**산학 캡스톤 디자인1**

**Weekly Report #03 - 인공지능 기술을 이용한 영상의 한국어 해설 에이전트**

**팀명 : DD**

**컴퓨터공학과 이신우, 고훈현**

**1. 서론**

WHO가 2014년에 발표한 바에 따르면 전세계 시각 장애인인구가 2억 8500만명이고, 이 중 3900만명이 맹인이며, 2억 4600만명이 저시력자라고 한다. 또, 2015년 연세대 세브란스 병원 시기능연구소에서 발표한 바에 따르면 저시력자 자살률이 정상인의 2배 수준이라고 한다. 이를 통하여 저시력에 의한 정신적 스트레스가 크다는 것을 인지하였고, 저시력자 및 시각 장애인들이 신체적 제약을 극복할 수 있는 서비스가 필요하다는 것을 느꼈다. 이태까지 배워온 전공지식을 활용하여 시각 장애인들의 신체적 제약을 극복할 수 있게끔 영상 음성 해설 웹 서비스를 제작하고자 하였다.

**2. 기존연구**

**2.1 Image Captioning**

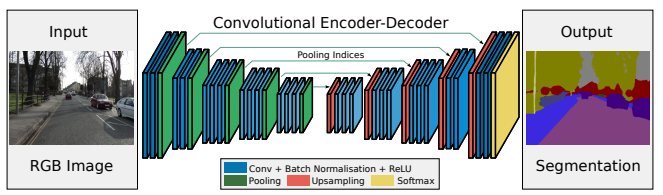
이미지 캡셔닝(Image Captioning)은 이미지를 설명하는 문장으로 변환시켜주는 기술로 인코더(encoder)를 이용하여 이미지의 특징을 추출하고, 추출된 특징들을 디코터(decoder)에 전달하여 워드 벡터 시퀀스를 출력으로 받는다. 출력된 시퀀스들은 단어사전과 매칭되며 최종적으로 사람이 알아볼 수 있는 문장으로 출력된다.

**2.2 Image Segmentation**

영상 분할(Image Segmentation)은 이미지의 각 픽셀이 어떤 클래스에 속하는지 분할하는 것이다. 영상 분할은 시멘틱 분할(Sementic Segmentation)과 인스턴스 분할(Instance Segmentation)으로 나눌 수 있는데, 전자는 픽셀 단위로 객체를 분류하는 것이고, 후자는 객체 탐지(object detection)를 추가하여 각 객체별로 분류한다.

**2.2.1 SegNet**

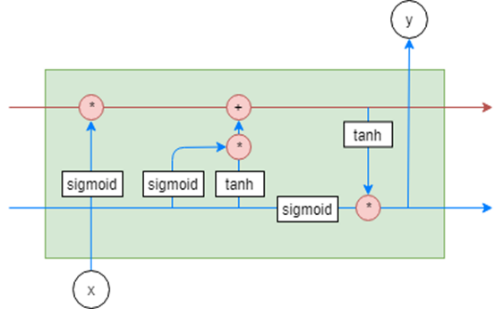
SegNet은 자율주행에서 사용하는 영상 분할 모델이다. 모델은 인코더(Encoder)와 디코더(Decoder)가 결합된 형태이며, 인코더의 네트워크는 VGG16에서 완전연결층(fully connected layer)를 제외한 컨볼루션 레이어를 그대로 사용한다. 디코더의 네트워크는 Upsampling과 컨볼루션 연산을 수행하며, 마지막 레이어에는 softmax 분류기가 배치되어 있다[1].

****

**그림 1** SegNet의 구조

**2.2.2 LSTM**

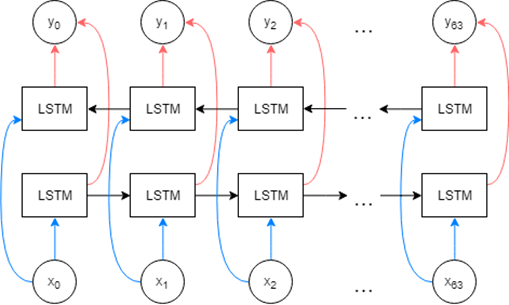
LSTM은 크게 망각 게이트, 입력 게이트, 출력 게이트로 이루어져 있습니다. 망각 게이트는 이전의 셀정보를 망각할지 결정하는 역할을 하고, 입력 게이트를 통해 어떤 값을 셀에 업데이트 할지 결정하는 역할을 하고, 출력 게이트를 통해 무엇을 출력으로 내보낼지 결정하는 역할을 한다.

