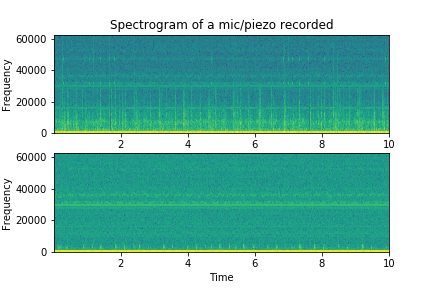
**10.11(Fri)**

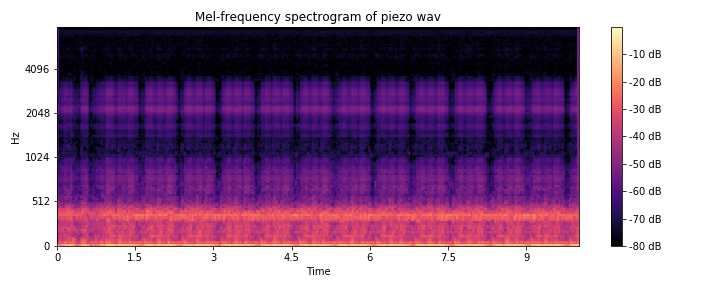
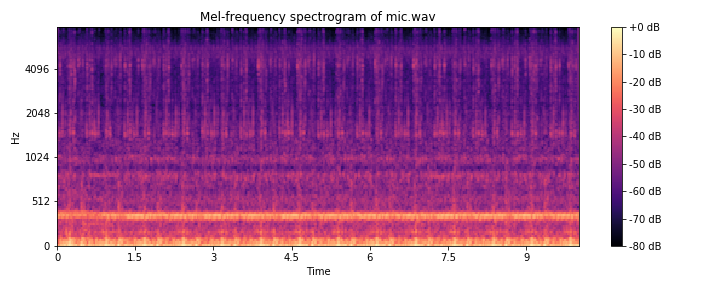
2PM ~ 5PM

* Collected asphalt dataset
* Implementing code for different spectograms

Basic spectogram(top: mic below: piezo)



Mic has much more noise when heard with ears



<Log-mel spectogram>

이번주 목표

- 1차 데이터 수집 완료(아스팔트, 시멘트 3,5,8,10,15,20km 각 10개씩 +a)

- baseline 코드 작성, training까지 해보기 (log mel spectogram, cnn)

10.28 (Mon)

시멘트 데이터 수집

* 3km, 5km, 8km, 10km, 15km(14km), 20km(21km) 각 속력별로 10초 \* 10개씩 데이터 수집
* Sampling rate: 125kS/s, analog signal
* Piezo / mic signal 각각 수집
* Mic의 경우 noise 심하고 변수(킥보드 종류마다 모터소리 다름, 실험환경과 외부 환경 너무 다름 등)가 너무나 많기 때문에 piezo 가 더 적합하다 생각

10.29 (Tue)

Train data 생성 코드 작성 중

* 10초짜리 데이터를 2초짜리로 분할 -> 0-2s, 1-3s, … , 8-10s 로 10초짜리 1개당 9개의 data 생성
* Csv -> Wav 파일 변환
* Wav -> 스펙토그램 이미지(.png) 파일로 변환

10.30(Wed)

Train data generation 코드 작성 완료

10.31(Thur) - 11.1(Fri)

