AI를 활용한 음색 전이 프로그램

Sound Style Transfer Program based on AI

이홍석(Hongseok Lee), 박민재(Minjae Park)

요약

논문 내용 요약

이미지 처리 분야에서 사진을 그림처럼 바꾸거나, 만화 캐릭터처럼 얼굴을 변환하는 기술은 널리 사용되고 있다. 이처럼 이미지에서의 Style Transfer 기술이 발전하면서, 이를 유사한 다른 데이터에도 적용할 방법을 고민하였고, 데이터 생성에 효과적인 모델들을 사용하여 음성 Style Transfer에서 높은 성능을 내는 모델을 선정하고, 실제 음성 데이터를 입력하여 결과값을 확인하였다. 그 결과, CycleGAN이 정성적으로 가장 좋은 산출물을 내었으며, 추후 trian data와 현재 Style Transfer에서 SOTA 모델들을 추가로 시도할 예정이다.

키워드: AI, Transfer Program, CycleGAN, 음성 변환, 데이터 학습, 데이터 분할, 학습 데이터 수집

Abstract

abstract of paper

In the field of image processing, technologies that change photographs into pictures or transform faces like cartoon characters are widely used. With the development of Style Transfer technology in images, we considered how to apply it to other similar data, selected high-performance models in voice Style Transfer using models effective for data generation, and checked the results by entering actual voice data. As a result, CycleGAN produced the best qualitatively, and SOTA models will be additionally attempted in trian data and current Style Transfer in the future.

Keyword : AI, Transfer Program, CycleGAN, sound conversion, Data Learning, Data Division, Collect Training Data

1. 서론

이미지 처리 분야에서의 Style Transfer기술은 아주 사실적으로 변하여, 실제 사람과 AI가 생성한 사람의 사진을 구분하기 어려울 정도까지 발전하였고, 이는 이미지 처리 분야에서 생성 모델의 성능이 개선되었음을 방증한다. 반면, 음성 처리 영역에서는 상대적으로 사용 가능한 정도의 완성도를 제공하는 서비스의 양이 적다. 일반적인 음성 스타일 전이 모델은 인물의 목소리 데이터를 tts와 한 쌍으로 학습시킨 후, 텍스트 데이터를

input으로 받아 음성을 생성하는 모델인데, 이는 별도의 텍스트 데이터를 제공할 수 없는 음원 데이터에서 적용하기 어렵다는 한계가 있다. 이에, 우리는 다양한 생성 모델을 시도한 끝에, unpaired learning 방법을 사용할 수 있고, 음성 전이에서 높은 성능을 보이는 CycleGAN을 사용하기로 결정하고, hyperparameter를 조절해 가며, 음원 데이터의 실제 변환값을 관측하였다.

2. 관련 연구

2.1. image data에 대한 CycleGAN

CycleGAN은 기존의 pix2pix를 통해 미완성 스케치를 완성된 형태로 바꾸는 연구에서, paired image를 학습 데이터로 사용해야 된다는 단점을 해결하기 위해, unpaired dataset을 사용하면서도 Style Transfer에서 좋은 성능을 보일 수 있도록 개선된 모델이다.

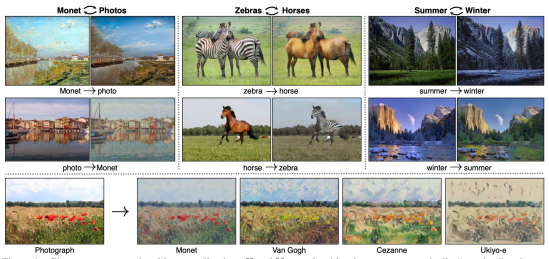

Figure 1: Given any two unordered image collections X and Y, our algorithm learns to automatically "translate" an image from one into the other and vice versa: *(left)* Monet paintings and landscape photos from Flickr; *(center)* zebras and horses from ImageNet; *(right)* summer and winter Yosemite photos from Flickr. Example application *(bottom)*: using a collection of paintings of famous artists, our method learns to render natural photographs into the respective styles.

그림 1 변환파일 추출

이것을 통해, unpaired dataset으로부터 학습된 모델을 가지고, 얼룩말을 일반 말로 변환하거나, 사진을 그림풍으로 변환하는 등의 연구에서 일부 유의미한 성과를 보였다.

