-본선 설명회

3바퀴 돌고 주차해야함(3바퀴 돌기 전에 하면 실격)

횡단보도의 정지선을 차가 넘어가면 감점

횡단보도 직접만들어보면서 테스트 해보기

주차 p의 이미지를 인식해도 되고

표지판에서 벽 사이에서 주차하러 들어가야함. 규격이 있으니까 흰색 테이프나 종이를 이용해서 직접 만들어서 해보기

주차하면서 블록 3개를 쓰러트리면 안됨

보드 전원누르고 빨간 버튼 누르고 초록색 버튼 누르면 와이파이 연결

vnc연결해서 ( 연결 안되면 보드에 리셋버튼 누르기)

-비번=nvidia

-화면 크기 확장 누르면 640\*480에서 1000\*1000이 되니까 이걸로 하기

파란버튼 누르면 모터 전원 켜기

충전-보통 16v로 배터리 충전 / 배터리 쓰기 싫을 때가 생긴다 그때는 19v짜리를 직접 보드에 연결해 충전

(충전 중일때 배터리가 점점 차오르는 모양을 함)

모터 배터리 충전 중일때- 충전 안할때는 빨강 / 충전 중일때 초록색 깜빡깜빡 / 완충시 초록불빛 계속됨

스피드 조절

xycar\_ws/src/auto\_drive/src에 들어가서

gedit ./hough\_drive\_a2.py의 232줄 angle 옆에 4를 바꾸면 스피드 조절

노출값(카메라)

바탕화면에 노출값 확인에서 확

디렉토리 xycar\_ws/src/auto\_drive/launch에서

cam\_tune.launch파일 열기-’exposure’값을 변경

바탕화면

-카메라 위치 조정→ 팀끼리 각자 눈격자를 통해서 조정

-뷰어실행 → imu, 라이다, 카메라

-자율주행 → 차선따라 자율주행함

-0819 스터디

cd Desktop

ls

gedit x0.desktop

실제 하는것은 Exec / Name이 실제 우리가 바탕화면에서 보는 이름

-x1.sh 에서 가장중요한거는 맨 아래 roslaunch

-x2.sh=자율주행

roscd auto\_drive로 해서 들어가면 빨리 경로로 들어갈 수 있음

motor는 include해서 실행되고 있음

여기서 width, height변경

hough\_drive\_a2.launch

640\*480은 화각이 굉장히 작을텐데 이걸 조정해서 알맞게 하기

img\_sub라는변수에 640\*480이미지가 들어옴

h,w,c=img.shape[]해주면 일일히 안바꿔줘도 됨

rqt\_graph해서 보면 실행되는 것을 전체적으로 볼 수 있음

auto\_drive 노드만 보면될듯!

msg는 xycar\_ws/src/xycar\_motor/msg 에서 gedit xycar\_motor.msg에 들어가면

int라고 써있는데 이거 float으로 바꾸면 더 정밀하게 하기!

바꾸기전에 vesc에선 뭘로 받을까?!

xycar\_motor\_a2에 들어가보자

-hough\_drive\_a2.py

hough로 되어 있는데 그거 slidewindow로 바꿔도 되고 pid빠져있으니까 그거 추가해도됨!

-라이다 데이터(xycar\_lidar.launch)

rostopic list에서 무슨 토픽있는지 보고

클러스터링을 통해서 장애물인지를 판단

obstacle\_detector패키지로 하면

-obstacle\_detector패키지

깔아서 쓰기(armao,,, c++도 깔아서 써야함)

-인터넷/wifi 사용하기

sudo vim /etc/modprobe.d/bcmdhd.conf해서 세번째꺼 주석처리 하고 재부팅 후 와이파이 연결하면 됨

그럼 vnc는?!

xycar에서 ifconfig에서 wlan0 번호 보기

내pc에서 연결하려면 같은 wifi여야함.

이제 vnc켜서 xycar에 ifconfig되어있던 번호 치면 끝!

→ 같은 네트워크 안에 있으면 됨

1000\*1000이 꾸지니까 내 화면에 맞춰서 하기-→ 1920\*1080으로 바꿨음

지금 calibration이 빠져있으니까 작년 xycar에서 빼가기(카메라 왜곡)

-파일 보내주기

scp는 무조건 로컬에서 하기! nvidia에서 보내려면 로컬 컴퓨터 ip주소를 알아야하기 때문임

cp랑 같은 명령어임

ssh 연결되어있을 때 cp해주는 것이 scp

nvidia→내 로컬 : 내 로컬에서 scp [nvidia@10.42.0.1](mailto:nvidia@10.42.0.1):~/Desktop/vnc\_up ~/Desktop으로 하면 끝!