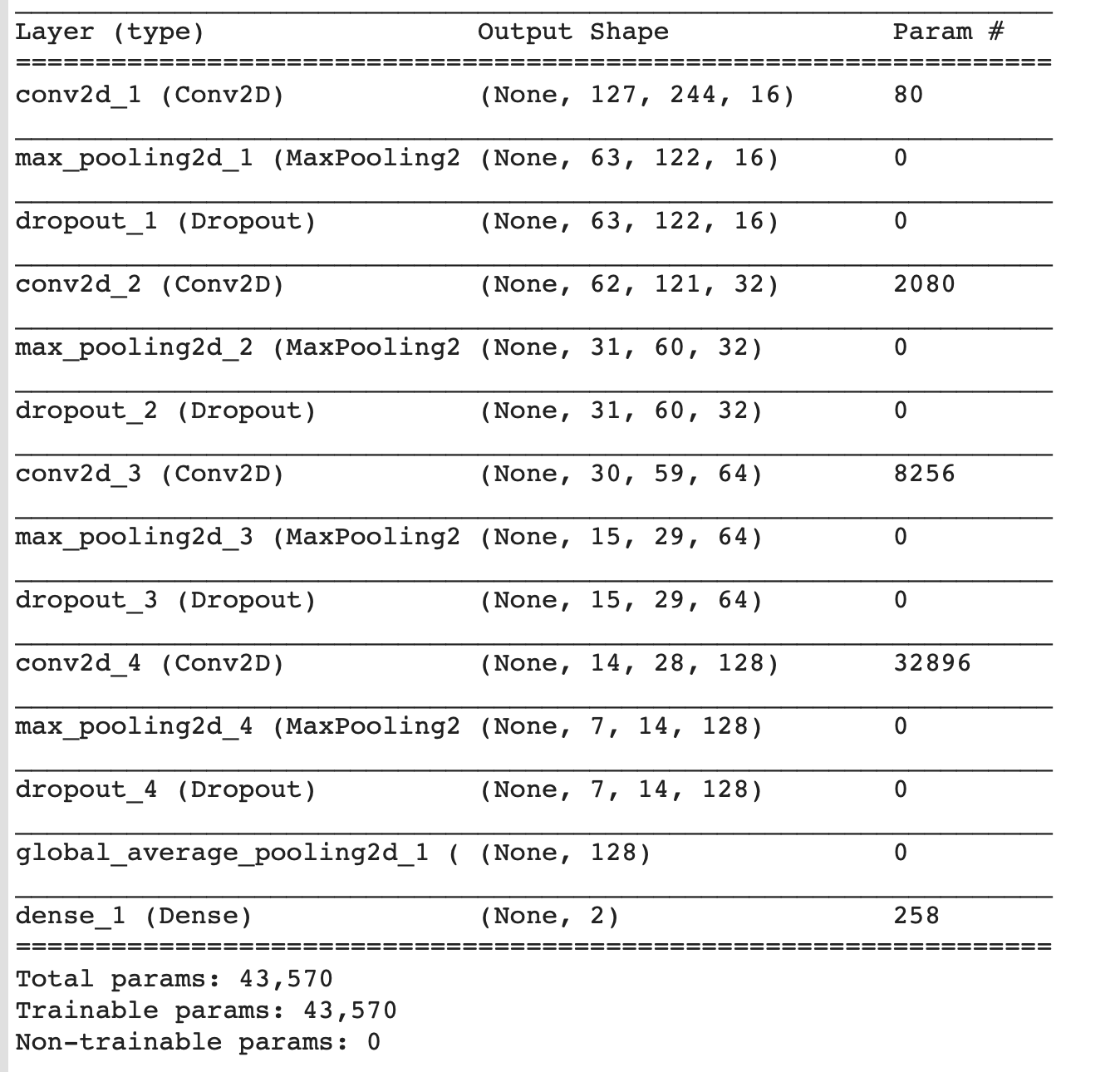
Implementing model code, 아스팔트 데이터 수집 1/3

11.4 (Mon)

* 아스팔트 데이터 수집 완료
* 로컬에서 짠 코드 GPU사용을 위해 google colab으로 포팅 중
* Baseline 모델 구현 중

11.5(Tue)

* Baseline code 작성 완료



사용한 모델

Colab 코드 링크 : <https://colab.research.google.com/drive/1XE2VWf6U1xUIvn0cdX551mAvFebJW15r>

11.6(Wed)

* 코드 수정중

Train data 3km -> test data 나머지 :

Testing Accuracy: 0.742

Train data 5km -> test data 나머지 :

Testing Accuracy: 0.735

Train data 8km -> test data 나머지 :

Testing Accuracy: 0.751

Train data 10km -> test data 나머지 :

Train data 15km -> test data 나머지 :

Train data 20km -> test data 나머지 :

11.11(mon)

* random speed 데이터 수집
* 코드 정리

11.13(wed)

* 리눅스 환경 세팅
* 비트스코프 연결, 구동해보기
* 리눅스에서 돌아가도록 코트 정리 중

11.15(Fri)

* Virtual box 세팅, virtual box에서 usb가 안됨, anaconda 도 안됨..

11.25

비트스콥 실시간 inference 코드 작성

자율 속도 변환 코드로 테스팅

시멘트 바퀴 부서져서 봉합 작업

11.26

비트스코프와 살라에(로직)으로 샘플링한 데이터의 형태가 달라서인지 비트스코프로 inference를 하면 제대로 프리딕션을 못함. 데이터 정규화를 시켜보았으나 효과 없음

11.27

비트스코프로 데이터 모으고 새로 트레이닝 해야 할 것 같다.