****

**그림 2** LSTM의 구조

**2.2.3 양방향 LSTM**

양방향 LSTM은 LSTM의 성능을 더욱 향상시키기 위해, 역방향 LSTM계층을 추가한 모델이다.

****

**그림 3** 양방향 LSTM의 구조

**2.3 Visual Attention**

Visual Attention은 이미지의 중요한 부분에 집중하고 더 자세히 묘사하는 방법으로 Top-Down Approach와 Bottom-Up Approach의 장점을 합친 것이다.

**2.4 End-To-End Speech Recognition**

입력 음성으로부터 하나의 통합 신경망을 거쳐 문자열 혹은 단어열을 인식하는 음성인식 방법을 뜻한다.

**2.4.1 Tacotron2**

Tacotron2는 End-to-End 음성 합성 모델 중 하나로, 모델은 크게 Embedding, Encoder, Decoder, 후처리부분으로 나뉘어 있다. 먼저 임베딩을 통해 문자를 벡터화 하고 인코더에서 벡터의 특징을 추출한 뒤, 디코더에서 mel-spectogram을 생성한다. 만들어진 mel-spectogram을 보코더에서 후처리하여 wav파일 형식으로 만들어낸다[2].

**스크린샷, 시계이(가) 표시된 사진

자동 생성된 설명**

**그림 4** Tacotron2의 구조

**2.4.2 Griffin Lim Vocoder**

Griffin Lim은 Spectogram을 음성으로 만들어주는 알고리즘이다[3].

**2.2 데이터셋 조사**

**2.2.1 MS COCO**

MS COCO 데이터셋은 마이크로소프트 리서치에서 공개한 이미지 캡셔닝 데이터 베이스로 총 33만개의 이미지로 구성되어 있으며 각 이미지마다 해상도 정보와 사람이 직접 작성한 5개의 문장과 이미지 세그멘테이션 정보도 제공한다.

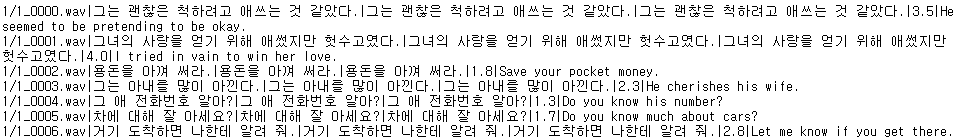
**잔디, 테이블, 여자, 장난감이(가) 표시된 사진

자동 생성된 설명**

**그림 5** MS COCO 데이터셋의 샘플 이미지

**2.2.2 KSS**

KSS 데이터셋은 약 12시간 한국인 성우 음성파일과 대사로 이루어진 데이터셋으로, Kaggle에서 공개 중인 단일화자 한국어 음성 데이터 셋이다. 총 12시간 분량의 음성파일로 구성되어 있으며 음성파일에 대한 한국어 스크립트파일을 제공한다.

****

**그림 6** KSS 데이터셋의 스크립트 파일

**3. 제안 방법**

**3.1 문제 정의**

정지 영상에 대한 음성 해설 모델을 만들어 웹 어플리케이션으로 데모 프로그램을 제공하고자 한다.

