**Final-Term Project**

**Implementing Basic Generative Adversarial Network (GAN)**

컴퓨터전자시스템공학부

202030014 곽민호

1. **Term Project 목표**

간단한 Generative Adversarial Network (GAN) 구현 및 분석하기

1. **구현 환경**

|  |
| --- |
| Ubuntu 18.01 LST |
| Visual Studio Code |
| Python 2.7 |
| Numpy 1.12.1  Tensorflow 1.4.0  Matplotlib 2.0.1 |

1. **사전 작업**

* 제공된 Dataset을 Classification에 적합하도록 Label 별로 분리
* [Right, Left, Straight, Up] 순으로 나열
* 원본 영상이 8bit Grayscale에 해당하는 .pgm이미지이다. Keras에서 이러한 이미지를 입력으로 직접 받을 수 있는 포맷은, 동일한 8bit Grayscale을 지원하는 포맷png뿐이므로, 해당 형식으로 .pgm 파일을 변환 한다.  
  Jpg 파일은 grayscale을 지원하지 않아 이미지가 흑백 임에도, RGB에 해당하는 3채널이 입력으로 들어가므로 선택하지 않았다.

1. **ANN 구조 요약**

Metrics = Accuracy, Loss = Cross-entropy

Flatten Layer ( 들어온 이미지를 벡터화 하여 960 x 1 로 만들어주는 층)

Optimizer = SGD

Dense Layer ( Hidden Unit 3개, Activation = Sigmoid )

Dense Layer ( Hidden Unit 4개, Activation = Sigmoid )

Initialization = xavier initialization

스크린샷이(가) 표시된 사진

자동 생성된 설명

* **Assignment 1**

**Analyze a given source code**

* **Assignment 2**

**Draw plot of (1) discriminator and (2) generator loss values with respect to iterations**

**각각의 Iteration에 해당하는 자세한 Loss 값은 Loss.csv 파일에 있습니다.**

* **Assignment 3**

**Show the generated images at 0, 10000, 60000 iteration**

* **Assignment 4**

**Discuss experimental result**