비트스코프에서 트레인 데이터 수집을 위한 코드 작성중

스피드 컨트롤러 동작을 안함

11.29

데이터수집 - 비트스코프를 이용한 실시간 수집/ sinosoidal

Cement/asphalt 각각 2800개씩(각1초)

컴퓨터 멈춤 현상이 잦아졌어요 ㅜㅜ 아마 비트스코프로 큰 데이터 받기 &모델 트레이닝 등 무거운 일을 하면 갑자기 멈추는 것 같습니다 => 확인해본 결과 메모리가 부족해서 생기는 문제는 아닌 것 같은데 잘 모르겠네요.. => 그냥 컴퓨터에 문제가 있는 것 같습니다. 코딩만 해도 갑자기 멈춥니다

12.26

코드 포팅 완료

Anaconda, tensorflow, kearas, librosa 등 설치, 코드 포팅

데이터 수집 코드 실행법

1. Anaconda prompt3 관리자권한으로 실행
2. 디렉토리 변경(cd C:/Users/Makinteract/makinteract)
3. *저장할 데이터 타입 설정 : (notepad bitscope\_createdata.py -> 아스팔트면 as, 시멘트면 ce로 바꿔주기)*
4. *실행 (python bitscope\_createdata.py) -> 한번 실행시 300초간 데이터 수집*

12.30

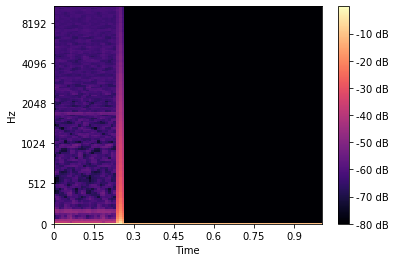
수정사항

1. Command prompt 관리자권한으로 실행(not anaconda)
2. 디렉토리 변경(cd C:/Users/Makinteract/makinteract)
3. 실행 - Python bitscope\_createdata.py
4. a 인지 b인지 입력, file indicator 설정(ce, 1이면 ce 폴더에 ce\_1\_xx.npy로 저장)

[Sampling rate 125k -> 12.5k 으로 낮추고, 개당 2초로 늘림(25k samples)] -> not working due to problems below

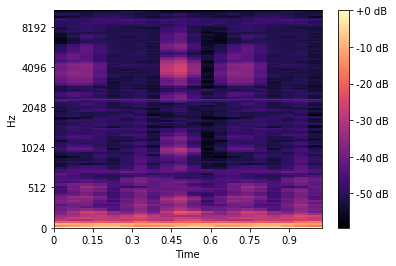
Vibration signal의 경우 고주파가 없기 때문에 sampling rate이 높을 필요가 없고 오히려 높으면 비효율적. 더 긴 timeframe을 가지고 분류하는 것이 정확도 높이는데 도움 될 것.

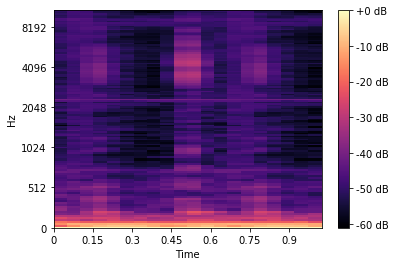
Windows problem(or usb port problem?) - only able to sample up to ~10k samples / sec - think it is sufficient, but don’t know..

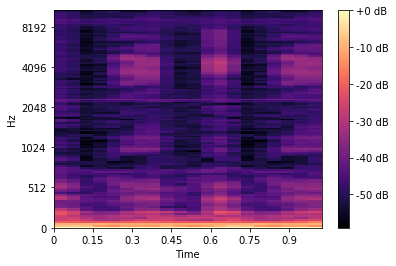


concern) Normalizing - normalize before making spectrogram or not?

