

메이커스! AI 자율주행자동차

나만의 인공지능 자율주행자동차 만들기

강사 소개

- 서울대 사회학과
- 전)LG-CNS 시스템 엔지니어
- 현)유비온 부사장/연구소장
- ISO/IEC JTC1 SC36 전문위원

목 차

1. 과정소개 및 자동차 조립
2. 라즈비언 운영체제 및 파이썬 실행환경 이해
3. 모터 제어 및 키보드 키값 활용하기
4. 카메라 활용 및 데이터 수집 방법
5. 인공지능 학습프로세스 이해 및 데이터 수집
6. 인공지능 훈련하기
7. 자율주행 테스트 및 추가 훈련하기
8. 인공지능 원리에 대한 이해

학습목표 및 학습내용

- 학습목표

소프트웨어 개발과 사물인터넷에 대한 개념을 이해하고 인공지능의 작동 원리를 실습을 통해 경험한다

- 학습내용

- 자율주행자동차의 개념, 기능, 장치, 단계에 대해 배운다.
- AI자동차를 조립한다.
- 모터와 카메라를 프로그램으로 조작한다.
- 인공지능을 위한 학습데이터를 수집하고 인공지능을 훈련시킨다.
- 인공지능 원리와 용어에 대해 배운다.

Challenge

- 인공지능 지도학습을 위한 훈련데이터(x, y)가 10,000개가 있는데, class1, class2, ..., class10 로 잘 정렬 되어있다. 먼저 class1에 대해 학습하고 이후 순차적으로 class2를, 이런식으로 class10까지 학습을 진행하였다. 훈련 결과는?(잘됨, 못됨)
- 가위, 바위, 보 사진을 찍은 후 라벨링까지 진행하고 인공지능 훈련을 진행하려고 한다. 가위는 100장의 이미지가 있고 바위는 50장, 보는 30장의 이미지가 준비되어 있다. 딥러닝 CNN 기법을 활용하여 훈련하고자 한다. 인공지능 훈련을 하기에 앞서 데이터 가공이 필요한데, 가공을 위한 어떤 작업이 필요한가?
- (딥러닝 관련)인공지능을 위한 학습데이터는 Training Data, Validation Data와 Test Data로 나뉜다. 이 중 Training Data와 Validation Data는 인공지능 훈련에 사용되며 특히, Back propagation에 같이 활용된다. (참, 거짓)
- Test Data를 Validation Data로 사용하면 문제가 되는가?

실습 관련 사항

- 노트북에서 vnc viewer or visual studio code로 kit 연결
wifi ap: ssid RpiAICar001~10 password 12345678
id address: 192.168.50.1 id/pw: pi/raspberry
- 자동차 차륜정렬
- 데이터 수집
- 데이터 확인 및 정제/가공
- 인공지능훈련
- 성능검증
- 자율주행 확인
- 데이터 삭제: python data_delete.py

1

과정소개 및 자동차 조립

자율주행 자동차란?

- "자율주행자동차"란 운전자 또는 승객의 조작 없이 자동차 스스로 운행이 가능한 자동차를 말한다([자동차관리법](#) 제2조 제1호의3)
- 자율주행자동차는 인공지능(AI), 사물인터넷(IoT), 5세대 이동통신(5G)을 아우르는 기술 복합체로서 미래 산업의 핵심이 될 것으로 예측되고 있습니다.
- 구글, 테슬라를 비롯해서 전세계 자동차 브랜드들이 기술 경쟁을 벌이고 있습니다. IEEE의 예측에 따르면 2040년이 되면 전체 차량의 87%가 자율주행자동차 일 것으로 예측된다고 합니다.

과정소개 및 자동차 조립

자율주행 자동차 기술 수준

도표 28 자율주행 기술 수준 단계

					
	Level 0	Level 1	Level 2	Level 3	Level 4
	No Automation	Function Specific Automation	Combined Function Automation	Limited Self-Driving Automation	Full Self-Driving Automation
	운전자에 의해 완벽하게 제어되는 차량	1개 이상의 특정 제어기능을 갖춘 자동화 시스템	2개 이상의 특정 제어기능을 갖춘 자동화 시스템	가속, 주행, 제동 모두 자동으로 수행하는 자동화 시스템 (필요 시 운전자 개입)	100% 자율주행
운전자 역할	직접 운전	직접 운전 (운전 보조장치)	운전자 주행상황 항상 주시	운전자 자동운전 결정	운전자 목적지 입력까지만
대표 기능		크루즈컨트롤, 전자식안정화컨트롤, 차선 인식 등	장애물 회피, 브레이크 제어, 주차보조기능 등	교통혼잡시 자동차 스스로 저속주행, 운전자 조작없이 고속도로 주행, 자동 차선변경 등	
	Now			2020+	2025+

자료: NHTSA 참고, KT경제경영연구소 재구성, 유진투자증권

1

과정소개 및 자동차 조립



1

과정소개 및 자동차 조립

자율주행 자동차 기능

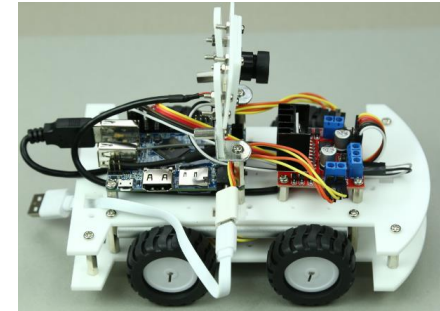
자율주행 자동차에는 어떤 기능과 장치가 있는지 살펴 보겠습니다.

- 차선유지기능: 카메라를 통해 도로를 인식하고 차선이 유지되도록 핸들을 조종합니다.
- 주행기능: 카메라 및 전파탐지기(RADAR) 장치를 통해 도로표지판, 신호등, 앞뒤 차간 거리를 파악해서 속도를 조절합니다.
- 주변인식 및 위험감지기능: 광선레이더(LIDAR), 카메라를 통해 차량 주위의 상태를 모니터링 합니다. 보행자, 자전거, 자동차 등의 움직임을 예측해서 사고를 미리 예방합니다.
- 주차기능: 자동차가 스스로 주차구역의 상황을 인식해서 자율주차를 수행 합니다.
- 기타: 이 밖에도 운전자 졸음 인식, 충돌예측, 주행선이탈감지, GPS, 등 다양한 기능이 있습니다.

1

과정소개 및 자동차 조립

자동차 조향 방법



1

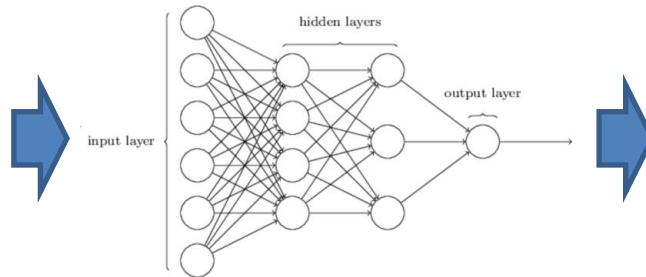
과정소개 및 자동차 조립

차선 유지 원리(실제자동차)

원인값



인공신경망(CNN)



예측값

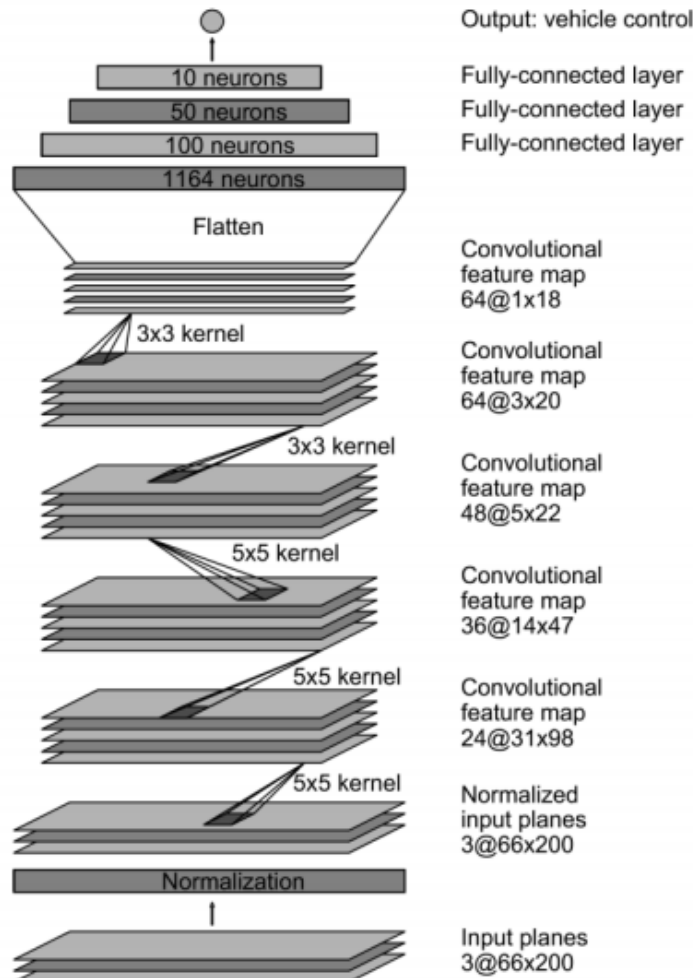


참값(결과값)



