CNN과 환경 소리를 기반으로 한 장소 구분

Environment Sound Classification with CNN

요 약

ESC: Dataset for Environmental Sound Classification은 우리 주변에서 흔히 볼 수 있는 환경 소음을 이용하여서 이를 구분하는 연구를 한 논문이다. ESC-10, ESC-50기반의 데이터 셋으로 SVM, K-NN, Random Forest 기법을 이용하여서 이를 실험하였다. Performance Comparison of Acoustic Features for Sound Classification on Ubiquitous Environment 는 음성 Feature Mel-Spec, MFCC, STFT, Contrast, Chroma, Tonal Centroids와 CNN을 이용하여서 주변 환경 소리를 구분하는 논문이다. 이 프로젝트는 두 가지 논문을 참고하여서 특정 환경 소음이 아닌 여러 환경 소음이 섞여 있을 때 그 장소를 구분하는 방법을 CNN을 이용하여서 연구하는 것을 목표로 한다.

1. 서 론

최근에 CNN을 이용하여서 이미지뿐만 아니라 음성 쪽 연구도 활발하게 이루어지고 있다. 특히, Speech Recognition쪽은 활발하게 연구가 되어 Kaldi, Google API[3,[4],[5]등 많은 결과물이 나오고 있다. 이러한 음성 연구 중에서 환경 소음과 관련된 내용을 찾아 볼 수 있는데 연구들의 대부분은 특정한 환경 소음을 구분하거나 분류하는 연구들이었다. 즉, 우리 주변에서 실제로 볼 수 있는 복합적인 장소에 대한 분류에 대한 연구는 찾아보기 어려웠다. 만약 이러한 복합적인 장소를 구분 할 수 있는 모델을 만들 수 있다면 우리는 이를 Mobile Device, IoT Device와 접목 시켜서 편리한 Application을 만들 수 있을 것이라고 생각했다. 때문에 복합적인 소리를 이용하여서 장소를 구분하는 모델을 CNN을 기반으로 만들어 보는 것이 이번 프로젝트의 목표이다.

관련 연구로는 ESC: Dataset for Environmental Sound Classification[1]을 통해 특정 환경 소음에 대하여 ESC-10, 50이라는 데이터를 이용하여 구분하는 연구. 6가지 음성 Feature를 이용한 Performance Comparison of Acoustic Features for Sound Classification on Ubiquitous Environment[2] 등이 있다.

2. 기존 연구

2.1 ESC: Dataset for Environmental Sound Classification

이 논문은, 최근에 발전하고 있는 Machine Learning의 연구를 주변에서 흔히 볼 수 있는 환경 소음에 적용 하고자 하는 논문이다.

이 논문에는 환경 소음을 위한 연구를 위해서 ESC-10, ESC-50이라는 주변에서 흔히 볼 수 있는 환경 소음으로 구성된 데이터 셋을 이용한다. 두 데이터 셋의 차이는 분류하기 위한 class가 10과 50이라는 점이다. 연구를 위해서 사람이 직접 듣거나, K-NN, SVM, Radom Forest 기법을 통하여 분류를 얼마나 잘 할 수 있는지를 비교하였다. 이를 위해서 사용된 음성 Feature 추출 기법은 MFCC와 Zero-Crossing Rate를 이용하여 음성의 주파수 Feature를 이용하여서 연구를 진행하였다.

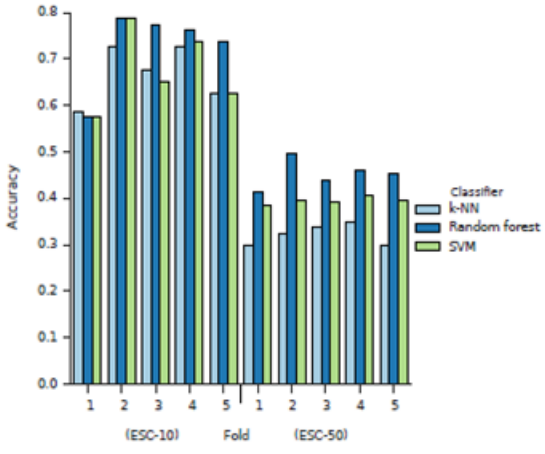

Figure 1 ESC with K-NN, Random Forest, SVM

ESC-10의 경우는 K-NN은 66.7%, Random Forest 72.7%, SVM은 67.5%의 Accuracy를 가졌다. 반면 class가 훨씬 늘어난 ESC-50의 경우는 K-NN은 32.2%, Random Forest는 44.3%, SVM은 39.6%를 기록함으로써 매우 낮은 정확도를 보여주었다. 이를 통해 구분하고자