내 로컬 → nvidia : 내 로컬에서 scp copy nvidia@ [10.42.0.1](mailto:nvidia@10.42.0.1):~/Desktop 하면 됨(copy가 내 로컬에서 보내주고 싶은 파일)

-리부팅할때ㅎ

터미널에 reboot쳐주면 됨

vesc/vesc\_driver/yaml/vesc.yaml

에서 offset을 맞춰줘야함! - 조이스틱을 써서 직진시켜서 맞춰보기

speed max도 바꿔주기! 작년에는 3만으로 바꿨음

xycar\_motor\_a2.lauch 에도 param angle\_offset을 바꿔도 됨

==> 뭐가 되었든간에 조향 맞춰줘야함!!!

wifi비밀번호 바꿔서 안털리게 하기!

**금요일에 할 것**

**-오늘 들은 것 설명**

**-허프라인 빼고 슬라이딩 윈도우 넣어보기**

**-pid 넣어보기**

**-작년 xycar에서 calibration빼오기**

**-obstacle\_detector넣기**

**-offset은 자주스 열리면 하기**

**-process이미지함수**

**\*횡단보도 인식하기 위해**

**sobel 가로인식 함수 + houghline함수 넣어서 허프라인에서 구한 게 길이가 일정이상이면 횡단보도로 인식해서 멈추게 하기**

**-obstacle\_detector**

**이거 xycar\_car에 해줘야하니까 런치파일이고 뭐고 바꿔주기**

**-작년코드로 장애물인식하는거 해보기**

**hough\_drive\_a2.launch에서 해상동 1280\*720으로 바꿔서 해상도 높게(두차선이 다 보이게!)**

**원래는 640\*480이었움**

**-git 쓰기!**

**-클래스별로 나누기**

**(pid제어하기(twiddle몰라도됨) / slidingwindow도 클래스로 따로)**

**9./14**

**1. flag –> 지금 무조건 true로 들어가서 안먹음-->check라는 변수 만들어서 테스트 해보기**

**2, 기본코드에 횡단보도 넣어보고 주행해보기**

**3. 작년 코드에서 curvedetector랑 pidcal 넣어보기**

**4. 클래스별로 나누기**

**(Curvedetector & obstacle\_detector & stop\_counter & hough\_drive\_a2 & pidcal & slidingwindow & crosswalk\_counter)**

**5. 메일 온 것 steering / max speed값 변경하기0. 속도도 앵글 처럼 바꾸기 xycar\_motor\_a2.py**

**7. 앵글값 주는게 알고보니 ratio?이상한게 곱해져서 단위가 이상했음 → 바꿔줌**

**8. x\_location구할 때 x\_represent에 차선의 width/2를 빼줌(roi 화면에서 확인한 차선의 너비)**

**9/16**

**오늘 한것**

**-x\_location y 축 아래로 낮춰서 더 가까이서 조향하게 함**

**-slidingwindow rectangle 범위 줄임**

**-slidingwindow 양쪽말고 왼쪽차선만 보고 달리게 함 ( 차선 폭 재보고 직접 값 넣어줌)**

**-stop\_counter : yellow범위 다시 잡아야함 - 인식안되고 있음**

**-crosswalk\_counter : 속도에 따라 정지하는 구간이 다름 – 속도 정하고 값 출력해서 줘야할 듯**

**-pid 줄 때 curve\_count=2라고 하고 2일때의 pid로 설정함 ( 작년엔 curve\_count<2일 때 p[0]=0.002고 else일 때 0.0035로 줬는데 지금 0.0035로 아예 정하고 돌림)**

**-obstacle : 랜덤으로 인식됨**

**-s자 구간 계속 튕겨나감**

**커브카운트 hough-main에 초기화하는 것 추가하기**

**차선 나가면 x\_represent 에러나는 것 예외처리하기!**

**==> 직선구간은 잘됨 / 커브 랜덤 /**

**장애물 피할 때 횡단보도 인식이 되어버림!**

**→**

**1. 장애물 인지하면 바로 flag를 false로 하고 장애물 미션 끝나면 바로 true로 바꾸는 방법 생각해보기~!**

**2. 장애물 피할 때 한번에 sin곡선 그려서 피하기 → 이 방법은 flag 안바꿔도 됨**

**-임태현오빠 조언-**

**s자 코스 : x\_location이 작아지다가 값자기 커지면 차선이 이상한거니까 오른쪽 차선보고 주행하기**

**출발선 : 노란색 다시 범위 잡기**

**차선인식->중심점을 왼쪽으로 더 보내서 왼쪽에 붙어서 주행하게 해보기 (시도만^^)**

**출발할 때 stop counter~obstacle끝까지 flag=1(obstacle개수 체크하기)**

**obstacle 끝~횡단보도 끝까지 flag=2 (-→ obstacle 개수 세는 법 생각해야함)**

**횡단보도 끝~start까지 flag=3**

**장애물 빠져나오면(stop\_counter.flag==2일 때로 함) 속도 2정도?로 고정해서 횡단보도 전까지 일정하게 해서 if문 조건 바꾸기 → 그럼 횡단보도 인식은 완벽!**

**9/21**

**자주스 : pid 값 크게 해보기**