오차(Loss)

Autonomous Driving: End-to-End



- 9 layers
 - 1 normalization layer
 - 5 convolutional layers
 - 3 fully connected layers
- 27 million connections
- 250 thousand parameters

1

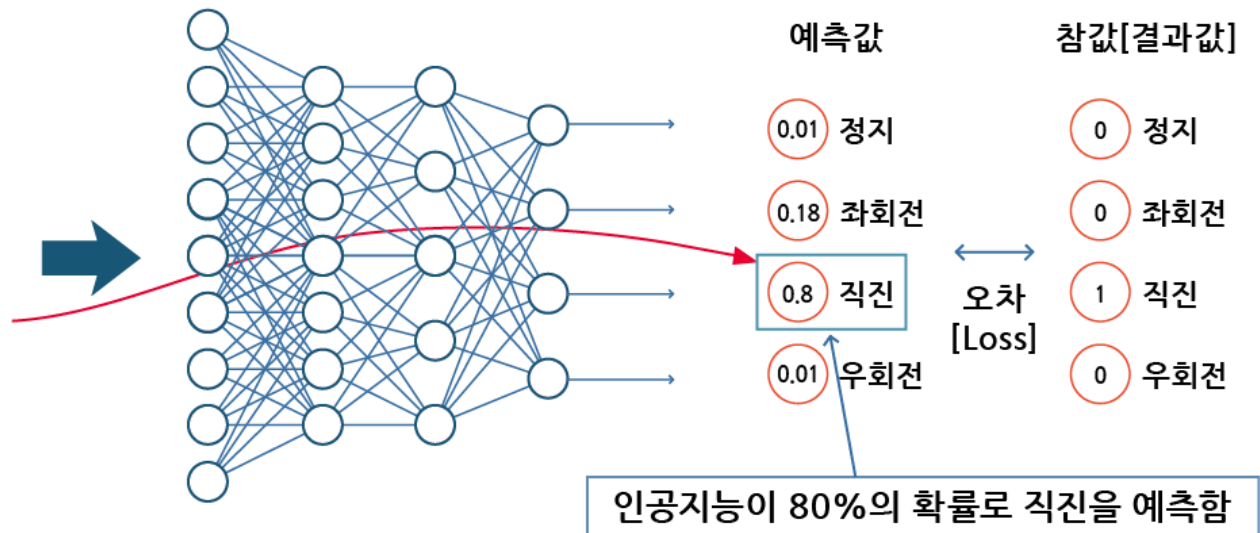
과정소개 및 자동차 조립

차선 유지 원리(AI자동차키트)

원인값



인공신경망[CNN]

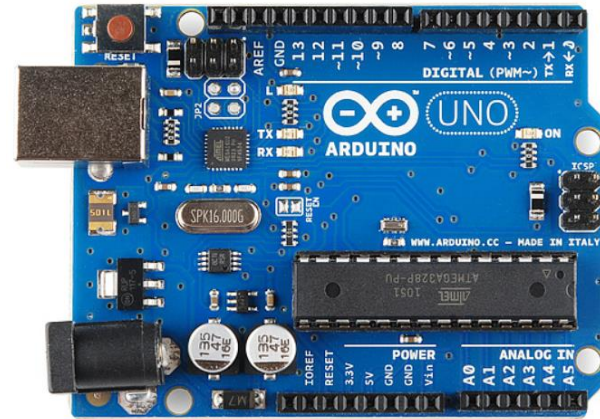


1

과정소개 및 자동차 조립

사물인터넷 기기

라즈베리파이 VS 아두이노



1

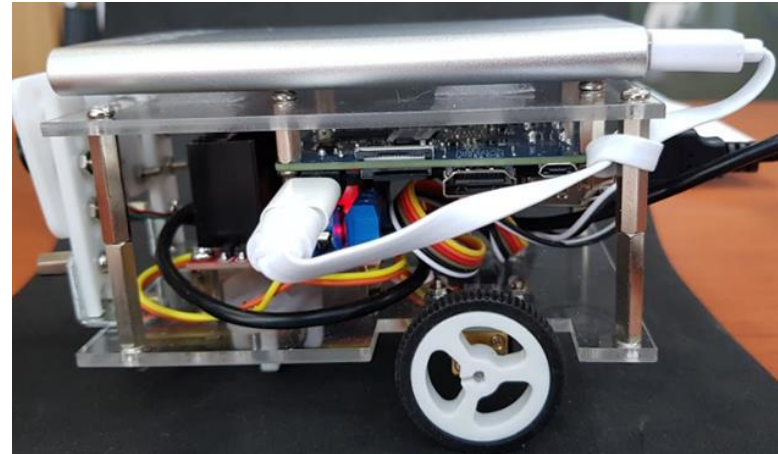
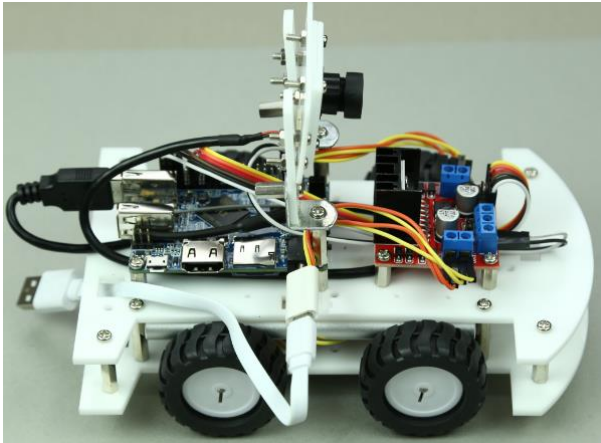
과정소개 및 자동차 조립

자동차 조립하기

설명서 대로 차근차근 따라서 조립합니다.
궁금한 점이 있으면 언제든지 손을 들어 주세요

<주의사항>

- 천천히 하세요
- 볼트/너트가 작아서 잃어버리기 쉬워요
- 카메라 연결이 다소 어려울 수 있습니다. 잘못 연결하면 라즈베리파이 보드 손상이 될 수 있으니 주의하세요.
- 아크릴 보드는 충격에 약하니 볼트/너트를 적당한 힘으로 돌려주세요. 너무 세게 돌리면 보드가 깨질 수 있어요.
- 단계별로 진행하시고 단계가 끝나면 강사/조교에게 점검을 받으시면 좋아요(잘못된 상태로 많이 가면 수정시간이 많이 들어요)



2

라즈비언 운영체제 및 파이썬 실행환경 이해

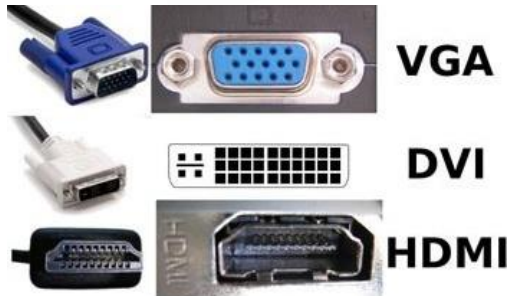
기기연결 및 부팅하기

- SD 카드 장착
- 모니터 연결
- 키보드/마우스 연결
- 충전선 연결
- 부팅하기
- 종료하기
- 정상종료/비정상종료

2

라즈비언 운영체제 및 파이썬 실행환경 이해

오렌지파이 연결 방법1(직접연결)



무선키보드 및 마우스가 있으면 편리합니다.