**3.2 자료 수집 및 선별**

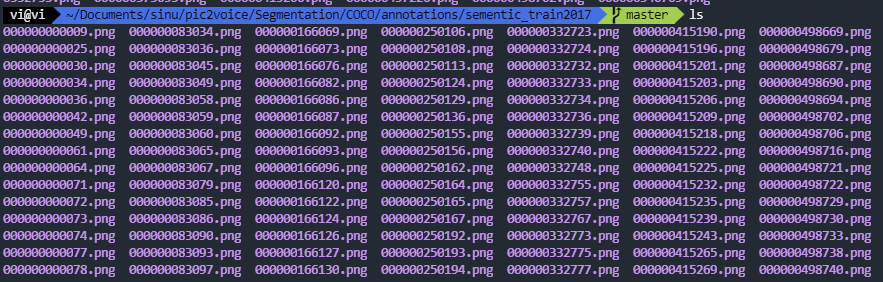
**3.2.1 MS COCO 전처리**

세그멘테이션 모듈을 학습하기 전에 세그멘테이션 마스크가 필요하기 때문에 MS COCO 이미지들과 이미지에 해당하는 세그멘테이션 정보를 가지고 세그멘테이션 마스크를 생성하였다. 그림 7에서 왼쪽의 이미지를 가지고 세그멘테이션 마스크를 생성하면 오른쪽의 그림과 같다. 각 클래스의 구별은 마스크 이미지의 밝기(0~90)로 구분한다.

****

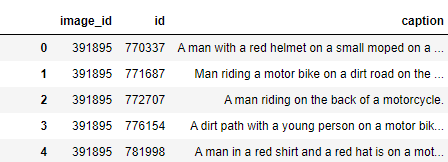
**그림 7** 세그멘테이션 마스크 생성

훈련 데이터와 검증 데이터에 대한 세그멘테이션 마스크를 생성하였고 그 중 일부의 목록은 그림 8에서 보여주고 있다.

****

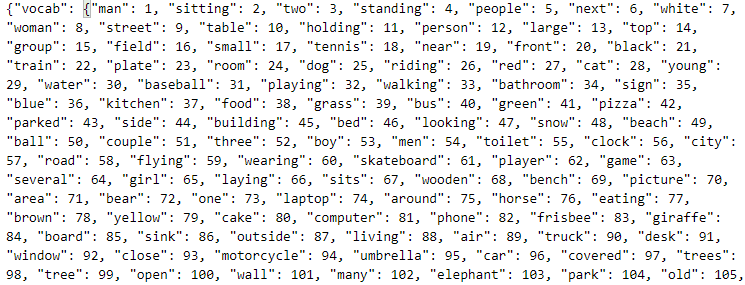
**그림 8** MS COCO 데이터셋을 전처리한 파일 목록 중 일부

이미지 캡션 데이터는 추출하여 csv파일로 저장하였고 그 중 일부는 그림 9에서 확인할 수 있다.



**그림 9** 추출된 캡션 데이터 중 일부

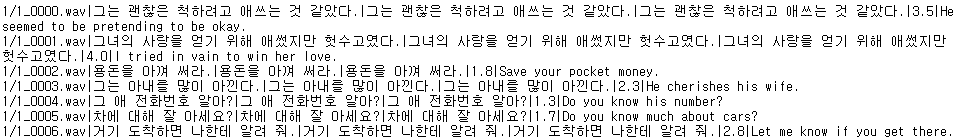
워드 임베딩 작업에 필요한 단어 사전을 제작하기 위하여 추출된 캡션을 가지고 단어 사전을 제작하였고 그 중 일부는 그림 10에서 보여주고 있다.

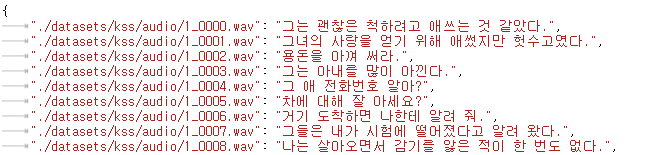


**그림 10** 제작된 단어 사전 중 일부

**3.2.2 KSS 데이터 전처리**

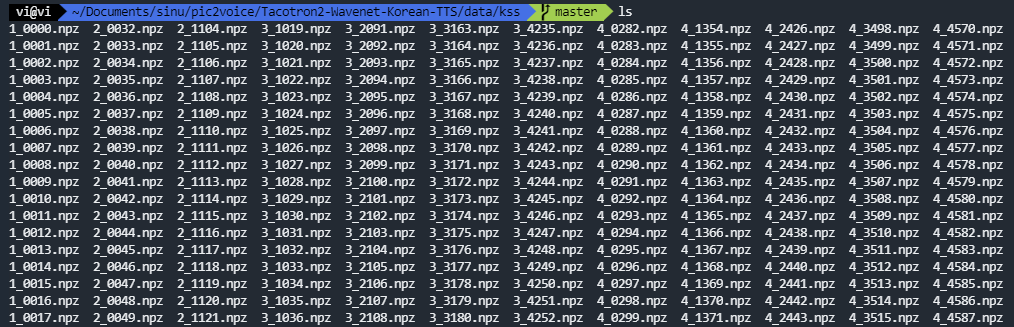
Tacotron2에서 스크립트 파일은 json 형식으로 받기 때문에 KSS데이터 셋에서 제공하는 스크립트 파일을 Notepad++의 매크로 기능으로 json파일 형식으로 수정하였다.