Sampling rate 10k, asphalt, 20km/h







1.1

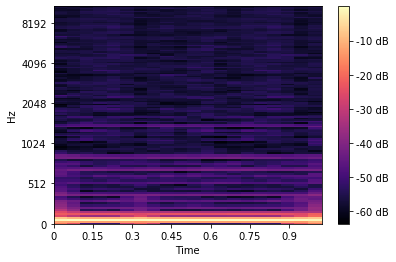
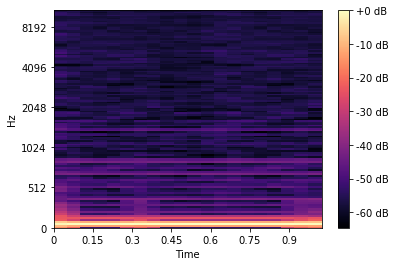
19:00pm ~ 03:00am

Completed implementing GUI for data collection

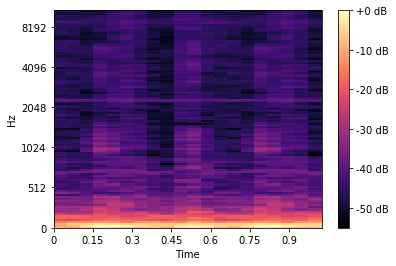
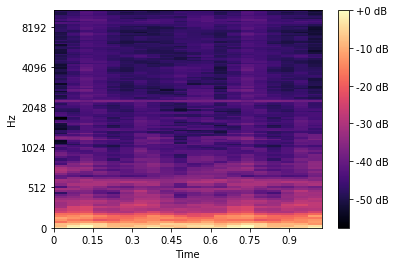
1.2

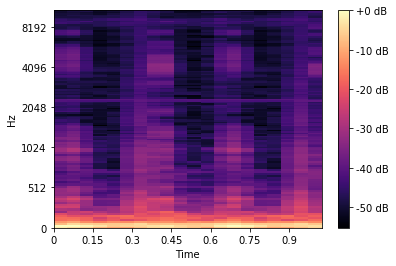
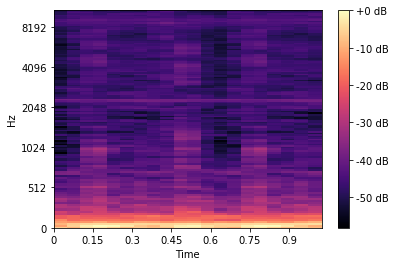
Testing GUI, checking mel spectrogram

**<nothing>**

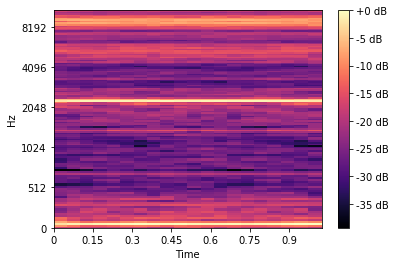
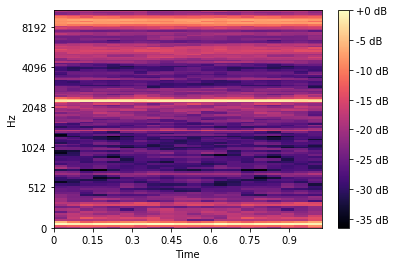


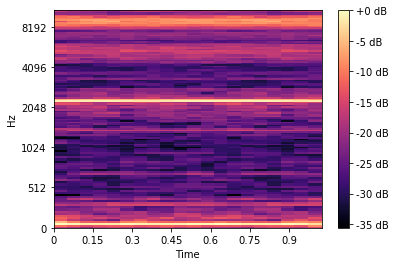
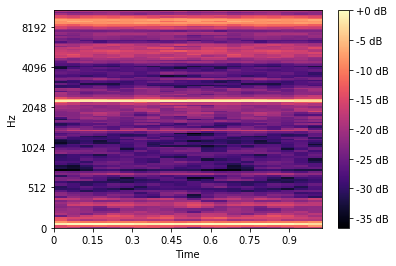
**<asphalt>**