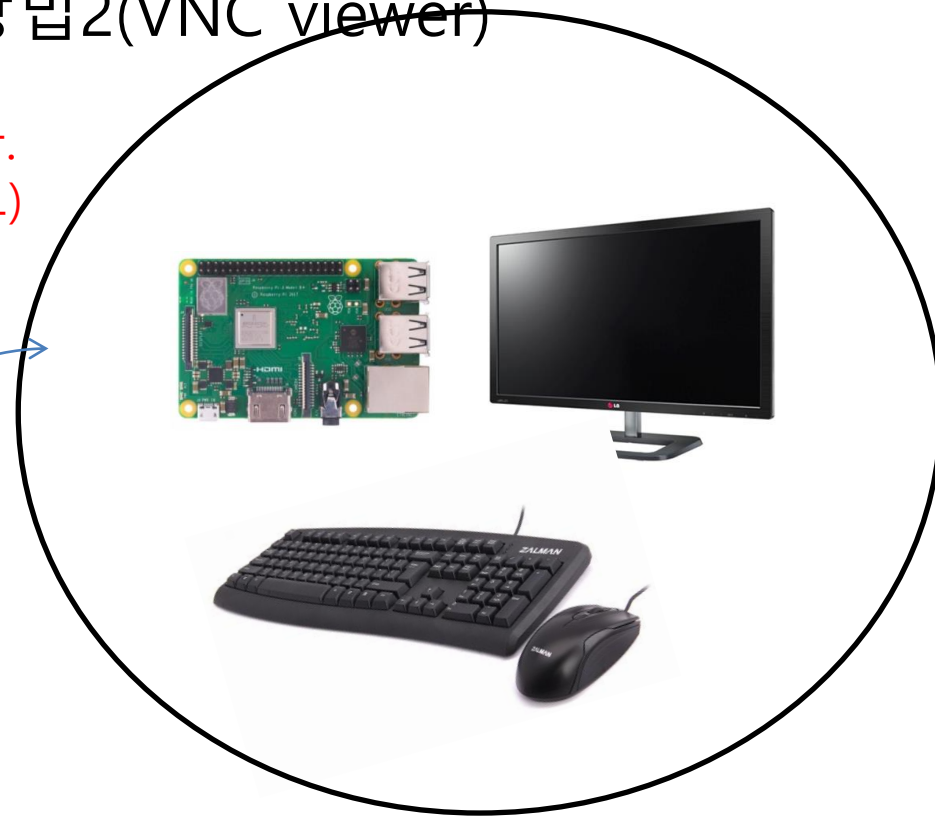
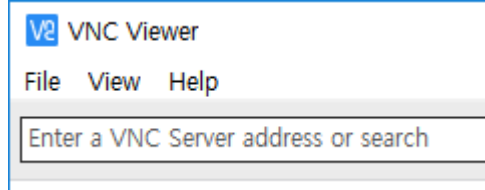
2

라즈비언 운영체제 및 파이썬 실행환경 이해

라즈베리파이 연결 방법2(VNC viewer)

노트북에서 와이파이 AP와 연결합니다.
AICar01~30(자신의 번호를 이용하세요)
비번: 12345678

원격 접속



Vnc viewer에서 192.168.50.1 ~30
자신의 번호를 이용하세요.

2

라즈비언 운영체제 및 파이썬 실행환경 이해

라즈비언(Raspbian) 운영체제 이해

- 운영체제란(OS)?
리눅스, 윈도우, Mac OS, 안드로이드
- 라즈비언은?
- 메뉴 살펴보기

2

라즈비언 운영체제 및 파이썬 실행환경 이해

터미널 실행 및 명령어 살펴보기

- 터미널 열기
- 터미널 명령어 살펴보기
 - su : 관리자 계정으로 사용하기
 - ifconfig: ip address 확인하는 명령어
 - pwd(Present working directory): 현재 폴더
 - ls(list): 폴더 구조를 보여줌
 - cd(change directory): 폴더 위치를 변경
 - ~: 사용자의 홈폴더
 - .: 현재폴더
 - .. : 현재폴더의 부모 폴더
 - ↑ : 이전 명령어 불러오기
 - ↓ : 다음 명령어 불러오기

2

라즈비언 운영체제 및 파이썬 실행환경 이해

파일관리자(File Manager)

- 폴더 이동
- 새폴더 만들기
- 새파일 만들기
- 폴더명/파일명 변경하기
- 폴더 지우기
- Texteditor(메모장) 사용하기

2

라즈비안 운영체제 및 파이썬 실행환경 이해

간단한 파이썬 프로그램 작성 및 실행하기

- 파이썬이란?
- 실행방법: python vs python3 실행파일명
- Hello.py 작성
- 코드/소스코드, 코딩/프로그래밍, 프로그램/라이브러리/시스템 용어의 이해
- 변수, 연산자, 할당, 함수, 파라미터 이해하기

3

모터 제어 및 키보드 키값 활용하기

모터와 모터쉴드

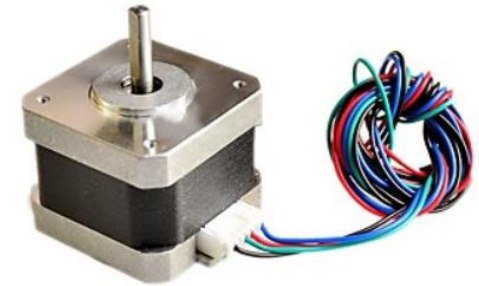
- 모터의 종류



DC Motor

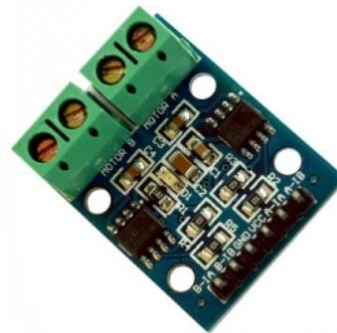
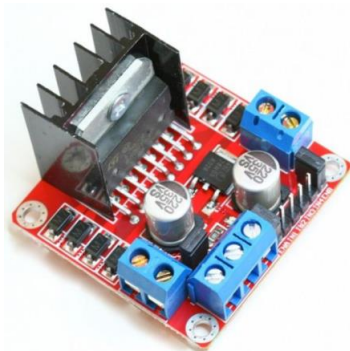


Servo Motor



Step Motor

- 모터쉴드(모터드라이브)



3

모터 제어 및 키보드 키값 활용하기

모터실습

- Motor_one: 자동차의 오른쪽 바퀴
- Motor_two: 자동차의 왼쪽 바퀴
- Motor_one_speed(속도): -100 ~ 100
 - : 뒤로 회전
 - 0 : 모터 정지
 - +: 앞으로 회전

3

모터 제어 및 키보드 키값 활용하기

키보드 키값 활용법

- 키보드 키값 받아오기
- 키보드 키값을 통한 프로그램 종료
- 토글키 방식 이해하기
- 키보드의 키값을 통해 자동차 좌/우회전 및 직진 구현하기
- 차륜정렬하기(wheel alignment)

4

카메라 활용 및 데이터 수집 방법

카메라 점검 및 영상 보기

- 카메라 렌즈 초점 맞추기
- 카메라 영상 불러 오기
- 카메라 이미지 저장하기

데이터 수집 방법

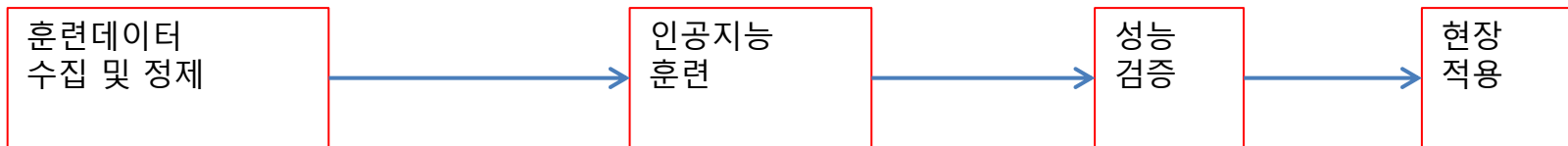
- CSV파일 만들기
- CSV파일에 데이터 저장하기
- CSV파일 편집하기

5

인공지능 학습프로세스 이해 및 데이터 수집

인공지능 훈련 프로세스

- 훈련프로세스



- 좋은 데이터 VS 나쁜 데이터
- 시범운전하기
- 훈련데이터 수집하기
- 훈련데이터 정제하기

5

인공지능 학습프로세스 이해 및 데이터 수집

좋은도로/나쁜도로, 좋은데이터/나쁜 데이터

좋은 도로 예제들



좋지 않은 도로 예제들

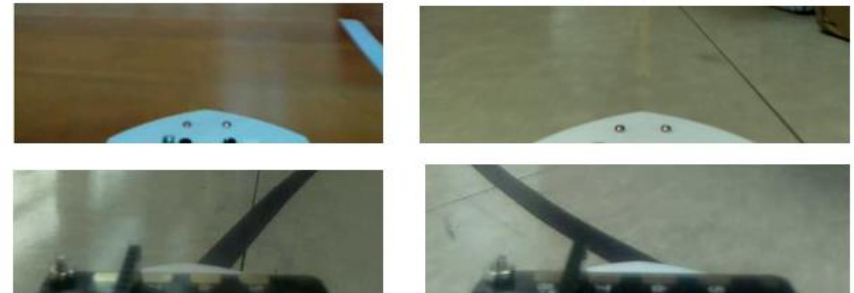


좋은 훈련데이터의 예



자동차가 도로 가운데 잘 위치해 있습니다.

