JSR프로젝트 - Nvidia의 End-to-End Deep Learning을 바탕으로 구현

1. **Model - CNN(Convolutional Neural Network)**

**CNN model :**

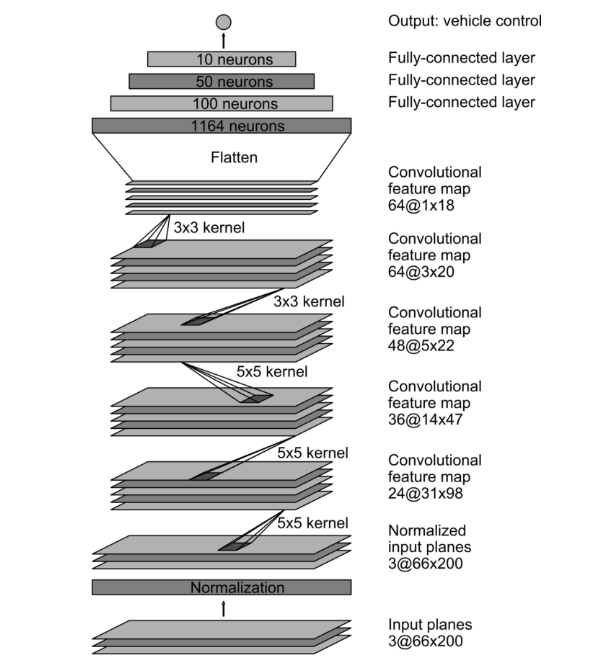
* Filter를 이용하여 합성곱 연산 수행
* 필터에 따라 이미지의 다른 특징을 추출

그림 1. CNN architecture

(출처 : <https://developer.nvidia.com/blog/deep-learning-self-driving-cars/>)

* Pytorch 1.4.0 버전 환경 (<https://pytorch.org/docs/1.4.0/nn.html>)
* **텍스트이(가) 표시된 사진

  자동 생성된 설명convolution**

각 채널의 크기에 맞게 in channel과 out channel 설정

텍스트이(가) 표시된 사진

자동 생성된 설명

위의 계산 과정을 거쳐 66x200에서 31x98 순서로 나오도록 stride를 설정

* **Flatten**

텍스트이(가) 표시된 사진

자동 생성된 설명

최종적으로 output이 1이 될 때까지 layer를 펴주는 작업

* **Forward**

텍스트이(가) 표시된 사진

자동 생성된 설명

X가 들어오면 convolution을 먼저 통과하고, torch.flatten 함수를 통해 tensor([[1,2,3], [4,5,6]])형식의 배열로 바꾼 뒤 flatten을 통해 최종 output을 도출

1. **Image normalization**

텍스트이(가) 표시된 사진

자동 생성된 설명

확장자가 .mkv인 파일을 읽어 프레임을 읽는다.

텍스트이(가) 표시된 사진

자동 생성된 설명

동영상으로부터 프레임을 얻으면 image 크기 720, 1280이고, 이것을 cnn모델의 input 형식(66x200)에 맞추어 (200, x) 크기로 줄여준다.

1280 : 720 = 200 : x

X=112.5(112)

Resize를 통해 현재 이미지 크기를 200, 112크기로 줄인다.

112-66= 46이므로 66부분을 버린 나머지 부분을 ROI로 설정

ROI영역으로 정리된 이미지를 image 폴더에 저장한다.

텍스트이(가) 표시된 사진

자동 생성된 설명

영상과 csv파일을 랜덤 이름으로 바꾼다.

1. **Train**

* 텍스트이(가) 표시된 사진

  자동 생성된 설명Google colab으로 학습

텍스트이(가) 표시된 사진

자동 생성된 설명image폴더에서 이미지를 읽고 학습된 파일을 save 폴더에 저장.

csv파일에서 현재 이미지 프레임 번호와 해당 노출 값을 읽어 csv\_data에 저장

텍스트이(가) 표시된 사진

자동 생성된 설명

Batch 크기는 100마다 지정하고 epoch가 2000이 될 때까지 학습시킴

Loss – mean squared error 구하기 위해 사용(예측 값과 타겟 값 사이의 에러)

nn.MSELoss()를 통해 측정

torch.optim.Adam(net.parameters(), lr=le-4)

parameters : 매개 변수 그룹을 최적화 하거나 정의하기 위한 매개 변수 반복

lr : 학습률 (기본 값 le-3)

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명2000번이 될 때까지 학습시키며, 매번 csv파일을 랜덤으로 섞는다.

Tensor type으로 x\_batch, y\_batch를 device에 올려주고 outputs으로 출력 값 도출

학습 수행 전 미분 값을 0으로 초기화 하고 네트워크 연산 결과와 실제 결과를 cross entropy 연산하여 손실함수 값 도출

가중치 w와 편향 b에 대한 기울기 계산 후 업데이트

텍스트이(가) 표시된 사진

자동 생성된 설명

이미지 폴더에서 이미지를 불러와 YUV형태로 바꾼 뒤 x\_batch에 저장하고 y\_batch에는 csv 값을 저장한다.

텍스트이(가) 표시된 사진

자동 생성된 설명

Epoch가 100만큼 증가할 때마다 학습 파일을 저장한다.

1. **데이터 수집 방법**

텍스트이(가) 표시된 사진

자동 생성된 설명

텍스트이(가) 표시된 사진

자동 생성된 설명현재 usb카메라의 device 이름 지정한다.

텍스트이(가) 표시된 사진

자동 생성된 설명자동차가 주행하면 해당 영상과 csv파일을 동시에 저장한다. 데이터 수집을 위해 여기에서는 “자이카 조이스틱”으로 수동으로 트랙을 주행 시키면서 녹화하였다.

CSV파일을 만들고 현재 시간, 프레임 번호, 해당 프레임의 노출 값을 저장한다.

텍스트이(가) 표시된 사진

자동 생성된 설명

Subscribe를 통해 data를 받으면 거기로부터 image를 추출하고 이 이미지를 바탕으로 다음 과정을 진행한다.

텍스트이(가) 표시된 사진

자동 생성된 설명이미지가 들어오면 해당 프레임 번호를 fre로 세고, cnt 를 노출 값으로 지정하여 5부터 70까지 범위에서 노출 값을 변경해준다. 이후 cnt값을 exposure\_set 함수를 통해 카메라에 세팅한다.

텍스트이(가) 표시된 사진

자동 생성된 설명

cnt값을 받아 v4l2명령어로 디바이스에 직접 명령을 내려 노출도를 바꿔준다.

1. **결과**



학습된 .pth파일을 model을 통해 load 한다.

텍스트이(가) 표시된 사진

자동 생성된 설명

모델을 로드하여 최종 값을 도출해낸다.

텍스트이(가) 표시된 사진

자동 생성된 설명

이후 들어온 이미지를 학습시킬 때의 이미지 크기와 똑 같이 만들어주고 end2end모델을 통해 출력 값을 도출한다. 여기에서 출력 값은 해당 이미지의 노출 값이 된다.



녹화한 영상 중 하나를 input file로 지정해주고 확인한다.

텍스트, 컴퓨터, 전자기기이(가) 표시된 사진

자동 생성된 설명

화면의 밝기가 달라지면 노출 값이 해당 영상에 따라 바뀐다.