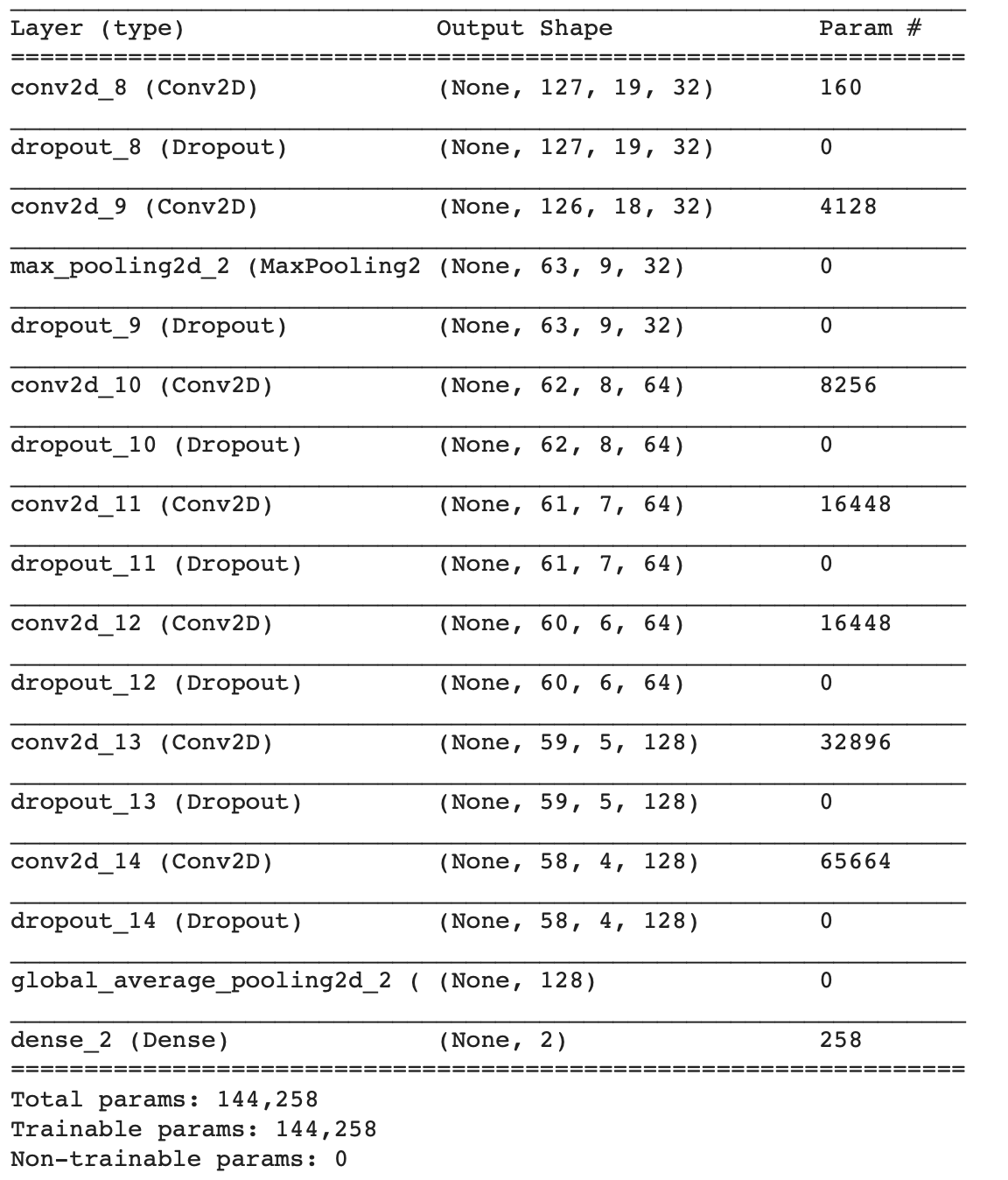
**<cement>**



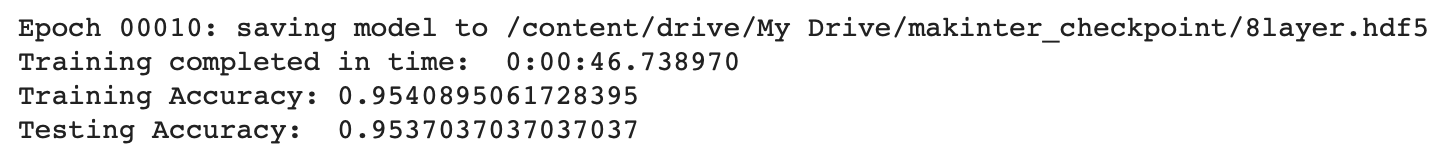


1.15

Model - 128\*20 으로 column dimension이 줄어 그에 맞춰 maxpooling도 한번으로 줄임



Training result



Anaconda and bitscope

Keeps yielding problems - installing tensorflow & keras without anaconda

1/21

Ur / as / block tri model

Data

5700 features for cement

6000 features for asphalt

2520 features for ur

Result

*Epoch 00010: saving model to /content/drive/My Drive/makinter\_checkpoint/8layer\_tri\_0121.hdf5*