하는 환경소음의 class의 양이 증가하면 증가할수록 더 구분을 못하는 것을 확인 할 수 있다. 또한, Category에 따라서도 정확도의 편차가 나타났다. 예를 들어서 사람의 소리, 동물 소리 같은 경우는 쉽게 구분을 할 수 있었다. 반면 기계의 소리, soundscapes/background 소리는 구분을 잘하지 못하는 현상을 보였다.

이 연구는 결과가 좋지 못하였고 한계가 존재한다. 하지만 환경 소음을 구별하기 위해서 Machine Learning을 사용한 의의가 있고 이를 위해서 ESC-10, ESC-50, ESC-US와 같이 환경 소음을 위한 데이터 셋을 공개적으로 공유한다. 때문에 이를 최근에 좋은 성능을 내고 있는 CNN이나 RNN을 적용하여 더 좋은 성능을 낼 수 있도록 해보는 것도 좋다고 생각한다.

2.2 Performance Comparison of Acoustic Features for Sound Classification on Ubiquitous Environment

이 논문에서는 CNN을 이용하여서 우리 주변에서 볼 수 있는 환경 소음을 구분하는 방법을 적용 하였다. 연구를 위해서 MFCC, STFT, Mel-Spectrum, Contrast, Chroma, Tonal Centroids 총 6가지의 음성 Feature를 사용하였다.

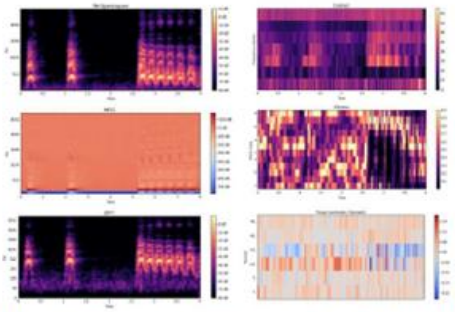

Fig. 1. Converted acoustic features (clockwise from right-top to left-top: Contrast, Chroma, Tonal Centroids, STFT, MFCC, Mel-spectrum)

Figure 2 Acoustic Features

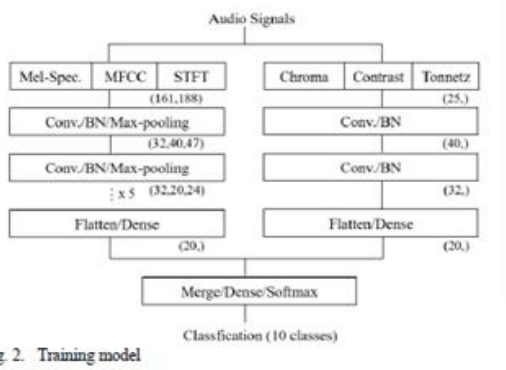

Fig. 2. Training model

Figure 3 Training Model

훈련을 위한 네트워크 모델은 Fig2와 같다. 왼쪽에서는 음원으로부터 Mel-Spectrum, MFCC, STFT를 이용하여서 총 161,188크기의 spectro-temporal acoustic features를 뽑아내어 5층의 Convolution, Max Pooling, Batch Normalization을 통하여 학습을 진행한다. 오른쪽에서는 Chroma, Contrast, Tonnetz를 이용하여 얻은 Feature로 2번의 Convolution과 Batch Normalization을 진행한다. 그 다음 양 네트워크를 Fully Connected를 이용하여 연결하고 마지막으로 2개의 네트워크에서 나온 결과 같을 Merge 하고 Fully Connected 한 다음 Softmax를 통해서 총 10개의 class를 분류 해낸다. 이를 통해서 Urban Sound 8K, Freefield1010 공개 데이터 셋을 이용하여서 훈련을 시켰다.