나쁜 훈련데이터의 예



자동차가 도로 차선을 벗어났거나 도로가 끝난 지점 또는 도로가 아닌 곳에 위치해 있습니다.

5

인공지능 학습프로세스 이해 및 데이터 수집

훈련데이터 분석

- python3 data_analysis.py 실행

```

Armbian - root@orangepilite: /home/orangepi/autonomousCar/lesson4
File Edit View Terminal Tabs Help
root@orangepilite:/home/orangepi/autonomousCar/lesson4# python3 data_analysis.py
Total data counts: 833
Stop data counts: 4 , ratio(%): 0.5
Left data counts: 131 , ratio(%): 15.7
strait data counts: 499 , ratio(%): 59.9
Right data counts: 199 , ratio(%): 23.9
root@orangepilite:/home/orangepi/autonomousCar/lesson4#
    
```

정지 상태의 데이터는 가급적 제거하는 것이 좋습니다. 데이터량이 다른 카테고리에 비해 과소하면 영향이 없을 수 있습니다.

5

인공지능 학습프로세스 이해 및 데이터 수집

데이터 분포도 조정하기

- python3 decalcom.py 실행
- python3 data_analysis.py 실행

```
Armbian - root@orangepilite: /home/orangepi/autonomousCar/lesson4
File Edit View Terminal Tabs Help
root@orangepilite:/home/orangepi/autonomousCar/lesson4# python3 decalcom.py
root@orangepilite:/home/orangepi/autonomousCar/lesson4# python3 data_analysis.py
Total data counts: 1662
Stop data counts: 4 , ratio(%): 0.2
Left data counts: 330 , ratio(%): 19.9
strait data counts: 998 , ratio(%): 60.0
Right data counts: 330 , ratio(%): 19.9
root@orangepilite:/home/orangepi/autonomousCar/lesson4#
```

- data.csv 내용 확인하기

5

인공지능 학습프로세스 이해 및 데이터 수집

데이터 분포도 조정하기

- 데이터 불균형 해소 방법
Down Sampling VS Up Sampling
- python3 data_upsampling.py 실행
- python3 data_analysis.py 실행

```
Armbian - root@orangepilite: /home/orangepi/autonomousCar/lesson4
File Edit View Terminal Tabs Help
root@orangepilite:/home/orangepi/autonomousCar/lesson4# python3 data_upsampling.py
root@orangepilite:/home/orangepi/autonomousCar/lesson4# python3 data_analysis.py
Total data counts: 2322
Stop data counts: 4 , ratio(%): 0.2
Left data counts: 660 , ratio(%): 28.4
strait data counts: 998 , ratio(%): 43.0
Right data counts: 660 , ratio(%): 28.4
root@orangepilite:/home/orangepi/autonomousCar/lesson4#
```

6

인공지능 훈련하기

- python3 train.py 실행

(이 작업은 CPU를 많이 쓰는 작업이라 다른 작업을 하지 않는 게 좋습니다. 다른 작업을 하는 경우는 마우스 커서가 잘 움직이지 않는 경우를 경험하게 됩니다.)

```

Armbian - orangepi@orangepilite: ~/autonomousCar/lesson4
File Edit View Terminal Tabs Help
orangepi@orangepilite:~/autonomousCar/lesson4$ python3 train.py
model read
/usr/lib/python3/dist-packages/scipy/misc/pilutil.py:479: FutureWarning: Conversion of the second argument of issubdtype from `int` to `np.signedinteger` is deprecated. In future, it will be treated as `np.int32 == np.dtype(int).type`.
  if issubdtype(ts, int):
/usr/lib/python3/dist-packages/scipy/misc/pilutil.py:482: FutureWarning: Conversion of the second argument of issubdtype from `float` to `np.floating` is deprecated. In future, it will be treated as `np.float64 == np.dtype(float).type`.
  elif issubdtype(type(size), float):
Epoch: 0, Step: 0, Loss: 1.38585
Epoch: 0, Step: 0, Loss_val: 1.38585
Model saved in file: ./save/model.ckpt
Epoch: 0, Step: 1, Loss: 1.38549
Epoch: 0, Step: 2, Loss: 1.34278
Epoch: 0, Step: 3, Loss: 1.37133
Epoch: 0, Step: 4, Loss: 1.36863
Epoch: 0, Step: 5, Loss: 1.32307
Epoch: 0, Step: 6, Loss: 1.24639
Epoch: 0, Step: 7, Loss: 1.12054
Epoch: 0, Step: 8, Loss: 1.14074
Epoch: 0, Step: 9, Loss: 1.18008

```

6

인공지능 훈련하기

```

Armbian - orangepi@orangepilite: ~/autonomousCar/lesson4
File Edit View Terminal Tabs Help
Epoch: 19, Step: 21, Loss: 0.402934
Epoch: 19, Step: 22, Loss: 0.399025
Epoch: 19, Step: 23, Loss: 0.331773
Epoch: 19, Step: 24, Loss: 0.436715
Epoch: 19, Step: 25, Loss: 0.345112
Epoch: 19, Step: 26, Loss: 0.331154
Epoch: 19, Step: 27, Loss: 0.502944
Epoch: 19, Step: 28, Loss: 0.421297
Epoch: 19, Step: 29, Loss: 0.379544
Epoch: 19, Step: 30, Loss: 0.357097
Epoch: 19, Step: 30, Loss_val: 0.389838
Epoch: 19, Step: 31, Loss: 0.316761
Epoch: 19, Step: 32, Loss: 0.381643
Epoch: 19, Step: 33, Loss: 0.397278
Epoch: 19, Step: 34, Loss: 0.43349
Epoch: 19, Step: 35, Loss: 0.362159
Train Accuracy: 0.82
Validation Accuracy: 0.83
begin: 2019-06-11_17-52-33
end: 2019-06-12_00-03-07
Run the command line:
--> tensorboard --logdir=./logs --port=6006
Then open http://0.0.0.0:6006/ into your web browser
orangepi@orangepilite:~/autonomousCar/lesson4$
    
```

6

인공지능 훈련하기

인공지능 학습 데이터

- 훈련데이터(Training Data): 인공지능이 스스로 학습을 하는데 사용되는 데이터입니다.
- 검증데이터(Validation Data): 인공지능이 직접적으로 활용하지 않고 훈련 데이터를 통한 학습과 비교하는 용도로 활용 됩니다.
- 테스트데이터(Test Data): 직접 훈련에 활용되지 않고 인공지능 훈련이 완료된 시점에 잘 작동하는지 확인하기 위해서 사용되는 데이터입니다. 테스트데이터는 현실 데이터로 간주할 수 있기 때문에 테스트데이터에서 잘 작동하는 것이 중요하고 이 때의 성능이 진정한 인공지능의 성능으로 볼 수 있습니다.
- 훈련데이터와 검증데이터 비율은 7:3 또는 8:2 를 많이 활용합니다.

6

인공지능 훈련하기

적절한 훈련 종료 시점

- Training loss vs Validation loss



- 과도한 학습(Over fitting)은 글자 그대로 훈련데이터에 지나치게 편중된 학습을 하여 훈련데이터 학습은 잘 된 반면 일반화가 잘 안되어서 실제 상황에서 잘 작동하지 않는 경우를 말합니다.