*Training completed in time: 0:00:49.969755*

*Training Accuracy: 0.8607594936708861*

*Testing Accuracy: 0.8456399438250417*

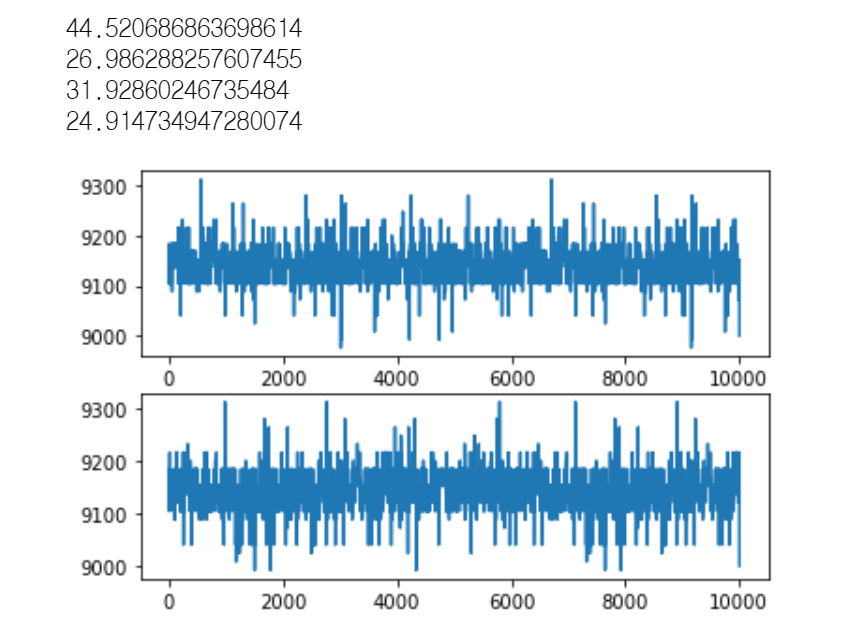
Epoch 00012: saving model to /content/drive/My Drive/makinter\_checkpoint/8layer\_tri\_0121.hdf5

Training completed in time: 0:00:58.827336

Training Accuracy: 0.8728023909985936

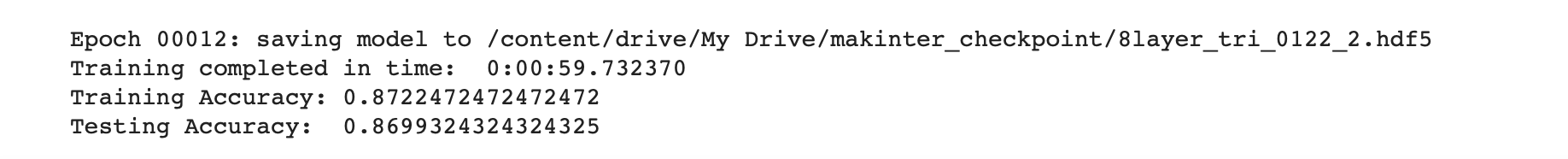
Testing Accuracy: 0.8604078761468289

Default state recording



Recording at steady state results in noise above(and the standard deviation)