3. 사용 방법

우선 학습시킬 데이터를 불러오는 과정을 거친다. 아래 사진과 같이 Google 드라이브에 저장해 놓은 데이터들을 불러 올 수 있도록 Path를 설정해 놓았다.

```
path = glob('/content/drive/MyDrive/data/datasets/Custom_data/%s/%s/*' % (self.dataset_name, data_type))
```

그림 2 학습 데이터 불러오기

다음으로, 데이터를 학습 시키고 저장해놓은 체크포인트를 불러온다. 아래 사진과 같이 학습을 하고 드라이브에 저장해 놓았던 Checkpoint를 불러오는 과정을 거친다.

```
#체크포인트 로드
checkpoint_path = '/content/drive/MyDrive/data/datasets/cyclegan_checkpoint_0601.h5'
checkpoint = ModelCheckpoint(checkpoint_path, monitor='val_loss', verbose=1, save_best_only=False, mode='min')

if os.path.exists(checkpoint_path):
    self.combined.load_weights(checkpoint_path)
    print("Done")
```

그림 3 체크포인트 불러오기

그 다음, 변환하고자 하는 음성파일을 Colab환경에서 추가한다. 아래 사진을 보면 좋은 밤 좋은 꿈.mp3라는 음성파일을 추가한 다음 코드에서 해당 파일을 불러오는 작업을 거친다.

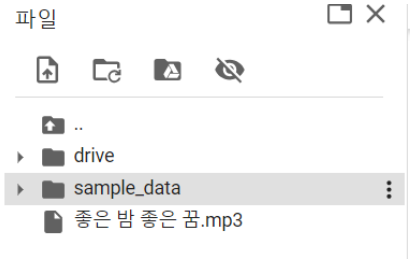

그림 4 음성파일 추가

```
y, sr = librosa.load("/content/좋은 밤 좋은 꿈.mp3")

min_len = 262000
y = y[:min_len]

n_fft = 2048
n_hop = 512
n_mels = 512

# Mel spectrogram
melspec = librosa.feature.melspectrogram(y=y, sr=sr, n_fft=n_fft, hop_length=n_hop, n_mels=n_mels)
```

그림 5 음성파일 불러오기

Piano에서 Guitar로 변환하고자 할 때는 model.g_AB.predict(input)를 사용하고,

```
out = model.g_AB.predict(input)
```

그림 6 음성변환_Piano to Guitar

Guitar에서 Piano로 변환하고자 할 때는 model.g_BA.predict(input)를 사용하여 변환한다.

```
out = model.g_BA.predict(input)
```

그림 7 음성변환_Guitar to Piano

마지막으로, 변환한 음성파일을 wav형태로 추출한다.

```
import soundfile as sf

# 생성한 오디오 파일(audio)을 WAV 파일로 저장
sf.write('/content/output.wav', audio, sr)
```

그림 8 변환파일 추출

4. 설계

여기서는 전체적인 설계 흐름과 구조, 학습 데이터를 설명한다.

4.1 순서도 (흐름도)

프로젝에 사용된 모델은 generator, discriminator로 구성된다.

