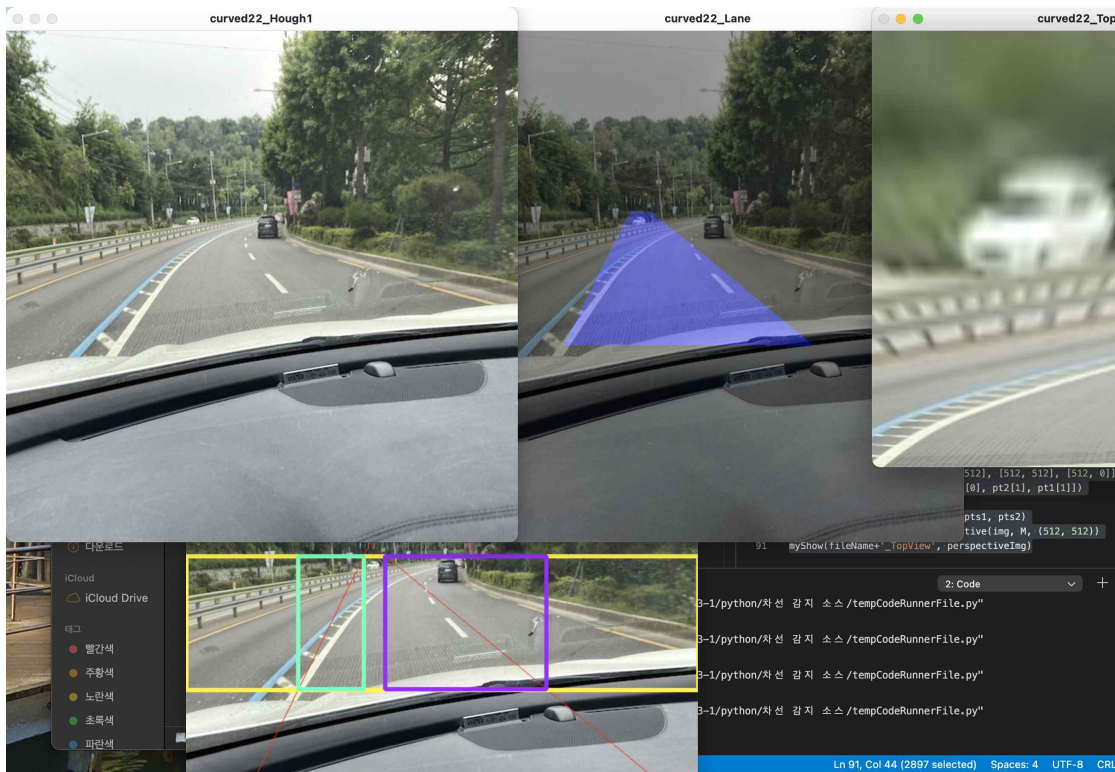
이후 **hough**의 선과 특정 거리의 차선을 선택하기 위해 **detecting**함수를 사용한다. **detecting**함수는 설정값에 사각형을 그린 후 **hough-line**과의 점점(4개의 꼭짓점)을 선택가능하게 한다. 이후 **fillConvexPoly**함수를 통해 얻은 꼭짓점을 기준으로 사각형(원근법에 의한 사다리꼴)을 이미지에 그릴 수 있다. 이는 차가 진행하는 도로를 선택할 수 있게 한다. 생성된 사각형의 윗변의 중앙, 밑변의 중앙을 잇는 선을 생성하면 현재 진행하는 **방향도 검출**할 수 있을 것 같다.

(6월16일)이 방법도 **Histogram Equalize**와 같이 차선이 없는 도로 이미지에서는 노이즈가 크게 발생하였다. 그렇기 때문에 ROI를 더 자세히 설정하여 차체에 맞는 범위를 선택하여 도로를 검출하여야 했다.