1/22 최종 데이터셋 결과

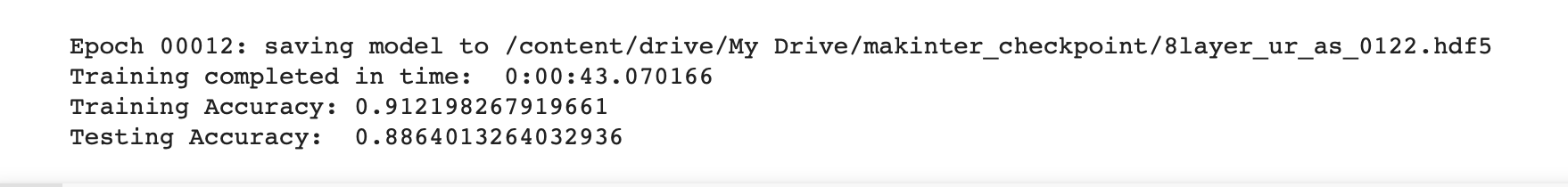


아스팔트와 우레탄이 잘 구분이 안됨

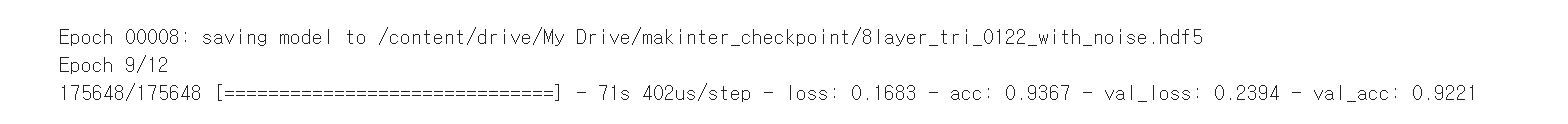
2 label classification 비교시

아스팔트-블락: ~96% accuracy

아스팔트-우레탄: ~88.6% accuracy



With gaussian noise



Accuracy improvement => 92%

But doesn’t work well, overfitting?, too many ur data

1/23 0123\_with\_noise

Accuacy 90%

Only 0,01 noise, reduce dataset to half

주행 결과

* 주행 특성 떄문에 생기는 결과: 매끄럽게 쭉 달리면 우레탄이라고 인식
* 데이터 수 불균형: block data 가 작은데, 여기에 overfitting 되어 block이 아예 안나타남

Train\_0129 data: new data tri without noise

len train x: 15968

len train y: 15968

Asphalt 5400

Cement 5130

Ur 5454

Epoch 00005: saving model to /content/drive/My Drive/makinter\_checkpoint/8layer\_tri\_0129.hdf5

Training completed in time: 0:00:34.763427

Training Accuracy: 0.8287199398797596

Testing Accuracy: 0.8135304145680157

(epoch 5)

8layer\_as\_ce\_0130.hdf5

As ce final model

len train: 10519

len test : 1169

Training Accuracy: 0.9196691700335362

Testing Accuracy: 0.9144568006843456

8layer\_as\_ce\_v2\_0130.hdf5 - added data for faster speed in cement

Epoch 00010: saving model to /content/drive/My Drive/makinter\_checkpoint/8layer\_as\_ce\_v2\_0130.hdf5

Training completed in time: 0:00:44.526304

Training Accuracy: 0.929129515585955

Testing Accuracy: 0.913601368691189

8layer\_as\_ce\_v3\_0130.hdf5

Epoch 00010:

Training completed in time: 0:00:45.537969

Training Accuracy: 0.9094585898617455

Testing Accuracy: 0.8930710009064197

<3 state>

Epoch 00010: saving model to /content/drive/My Drive/makinter\_checkpoint/8layer\_tri\_0131.hdf5

Training completed in time: 0:01:05.174114

Training Accuracy: 0.8871807435645749

Testing Accuracy: 0.8595793059497072

data:

Asphalt 5400

Ce 5490

Ur 5454

len train x: 16327

len train y: 16327

<최종 결산>

2개 짜리: 8\_layer\_as\_ce\_v3\_0130.hdf5 가 최종모델

3개 짜리: 8\_layer\_tri\_v2\_0131.hdf5 가 최종 모델

Gaussian noise를 더한 데이터로 트레인 한 경우 training & testing accuracy는 높게 나왔지만(3 way 도 93% 정도) 실제 주행에서는 굉장히 안좋은 성능이 나타남. Noise를 랜덤으로 추가해서 데이터 수를 10배로 키웠는데 아마도 noise의 variance가 너무 작아서 데이터가 매우 유사 했기 때문에 정확한 원인은 모르지만 overfitting이 일어났을 수도 있겠다 판단함. 결론적으로 gaussian noise를 더한 데이터셋은 사용하지 않음.

frequency 를 잡아내는 spectrogram의 특성상 진동수가 매우 중요. 따라서 아스팔트와 우레탄 같이 진동에 큰 차이가 없는 도로의 경우 도로에 따른 차이 보다는 속도에 따른 차이를 도로의 특성으로 인식함. 예를 들어 주로 차도인 아스팔트에서는 주로 빠른 속도로 주행하고, 방지턱에 따른 급정거 하는 특성이 있기 때문에 데이터에서도 그러한 쪽으로 특성이 편향되어있음. 향후 모델을 더 발전시키기 위해서는 주행 속도와 특성이 균등한 좋은 데이터셋이 필요. 또 아스팔트와 우레탄의 미세한 차이를 잡아내기 위해서는 더 큰 sampling rate으로 데이터를 수집해보고, melspectrogram 이외에 다른 방식으로도 데이터를 가공해볼 수 있을 것 같음.