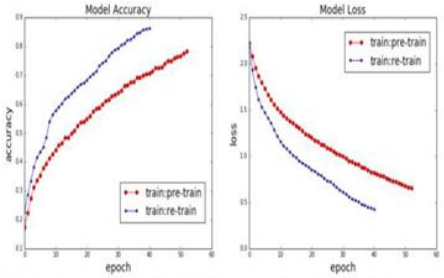

Fig. 4. Performance comparison between pre-train and re-train models

Figure 4 Performance comparison between pre-train and retrain models

결과는 Freefield1010를 이용하여 미리 훈련시킨 model의 경우는 86%, 그리고 재 훈련시킨 모델에 대해서는 78%의 결과를 내놓았다.

이 논문을 통해서 우리는 다양한 음성 Feature를 이용하여서 주변 환경 소음을 구분할 수 있다는 점을 우리의 연구에서 사용할 수 있을 것이다.

3. 시스템 모델

3.1 기존 연구와 차이점 및 해결방안

위에서 언급 된 2개의 논문들의 목적은 특정 환경 소음을 구분하는 것이다. 때문에 우리 주변 환경에서 흔하게 볼 수 있는 실제 여러 환경 소음이 복합적으로 섞여 있는 소음을 구분하는 연구는 아니다. 따라서 우리는 주변환경에서 들을 수 있는 복합적인 환경 소음이 섞인 소리를 이용하여서 장소를 구분 할 수 있는 Application을 만드는 것이 기존의 연구와 다른 점이다.

기존의 ESC-10과, 50을 간단한 CNN 모델로 만들어서 학습하고 Accuracy를 측정한 코드는 존재하였지만 훈련에 필요한 복합적인 환경

소리에 대한 데이터 셋은 없었기 때문에 이를 직접 수집한 다음 CNN에 훈련하기 알맞은 형태로 가공하는 작업을 해야 한다. 또한 데이터 셋의 변화 또는 모델의 형태에도 변화를 주어서 정확도를 계속해서 관찰하고 증가시키는 것을 목적으로 연구를 진행한다.

마지막으로 실제로 훈련 데이터, 검증 데이터, 테스트 데이터 외의 소리 파일을 이용하여서 모델의 객관성을 확인한다.

3.2 프로젝트 내용

이 프로젝트는 기존의 연구와는 다르게 복합적인 환경 소음에 대하여 장소를 판별하는 시스템이다. 목표는 정확도 80% 이상으로 잡으며 5개의 Category 영역과 15개의 세부 class를 구분하는 것을 목적으로 삼는다. 사용한 음성 Feature는 Mel-Spectrogram이기 때문에 모델 학습을 위해서 Python의 librosa library를 이용하여서 데이터 셋을 직접 만들었다.

Figure 5 Manufacture Data

Category & Class

Nature	Loud	Music	Quiet	ETC
Mountain	Factory	Concert	Home	Road
Sea	Airport	Classic	Office	Lecture
Cave	Harbor	Club	Library	Café

프로젝트를 진행하기 위한 최종 결과 학습 모델은 다음과 같다.

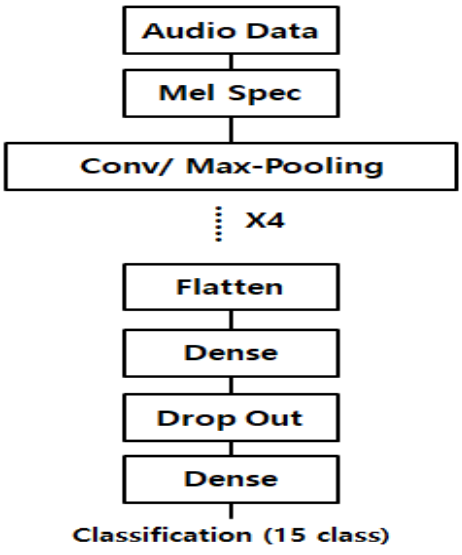

Figure 6 Train Model

먼저 오디오 데이터를 Figure5처럼 훈련을 위한 데이터 셋으로 변환을 한 다음 4개의 Conv/Max-Pooling Layer를 이용하여서 학습을 시킨다. 그 다음 Flatten 하나로 합친 다음 Drop Out의 수치를 50%로 두어 Over fitting을 방지할 수 있도록 한다. Class분류를 위해서 Optimizer는 Adam을 사용하였고, Loss Function은 Sparse Categorical Cross Entropy를 이용하여 One-Hot Code 방식이 아닌 class 번호에 맞는 형식으로 데이터를 분류 할 수 있도록 하였다. 전체 데이터의 90%를 Train Data, 10%를 Test Data로 삼았으며, Train Data에서 다시 9:1의 비율로 Train과 Validation으로 나누어서 프로젝트를 진행 하였다.