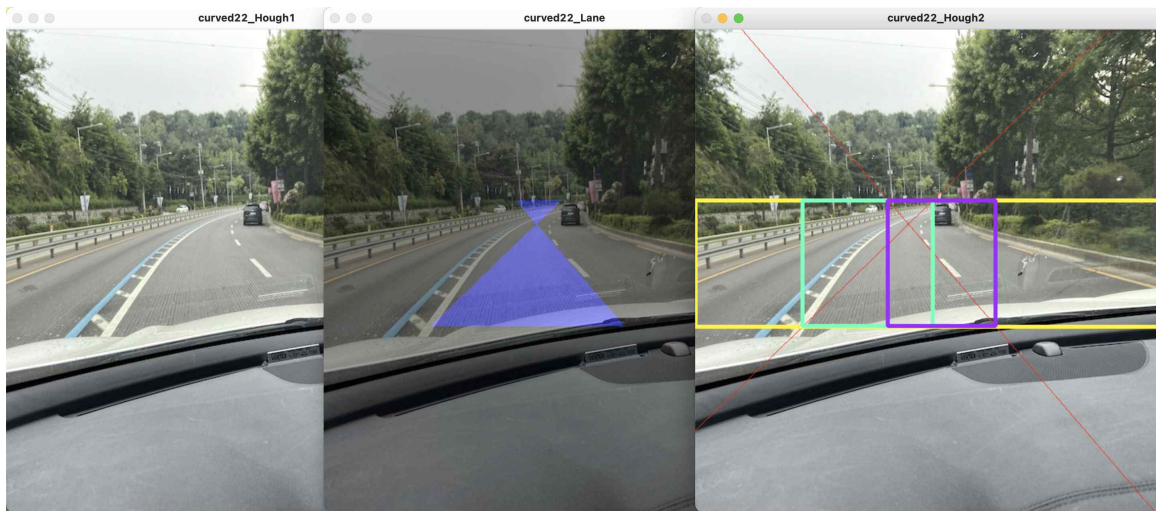


차선이 없는 도로의 검출은 다른 방법을 더 구안해야할 것 같다.

곡선차로에 경우는 위에 지정한 '**theta**'값이 오히려 차선 검출에 악영향을 끼쳤다  
아래 사진과 같이 곡률을 확인하지 않고 평균각도를 검출하면 진행하는 도로를 뜻하는 사다리꼴의 방향이 진행방향과 맞지 않았다.



따라서 곡률을 계산하여 'theta'값의 범위를 설정해주어야 하는데 곡률계산을 위해서는 따로 더 검색을 해보아야 할 것 같다. 우선은 해당 이미지에 차선 각도에 대해 맞는 'theta'범위 값을 지정해주어서 해결하였다.



#### 4. 소프트웨어 개발 결과

지금까지 완성한 차선 인식 프로그램은 곡선차선 등 더욱 섬세한 설정이 필요한 미완성 프로그램이지만 설정된 도로의 범위, 각도에 해당하는 차선은 검출이 가능하다.

#### 5. 개발된 시스템 사용 분야 및 활용 방안

아래 코드 중 'syROI','eyROI'는 세로, 가로 좌표의 범위를 지정한 값인데 속도에 따라 'syROI'의 값을 조절하여 관심 영역을 조절하는 기능이 추가된다면 활용도가 높아질 것이라 생각한다.



```
72     retval, pt1[i], pt2[i] = cv.clipLine((0, syROI, 511, eyROI-syROI), tuple(p1), tuple(p2))
73     cv.rectangle(imgHough, (0, syROI), (600, eyROI), (0, 238, 255), 3)
74     print(retval, pt1[i], pt2[i])
75     cv.rectangle(imgHough, tuple(pt1[0]), tuple(pt2[0]), (179, 255, 0), 3)
76     cv.rectangle(imgHough, tuple(pt1[1]), tuple(pt2[1]), (255, 0, 162), 3)
77     myShow(fileName+'_Hough2', imgHough)
```