6

인공지능 훈련하기

훈련 용어 및 환경

- Loss(or cost) : 오차
- Epoch: 훈련데이터 전체를 한번 훈련하는 것
- Batch(배치): 작업단위를 말함. AI자동차의 경우는 100개의 이미지를 한번에 처리함.
- Step: 배치를 한번 훈련하는 것
- Accuracy: 정확도(올바르게 예측한 개수/전체 데이터 개수)
- Model
- Input data: 독립변수(원인값)
- Label: 종속변수(결과값, 참값)
- Predicted value: 예측값
- 훈련관련 변수(hyper parameter): learning rate 등

6

인공지능 훈련하기

인공지능 훈련상태 관찰

```
Armbian - orangepi@orangepilite: ~/autonomousCar/lesson4
File Edit View Terminal Tabs Help
orangepi@orangepilite:~/autonomousCar/lesson4$ sudo python3 train.py
model read
/usr/lib/python3/dist-packages/scipy/misc/pilutil.py:479: FutureWarning:
ion of the second argument of issubdtype from `int` to `np.signedinte
recated. In future, it will be treated as `np.int32 == np.dtype(int)`.
    if issubdtype(ts, int):
/usr/lib/python3/dist-packages/scipy/misc/pilutil.py:482: FutureWarning:
ion of the second argument of issubdtype from `float` to `np.floating
ated. In future, it will be treated as `np.float64 == np.dtype(float)`.
    elif issubdtype(type(size), float):
Epoch: 0, Step: 0, Loss: 1.18701
Epoch: 0, Step: 0, Loss_val: 1.18701
Model saved in file: ./save/model.ckpt
Epoch: 0, Step: 1, Loss: 1.20592
Epoch: 0, Step: 2, Loss: 1.23162
Epoch: 0, Step: 3, Loss: 1.23711
Epoch: 0, Step: 4, Loss: 1.21384
Epoch: 0, Step: 5, Loss: 1.17371
```

```
Armbian - orangepi@orangepilite: ~/autonomousC
File Edit View Terminal Tabs Help
Epoch: 12, Step: 4, Loss: 0.32317
Epoch: 12, Step: 5, Loss: 0.276
Epoch: 12, Step: 6, Loss: 0.234383
Epoch: 12, Step: 7, Loss: 0.370988
Epoch: 12, Step: 8, Loss: 0.261214
Epoch: 12, Step: 9, Loss: 0.259528
Epoch: 12, Step: 10, Loss: 0.221162
Epoch: 12, Step: 10, Loss_val: 0.250309
Epoch: 12, Step: 11, Loss: 0.242677
Epoch: 12, Step: 12, Loss: 0.206282
Epoch: 12, Step: 13, Loss: 0.218454
Epoch: 12, Step: 14, Loss: 0.274882
Epoch: 12, Step: 15, Loss: 0.265156
Train Accuracy: 0.91
Validation Accuracy: 0.88
begin: 2019-02-19_11-40-06
end: 2019-02-19_13-25-15
Run the command line:
--> tensorboard --logdir=./logs --port=6006
Then open http://0.0.0.0:6006/ into your web browser
orangepi@orangepilite:~/autonomousCar/lesson4$
```

6

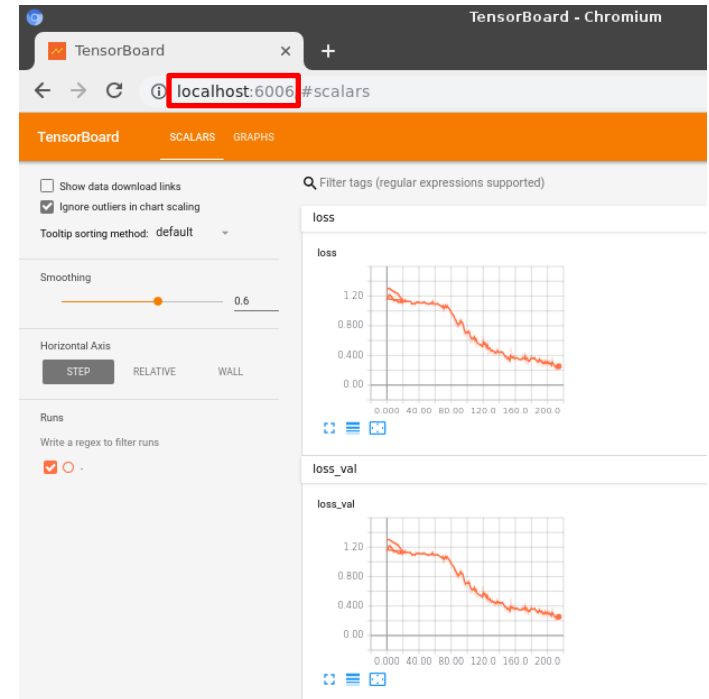
인공지능 훈련하기

인공지능 훈련상태 관찰

- tensorboard --logdir=./logs --port=6006 실행

```
Armbian - orangepi@orangepilite: ~/autonomousCar/lesson4
File Edit View Terminal Tabs Help
orangepi@orangepilite:~/autonomousCar/lesson4$ tensorboard --logdir=./logs --port=6006
w0219 13:30:15.233232 Reloader tf_logging.py:121] Found more than one graph event per run, or there was a metagraph containing a graph_def, as well as one or more graph events. Overwriting the graph with the newest event.
w0219 13:30:15.238102 Reloader tf_logging.py:121] Found more than one metagraph event per run. Overwriting the metagraph with the newest event.
TensorBoard 1.6.0 at http://orangepilite:6006 (Press CTRL+C to quit)
```

- 브라우저에서 아래 내용 실행
localhost:6006



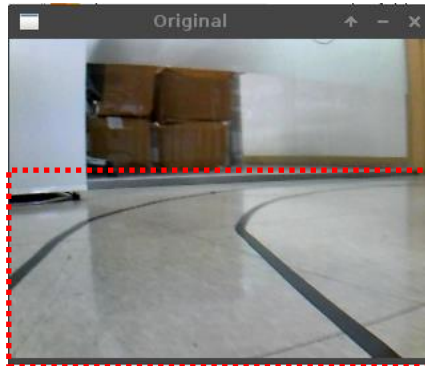
7

자율주행 테스트 및 추가 훈련하기

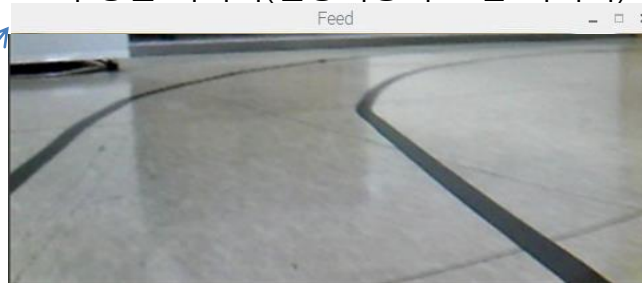
훈련 검증하기

- python3 simulate.py 실행

원본 이미지(카메라 이미지)



수정된 이미지(인공지능이 보는 이미지)

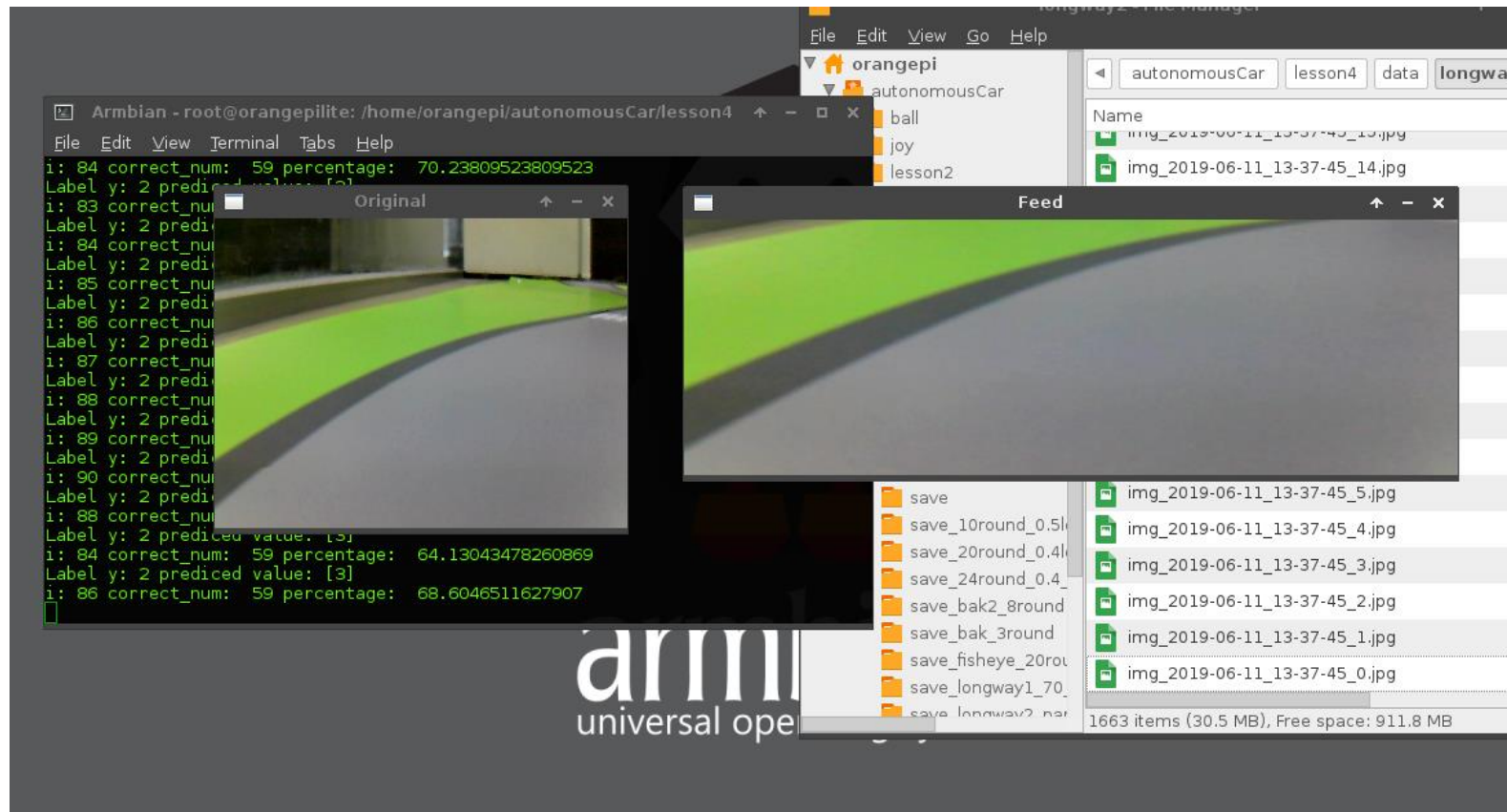


```

Armbian - orangepi@orangepilite: ~/autonomousCar/less
File Edit View Terminal Tabs Help
Label y: 1 prediced value: [1]
i: 1295 correct_num: 1128 percentage: 87.10424710424711
Label y: 1 prediced value: [1]
i: 1296 correct_num: 1129 percentage: 87.1141975308642
Label y: 1 prediced value: [1]
i: 1297 correct_num: 1130 percentage: 87.12413261372399
Label y: 1 prediced value: [1]
i: 1298 correct_num: 1131 percentage: 87.13405238828967
Label y: 1 prediced value: [1]
i: 1299 correct_num: 1132 percentage: 87.14395688991532
Label y: 1 prediced value: [1]
i: 1300 correct_num: 1133 percentage: 87.15384615384615
Label y: 1 prediced value: [1]
i: 1301 correct_num: 1134 percentage: 87.16372021521906
Label y: 1 prediced value: [1]
i: 1302 correct_num: 1135 percentage: 87.17357910906298
    
```