****

****

**그림 11** 스트립트 파일을 json형식으로 변환

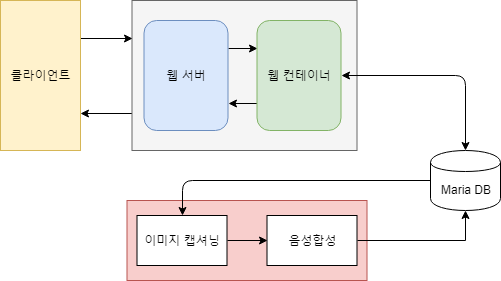
스크립트 파일을 제외한 나머지 전처리과정은 Tacotron2에서 제공하므로 그대로 이용하였다. 이를 이용하여 KSS 데이터의 전처리를 수행하였다. 'audio', 'mel', 'linear', 'time\_steps', 'mel\_frames', 'text', 'tokens', 'loss\_coeff'를 묶어 npz파일로 전처리한다. 생성된 npz파일 목록 중 일부는 그림 12에서 보여주고 있다.

****

**그림 12** KSS 데이터셋을 전처리한 파일 목록 중 일부

**3.3 전체 설계**

이미지를 음성으로 표현하는 프로그램을 만들기 위하여 이미지 캡셔닝 모듈과, 음성합성 모듈을 가지고 프로그램을 설계하였다. 그리고 웹서비스를 제공하기 위하여 웹서버, 웹컨테이너, DB가지고 설계하였다. 이미지 캡셔닝 모듈에서는 이미지를 입력받아 예측 문장으로 출력한다. 음성합성 모듈에서는 예측 문장을 받아 예측 음성으로 출력한다. 웹서비스는 클라이언트가 업로드 한 이미지를 DB에 업로드하고 이미지 캡셔닝 모듈과 음성합성 모듈을 거쳐 wav파일로 만든 뒤 DB에 저장한다. 이렇게 만든 wav파일은 웹 어플리케이션을 통하여 사용자에게 전달된다.

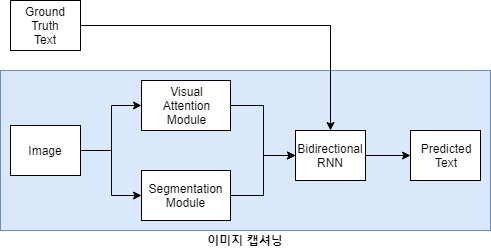
****

**그림 13** 전체 설계도

**3.4 세부 설계**

**3.4.1 이미지 캡셔닝**

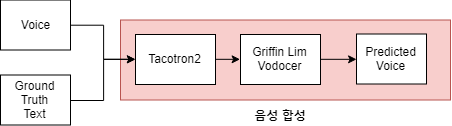
이미지 캡셔닝 모듈은 이미지를 설명하는 문장을 만들어내는 역할을 한다. 이미지 캡셔닝 모듈은 모델 학습과 문장 생성 두단계를 거칩니다. 첫번째로, 우선 이미지와 이미지 해설 문장을 받아서 모델 학습을 먼저 진행한다. 둘째로, 학습이 완료되고 서비스시에는 학습된 모델을 가지고 예측 문장을 생성한다. 학습 단계에서는 이미지와 문장은 세그멘테이션 모듈과 비쥬얼 어텐션 모듈, 그리고 양방향 RNN을 이용하여 학습을 진행한다. 문장 생성 단계에서는 이미지만을 가지고 예측 문장을 생성합니다. 이렇게 생성된 예측 문장은 음성합성 모델에게 전달된다.