4. 연구 결과 및 분석

다음은 이 프로젝트를 어떻게 진행을 하였으며 그 결과물과 그에 대한 분석이다.

4.1 Result 1 (Categories 3, class 9, sample 960)

처음 훈련의 경우는 15개의 class가 아닌 Nature, Loud, Music 3개의 Category들과 하위 9개의 class를 통해 만들어진 960개의 데이터로 실험을 해보았다.

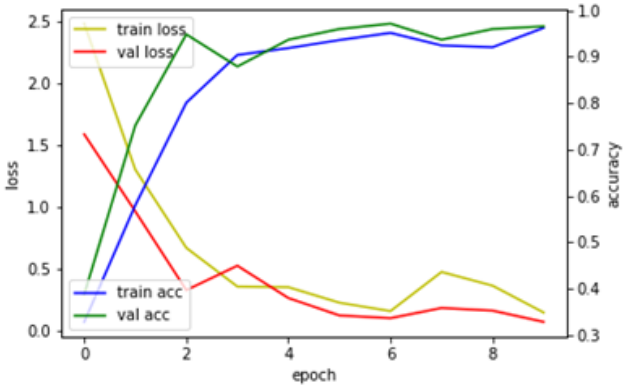

Figure 7 Result 1

파란색이 훈련의 정확도 이며, 초록색이 검증의 정확도, 노란색이 훈련의 loss 값이며, 붉은색이 검증의 loss 값이다. Over fitting 이 일어나거나 loss 값이 지속적으로 줄어드는 모습을 보이고 있다. 하지만 처음 나온 결과물이었기 때문에 데이터 셋을 좀더 확장 시키고 Train Model을 변화를 준다면 더 좋은 결과를 얻을 수 있다고 생각했다. 때문에 먼저 960개로는 데이터 셋의 객관성이 부족하다고 생각이 들었기 때문에 데이터 셋의 질과 양을 늘리는 쪽으로 프로젝트를 진행하였다.

4.2 Result 2 (Categories 5, class 15, sample 2974)

데이터를 더 늘려서 5 Category의 15개 class로 파생된 2974개의 데이터 셋으로 실험을 다시 하였다. 이 때 데이터 셋이 늘어 났기 때문에 훈련 epoch를 30까지 늘려서 좀 더 모델이 학습을 더 할 수 있도록 만들었다.

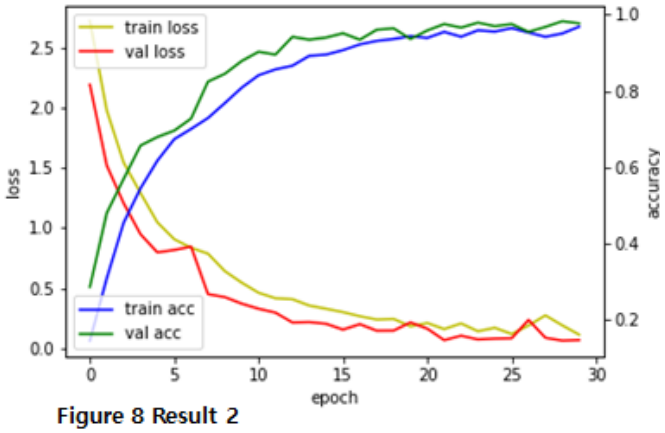

Figure 8 Result 2

Result1에 비해서 좀 더 안정적인 학습을 하고 있는 모습을 보여줬으며 정확도나 Loss가 더 줄어드는 모습을 보여주었다. 하지만 3000개라고 할지라도 각 class에 대한 audio sample을 200개 가량 밖에 되지 않았기 때문에 여전히 객관성이 부족하다고 판단하였고 마지막으로 데이터 셋을 6000개로 늘려서 훈련을 시켜보았다.