7

자율주행 테스트 및 추가 훈련하기



7

자율주행 테스트 및 추가 훈련하기

훈련 검증하기

- python3 train_analysis.py 실행

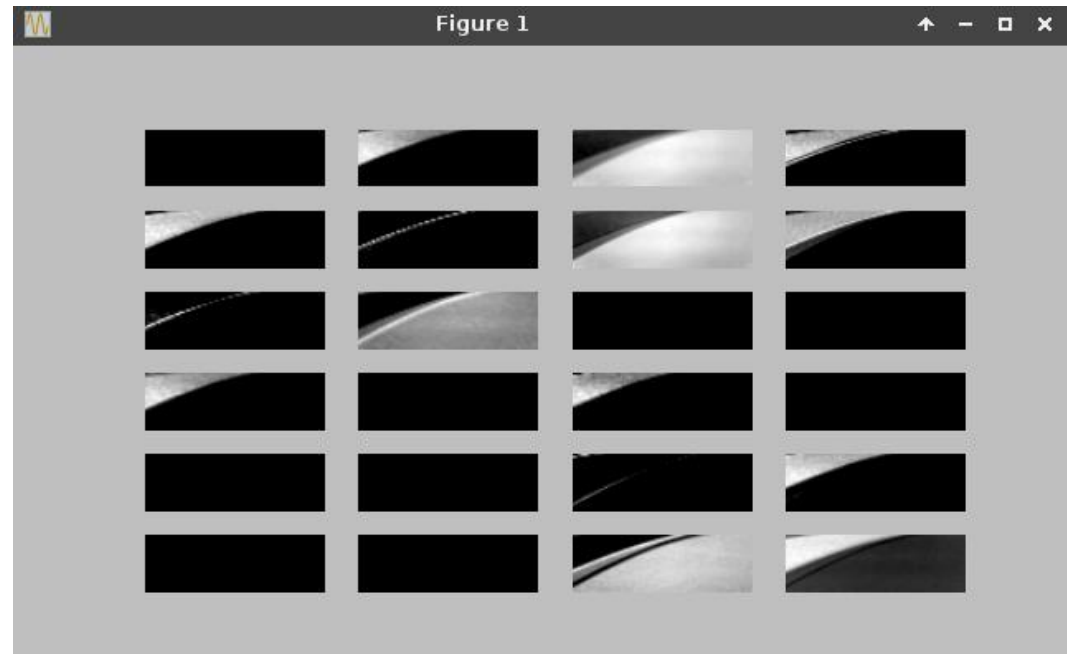
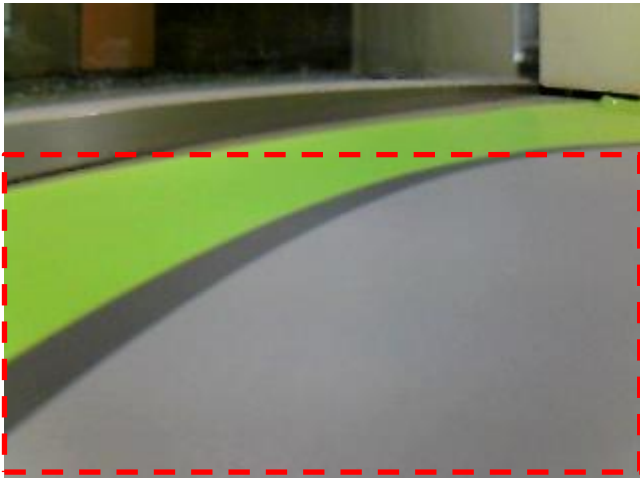
```
Armbian - orangepi@orangepilite: ~/autonomousCar/lesson4
File Edit View Terminal Tabs Help
orangepi@orangepilite:~/autonomousCar/lesson4$ python3 train_analysis.py
model read
3642
/usr/lib/python3/dist-packages/scipy/misc/pilutil.py:479: FutureWarning: C
ion of the second argument of issubdtype from `int` to `np.signedinteger`
recated. In future, it will be treated as `np.int32 == np.dtype(int).type`
    if issubdtype(ts, int):
/usr/lib/python3/dist-packages/scipy/misc/pilutil.py:482: FutureWarning: C
ion of the second argument of issubdtype from `float` to `np.floating` is
ated. In future, it will be treated as `np.float64 == np.dtype(float).type`
    elif issubdtype(type(size), float):
i: 3642 correct_num: 3063 percentage: 84.10214168039539
left_num: 1320 correct_left: 1320 percentage: 100.0
forward_num: 998 correct_forward: 435 percentage: 43.6
right_num: 1320 correct_right: 1308 percentage: 99.1
orangepi@orangepilite:~/autonomousCar/lesson4$
```