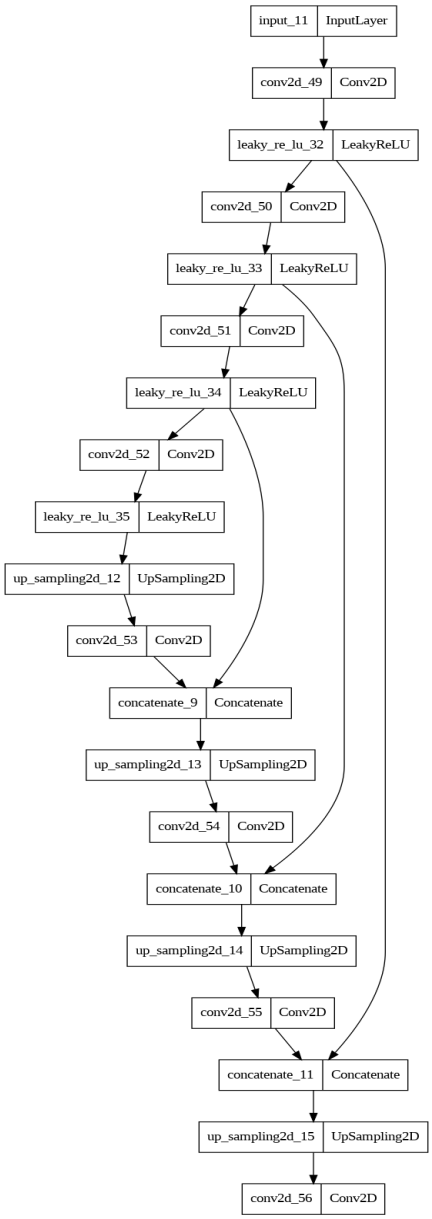

그림 9 generator

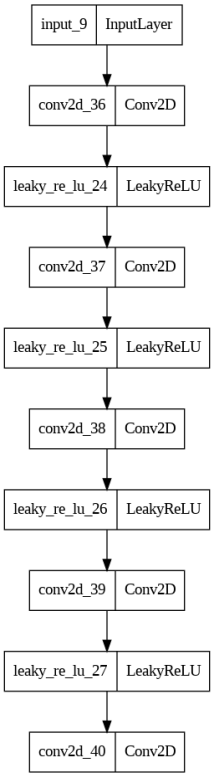

그림 10 discriminator

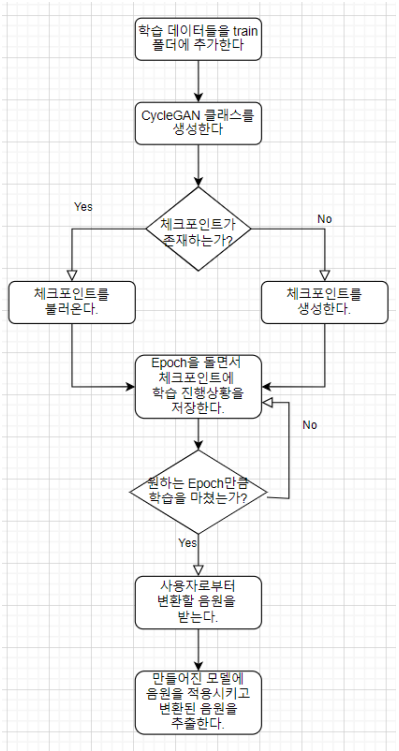

그림 11 음성변환 train 과정

4.2 학습 데이터

Google Colab을 사용하여 미리 준비해둔 데이터들을 학습시키기 위해서 우선 학습 데이터들의 전처리과정을 할 필요가 있었다.

피아노 음성파일들과 기타 음성파일들을 사전에 모아두고, CycleGAN 모델을 바탕으로 만든 모델 규격에 맞도록 음성파일들의 전처리과정을 거친다.

```
mel_A = librosa.feature.melspectrogram(y=y_A_ex, sr=sr_A, n_fft=2048, hop_length=512, n_mels=512)
mel_B = librosa.feature.melspectrogram(y=y_B_ex, sr=sr_B, n_fft=2048, hop_length=512, n_mels=512)

img_A = np.zeros((mel_A.shape[0], mel_A.shape[1], 1), dtype=np.float32)

img_A[..., 0] = mel_A
'''
img_A[..., 1] = 0
img_A[..., 2] = 0
'''
img_B = np.zeros((mel_B.shape[0], mel_B.shape[1], 1), dtype=np.float32)
img_B[..., 0] = mel_B
'''
img_B[..., 1] = 0
img_B[..., 2] = 0
'''

imgs_A.append(img_A)
imgs_B.append(img_B)

if(len(imgs_A)==batch_size):
  imgs_A = np.array(imgs_A)
  imgs_B = np.array(imgs_B)
```

그림 12 학습데이터 전처리과정

해당 전처리과정을 거치게 되면 음성파일들은 (4, 512, 512, 1)의 모양을 갖추게 된다(여기서 4는 batch_size 이다). 그 다음에 이렇게 만든 데이터들을 학습시키게 된다.

5. 결론 및 향후 연구

기타 특유의 쨍쨍거리는 소리가 담겨서 변환되는 결과를 얻었다. 이를 통해서 피아노에서 기타로의 변환은 생각한 것 이상의 유사도를 볼 수 있었다. 반면 기타에서 피아노로의 변환은 예상했던 것보다 부진한 결과를 얻었다. 이는 학습데이터 중 기타 음원들에서 일렉기타, 클래식 기타, 통기타 등

음색이 통일되지 않고 다양하게 존재했다는 점이 문제가 될 수 있을 것이라 생각하였다.

따라서 향후 연구에 있어서 해당 문제점을 극복하고자 다양한 방법을 생각해보았다. 그 중 하나는 동일한 음색을 가진 음원을 가져오기 위해서 midi파일로 모든 음원들을 변환시켜 학습 데이터군으로 사용하는 것을 생각해보았다. 또한, CycleGAN으로 생성한 음성의 특성 상 노이즈가 발생하는데, 노이즈 제거 AI를 후처리 모델로 사용하여 노이즈를 줄여 더욱 깔끔한 음성을 얻는 것이 가능할 것이다.

참고 문헌

1. Audio Style Transfer (2018.04.18) (https://gauthamsanthosh.medium.com/audio-style-transfer-d339abb430a3), Gautham Santhosh

2. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks (2017. 03. 30) (https://arxiv.org/abs/1703.10593) , Jun-Yan Zhu, Taesung Park, Phillip Isola, Alexei A. Efros