4.3 Result 3(Categories 5, class 15, sample 6000)

데이터 셋을 6000개까지 늘린 이후 데이터 셋에 대한 변화는 그만 두고 CNN 모델에 대한 변화를 주면서 훈련을 시켜보았다. Drop Out layer, Dense Layer쪽에 주로 변화를 주어서 실험을 해보았고 3가지 case 중 가장 좋은 정확도를 나타낸 것이 Figure6에서 나타난 모델이었다.

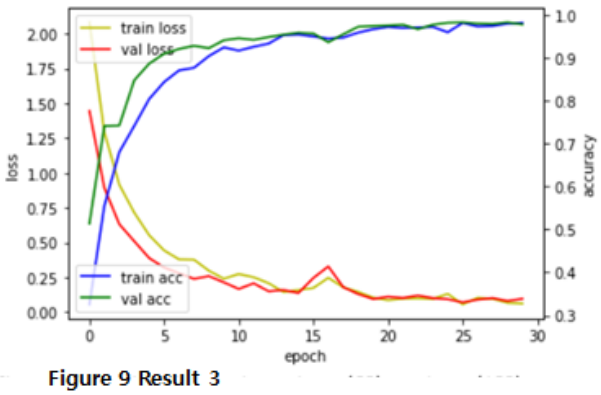

Figure 9 Result 3

가장 좋은 정확도 값과 loss 값을 가진 결과이다. 총 109개의 Sound Data로부터 6000개의 Train Sample을 만들었고 이를 바탕으로 네트워크에 변화를 주면서 나온 결과 물이다. Train Evaluate는 99.2%가 나왔고 Test Evaluate의 경우는 98.5%라는 높은 결과가 나왔다.

4.4 프로젝트 검증(일반화 확인)

위 실험에서 나온 3가지의 결과물은 정해진 훈련 데이터 셋 안에서 진행 된 내용이었다. 그렇다면 우리가 만든 모델이 실제로 잘 작동하는지를 위해서 훈련에 사용되지 않은 전혀 새로운 데이터를 통해서 일반화가 잘 이루어졌는지를 검증해보았다.

Model Validation Accuracy

Valid Count	Category	Class
33	72%	55%
45	71%	53%
100	65%	52%
150	66%	49%

위 표를 본다면 결과물은 목표치인 80%도 달성하지 못하였으며 Category는 60%후반, Class는 50%라는 낮은 정확도를 보였다. 그런데 이를 Category 별로 다시 정확도를 분석하였을 때 흥미로운 결과물이 나왔다.

Category Accuracy

Category	Category ACC	Class ACC
Nature	73%	66%
Loud	46%	40%
Music	93%	76%
Quiet	63%	20%
ETC	56%	43%

먼저 Category별, Class별 정확도의 편차가 매우 심하다. Music Category의 경우는 93%까지 정확도를 보였지만 Loud의 경우는 46%라는 낮은 점을 보였다. 또한, Category의 영역의 정확도와 class별의 정확도의 차이가 상당히 심한 것도 알 수 있다. Category는 아무래도 공통점이 존재하는 영역이다 보니 그 정확도가 Class에 비해서 더 높게 나오는 것으로 분석이 된다. 또한 정확도가 높은 Music Category는 음악이라는 명확한 음성 Feature가 존재하고 Nature의 경우는 인공적이거나 인위적인 소리가 적기 때문에 Feature를 잡기 쉬워서 정확도가 높게 나온 것으로 분석된다. 만약 Category 영역을 더 명확하게 하고 더 많은 데이터를 수집 할 수 있다면 특정영역에서만큼은 단순한 CNN모델로도 충분히 분류가 가능하다는 점을 알 수 있었다.

5. 결론 및 향후 연구

먼저 결과를 본다면 목표한 정확도는 달성하지 못하였다. 모델의 일반화가 실패 한 것은 크게

2가지 요인으로 보고 있다.