7

자율주행 테스트 및 추가 훈련하기

Feature Map 확인

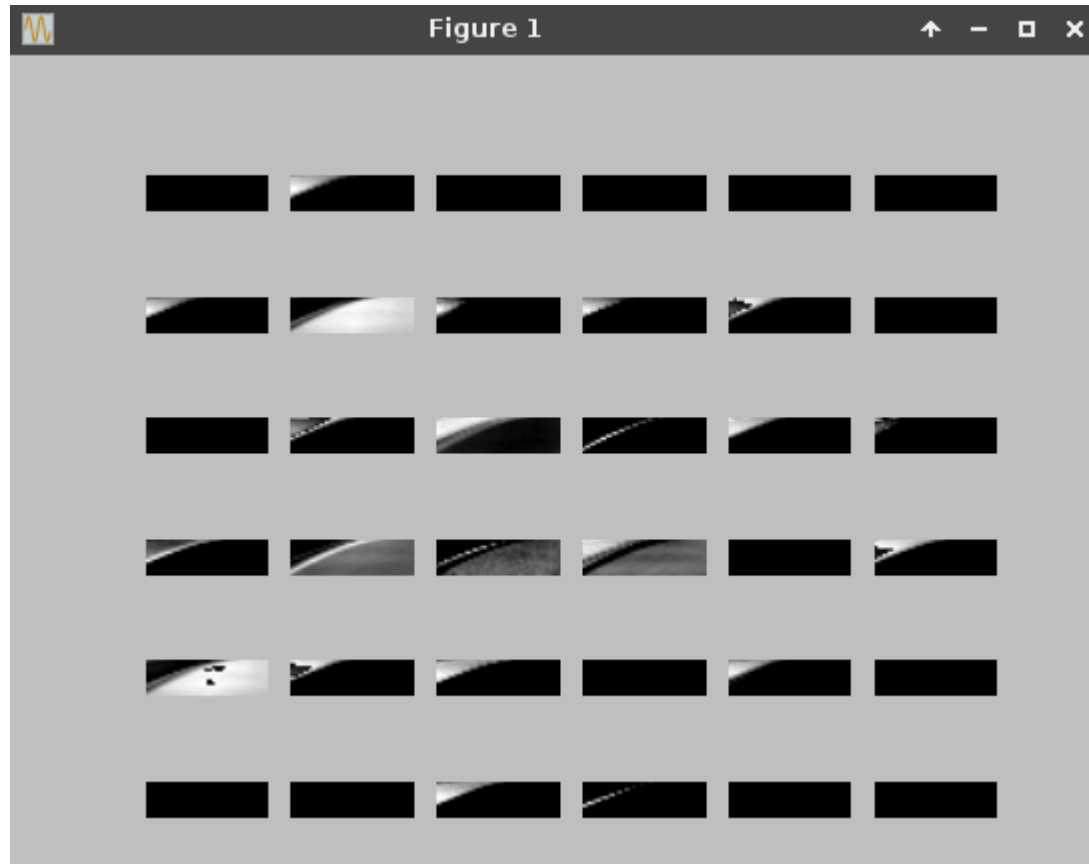
- python3 feature_view.py 실행



7

자율주행 테스트 및 추가 훈련하기

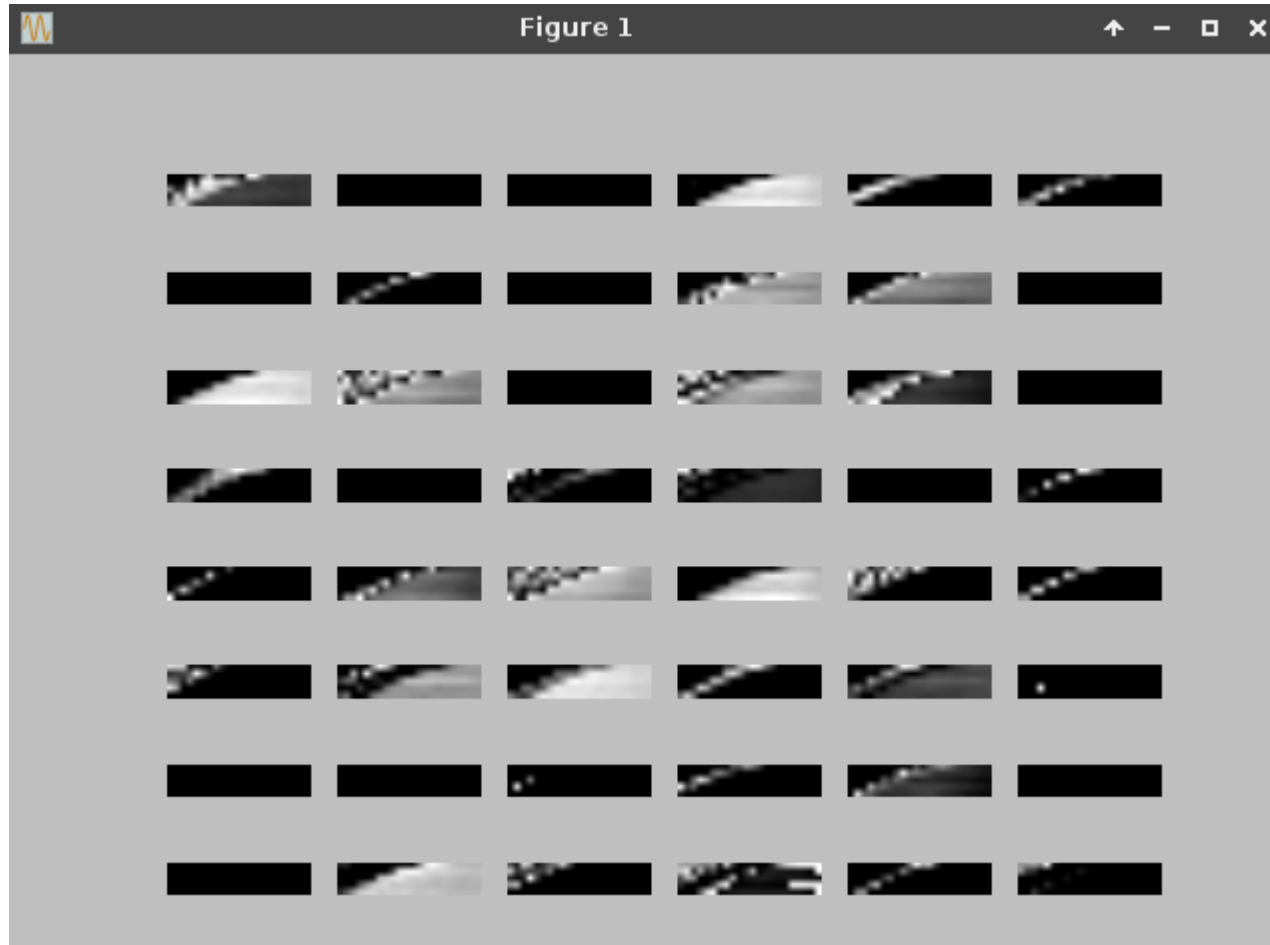
Feature Map 확인



7

자율주행 테스트 및 추가 훈련하기

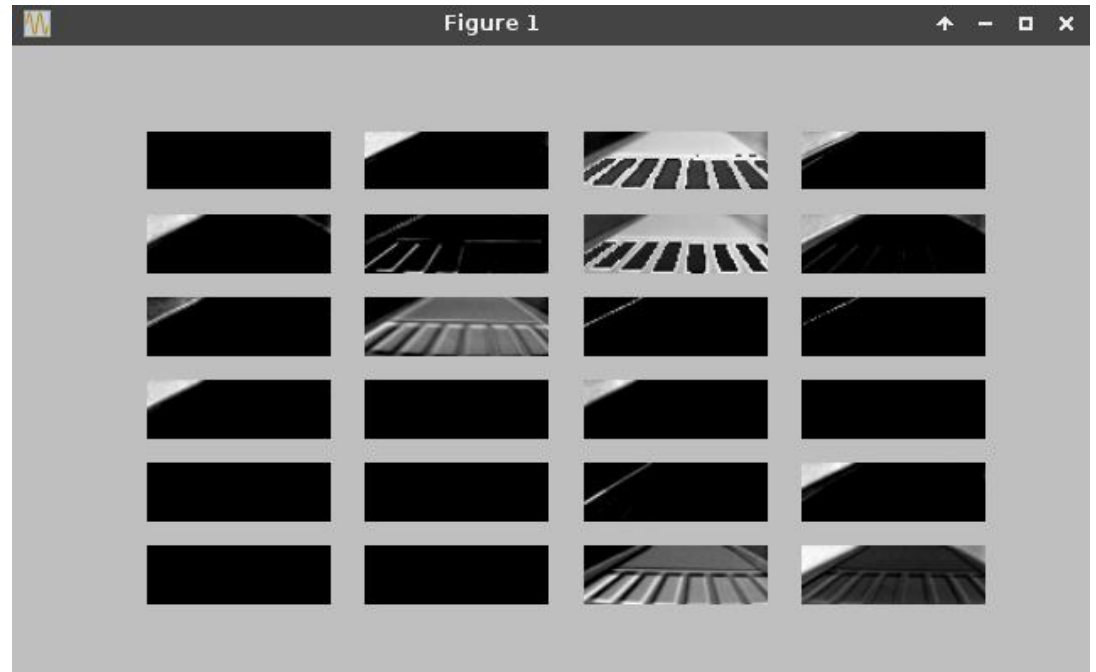
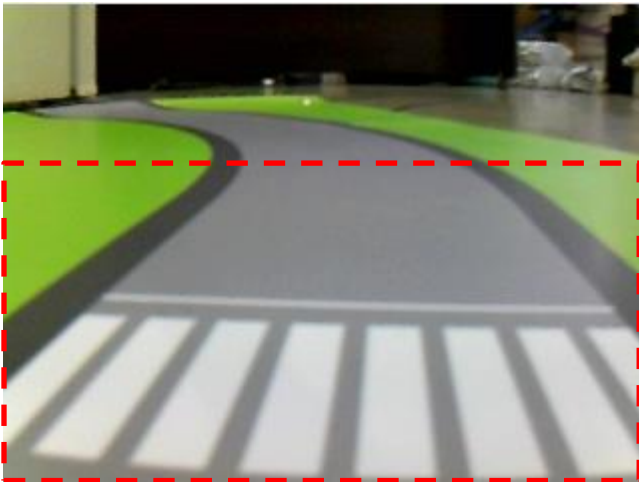
Feature Map 확인