****

**그림 14** 이미지 캡셔닝 모듈

**3.4.2 음성 합성**

음성 합성 모듈에서 문장을 wav파일로 합성하는 역할을 합니다. 음성 합성 모듈은 이미 다양한 구현체가 존재하기 때문에 Tacotron2를 사용하였다. 다만, Tacotron2의 보코더는 Wavenet이라는 보코더를 이용하는데 학습과 음성 생성이 오래 걸리기 때문에 시간상 Griffin Lim이라는 보코더를 이용하였다. 음성합성 모듈도 모델 학습과 음성 합성 두단계로 이루진다. 첫번째로, 우선 음성과 스크립트를 받아서 모델 학습을 먼저 진행한다. 둘째로, 학습이 완료되고 서비스시에는 학습된 모델을 가지고 wav파일을 생성한다. 이렇게 생성된 wav파일은 DB에 저장되며 웹 어플리케이션을 통해 사용자에게 전달된다**.**

****

**그림 15** 음성 합성 모듈

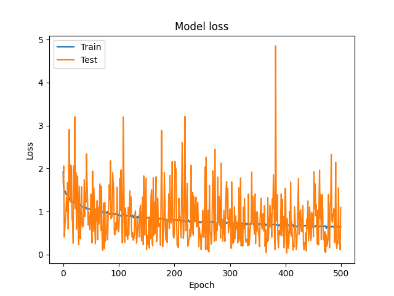
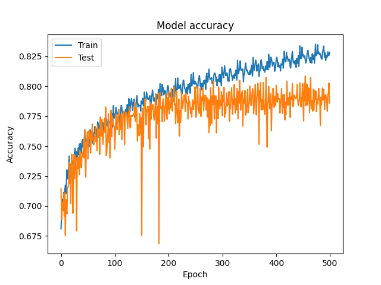
**3.4.3 웹 서비스 (추가 예정)**

**4. 구현 및 실험**

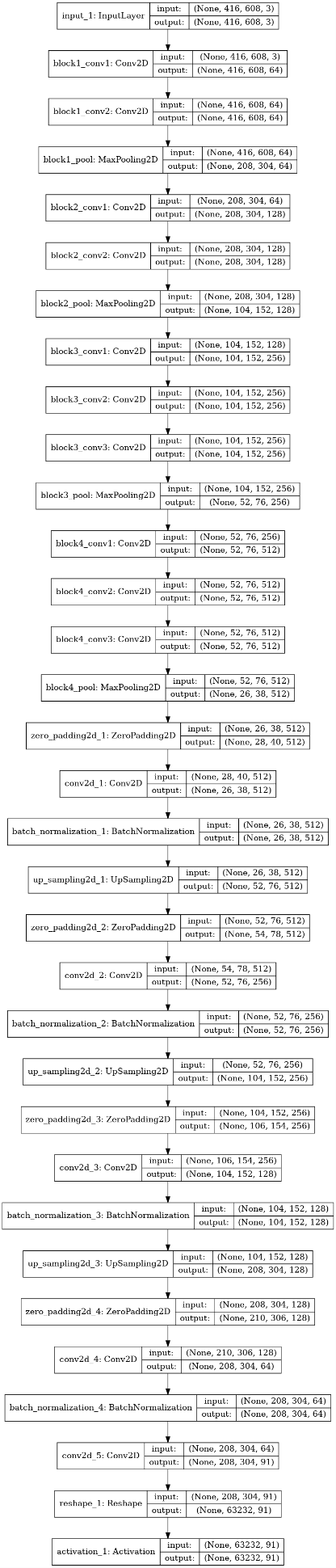
**4.1 이미지 캡션**

**4.1.1 영상 분할**

MS COCO 데이터셋을 이용하여 SegNet을 500epoch만큼 훈련시켰다. 훈련 중 모델의 정확도와 손실 그래프는 그림 16에서 보여주고 있으며, 모델의 구조는 그림 15에서 보여주고 있다. 500epoch 학습한 모델을 로드하여 그림 16의 왼쪽 이미지를 입력으로 모델의 예측 결과 오른쪽의 이미지를 획득하였다.