첫 번째 하나의 장소에 너무 많은 음성 Feature가 존재하는 것이다. 사람의 음성, 기계, 자연소리 등 수많은 Feature가 단 하나의 장소에 몰려 있었고 이를 CNN을 통해서 학습을 시켰기 때문에 사소한 음성 Feature조차 학습을 하여서 Over fitting이 일어났다고 생각이 든다. 때문에 일반화가 제대로 이루어지지 못하였다.

두 번째 학습에 사용된 데이터가 완벽하게 장소를 객관화 할 수 있는 데이터라고 보기 힘들다. 첫 번째 이유에서도 설명을 하였지만 하나의 장소에는 여러 복합적인 환경 소음이 존재한다. 그렇기 때문에 다양한 케이스가 발생 할 수 밖에 없고 그에 따라서 중심 Feature를 찾는 것도 매우 어렵다. 때문에 우리가 수집한 데이터가 그 장소를 대표하는 완벽한 객관적 데이터라고 보기 어렵다. 이를 해결하기 위해서는 6천개가 아닌 최소 10만개 단위의 데이터 샘플이 있어야 기본적인 객관화가 이루어 질 것이라고 생각한다.

목표한 정확도는 달성하지 못하였지만 검증 결과를 보았을 때 환경소리를 구분 할 수 있는 가능성 자체는 발견 할 수 있었다. Category를 보다 명확하게 하게 공통적인 Feature를 특정하여서 이를 바탕으로 훈련을 시킨다면 Environmental Sound Classification이 아주 불가능하지는 않다는 점을 알 수 있다. 이를 위해서는 먼저 데이터 셋의 절대적 크기를 늘려야 한다고 생각하고 Mel-Spectrogram 뿐만 아니라 다른 복합적인 음성 Feature로 이용을 해야 한다고 여겨 진다. 만약 이 연구가 실용화가 될 수 있다면 모바일 영역, IoT 영역과 연계를 하여서 주변 환경에 맞추어 Device의 상태를 자동으로 변경시킬 수 있는 Application을 만들 수 있을 것이다. 때문에 앞으로의 연구는 정확도를 올리는 방향으로 진행을 하면서 동시에 이를 활용한 Application의 응용 방법도 생각을 해볼 것이다.

참 고 문 헌

[1] Karol J. Pcizak, Institute of Electronic Systems Warsaw University of Technology Warsaw, Poland
ESC: Dataset for Environmental Sound Classification, 2015, Harvard Dataverse, V2

[2] Nac-Woo Kime, Jun-Gi Lee, Hyun-Yong Lee, Byung-Tak Lee, Su-Kyung Kang, Myung-Hye Park, Electronics and Telecommunications Research Institute, KEPCO Research Institute
Performance Comparison of Acoustic Features for Sound Classification on Ubiquitous Environment
Proceedings of Symposium of the Korean Institute of communications and Information Sciences, 2019.1, 1298-1299(2pages)

[3] Quan Wang, Hannah Muckenhirn, Kevin Wilson, Prashant Sridhar, Zelin Wu, John Hershey, RifA. Saurous, Ron J. Weiss, Ye Jia, Ignacio Lopez Moreno, Google Inc., USA, Idiap Research Institute, Switzerland EPFL, Switzerland.
VoiceFilter: Targeted Voice Separation by Speaker Conditioned Spectrogram Masking ((Submitted on 11 Oct 2018 (v1), last revised 19 Jun 2019 (this version, v6))

[4] ARIEL EPHRAT, INBA MOSSERI, ORAN LANG, TALI DEKEL, KEVIN WILSON, AVINATAN HASSIDIM, VILLIAM T.FREEMAN, MICHAEL, RUBINSTEIN, Google Research, The Hebrew University of Jerusalem, Israel.
Looking to Listen at the Cocktail Party: A Speaker-Independent Audio-Visual Model for Speech Separation

[5] Fully Supervised Speaker Diarization - Aonan Zhang, Quan Wang, Zhenyao Zhu, John Paisley, Chong Wang (Submitted on 10 Oct 2018 (v1), last revised 19 Feb 2019 (this version, v7))

[6] Tensorflow - Speech Commands Code
https://github.com/tensorflow/tensorflow/tree/master/tensorflow/examples/speech_commands

[7] Environmental-Sound-Classification
https://github.com/bheemnitd/Environmental-Sound-Classification