7

자율주행 테스트 및 추가 훈련하기

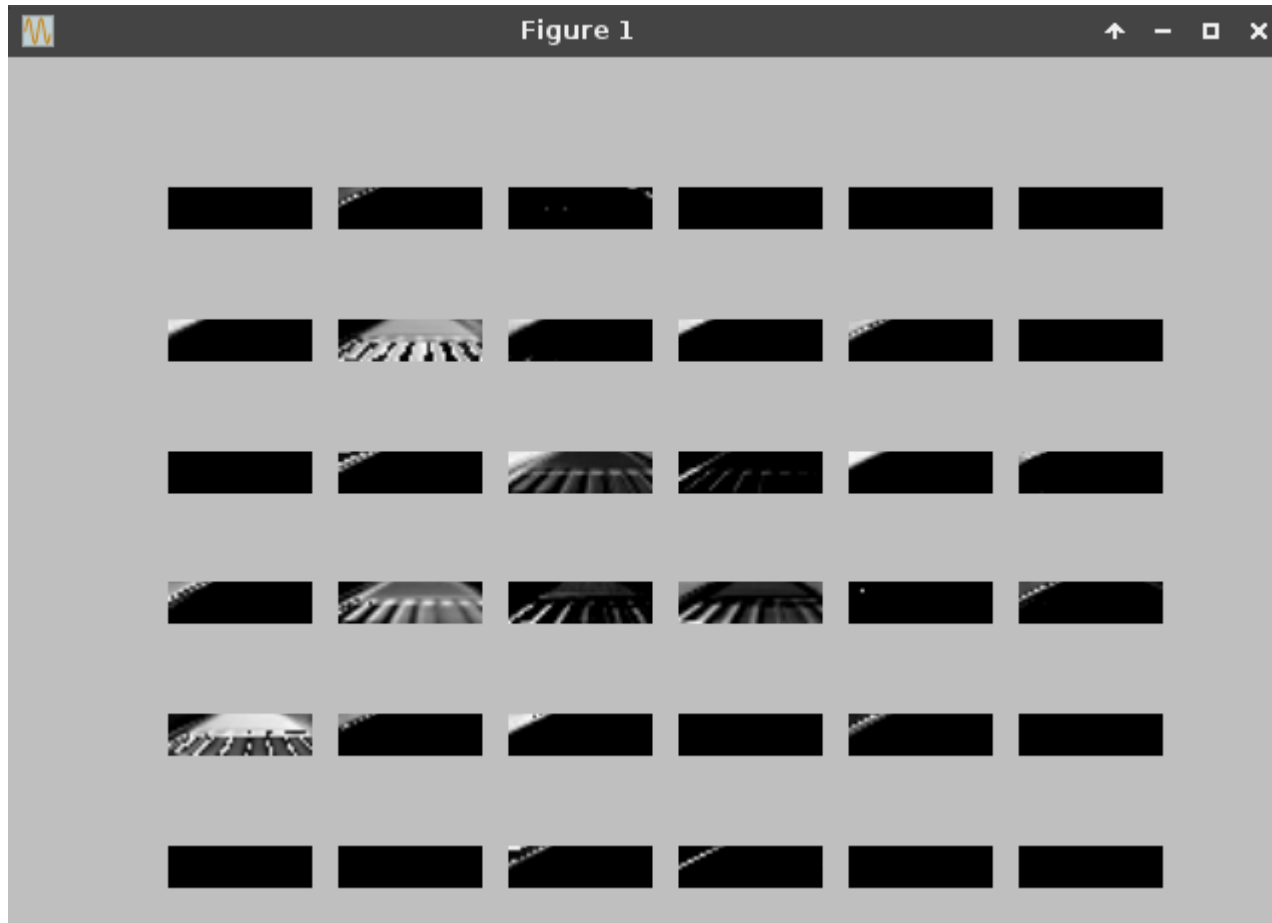
Feature Map 확인



7

자율주행 테스트 및 추가 훈련하기

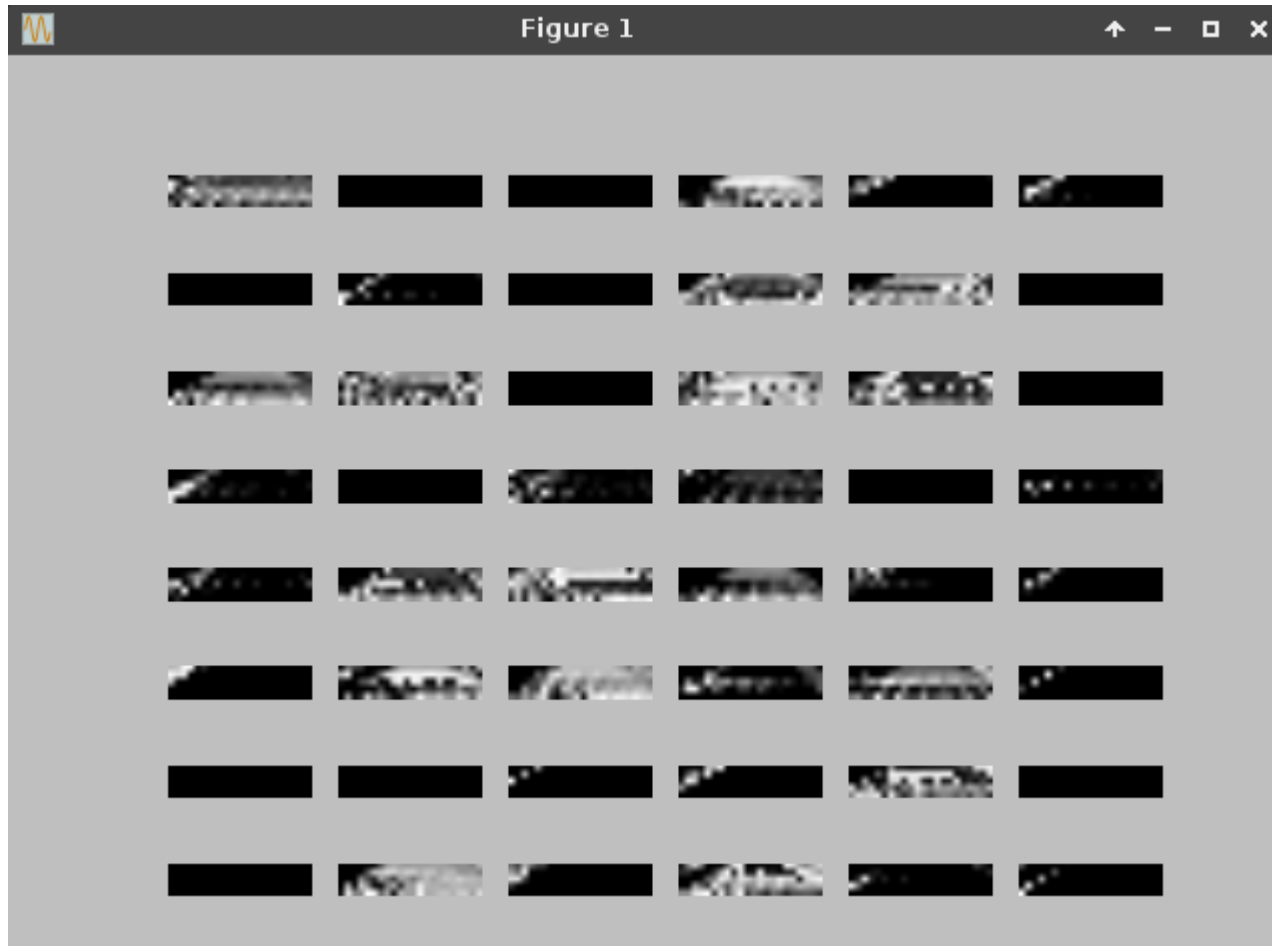
Feature Map 확인



7

자율주행 테스트 및 추가 훈련하기

Feature Map 확인



자율주행 확인하기

- python3 airun.py 실행
키보드에서 a키를 누르면 인공지능 자율주행이 시작됩니다.
다시 a키를 누르면 정지 상태가 됩니다.

자동차가 길을 벗어나는 경우가 발생하면 훈련을 더 시키거나 데이터를 더 모은 후 추가 훈련을 시켜야 합니다.

만약 추가 데이터를 모아도 잘 안 되는 경우는 데이터 정제작업이 안된 경우일 수 있습니다. 이 경우는 처음부터 데이터를 다시 모아야 합니다. 즉, 나쁜 데이터가 포함된 경우는 인공지능이 학습을 잘하지 못합니다.