****

**그림 16** 영상 분할 모델의 정확도 및 손실 그래프

****

**그림 17** 영상 분할 모델 구조

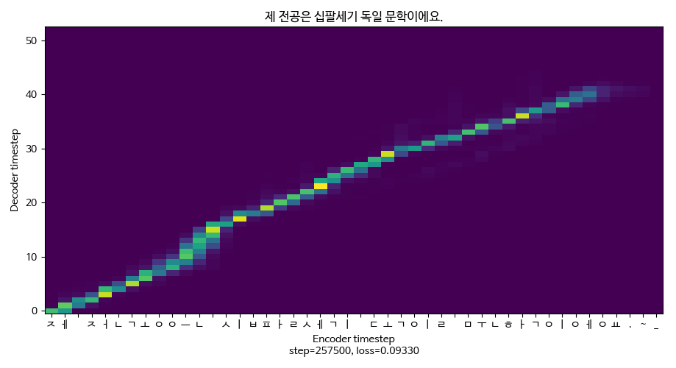
****

**그림 18** 모델의 예측

**4.1.2 이미지 캡션 (추가 예정)**

**4.2 음성 합성**

Tacotron2는 257,500epoch 만큼 학습하였으며, 테스트 문장에 대한 align 그래프는 그림 19과 같다.

****

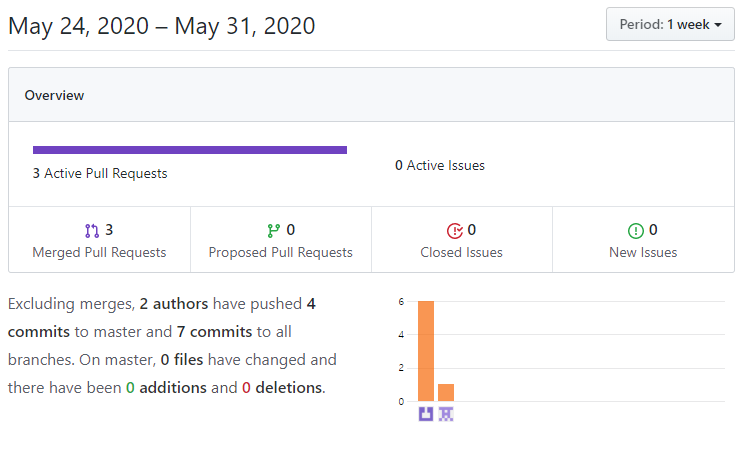
**그림 19** Tacotron2 음성 alignment 그래프

**4.3 웹 서비스 (추가 예정)**

**4.4 통합 테스트 (추가 예정)**

**5. 진행 상황**

**5.1 깃허브 로그화면**

****

**그림** 깃허브 로그

**6. 실험 및 결과**

**6.1 개발 환경**

실험은 Intel i9-9900K프로세서, GEFORCE RTX 2080 Ti 그래픽 프로세서, 64GB RAM, Ubuntu 18.04.2 LTS 컴퓨터를 이용하였다. 그래픽 프로세서를 이용한 GPGPU를 위하여 CUDA 10.2를 이용하였고, Docker Community Edition 19.03.8을 이용하여 Ubuntu 18.04.3 LTS 이미지로 가상환경을 설정한 후에 수행하였다. 가상환경에는 실험을 위해 python 3.6.9, tensorflow-gpu 1.15.2, keras 2.3.1를 이용하였다. 데이터 전처리를 위해 pandas 1.0.3, numpy 1.18.4, pycocotools 2.0.0를 이용하였고, 이미지 처리를 위하여 opencv-python 4.2.0.34, imageio 2.8.0를 이용하였다. 캡션 데이터 처리를 위하여 nltk 3.2.5 이용하였고, 영문 캡션을 한국어로 번역하기 위하여 pypapago 0.1.1.1를 이용하였다. 음성 합성을 위하여 ffmpeg 1.4, jamo 0.4.1를 이용하였다.

**6.2 고찰 및 토론 (추가 예정)**

**7. 참고 문헌**

[1] V. Badrinarayanan, A. Kendall and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 12, 2017, pp. 2481-2495.

[2] J. Shen et al., "Natural TTS Synthesis by Conditioning Wavenet on MEL Spectrogram Predictions," 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, 2018, pp. 4779-4783.

[3] N. Perraudin, P. Balazs and P. L. Søndergaard, "A fast Griffin-Lim algorithm," 2013 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, New Paltz, NY, 2013, pp. 1-4.

[4] Kelvin Xu, Jimmy Lei Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. Show, attend and tell: neural image caption generation with visual attention. In Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37 (ICML’15). 2015. pp. 2048–2057.