VOICE CONVERSION
Hung-yi Lee
李宏毅
What is Voice Conversion (VC)?

What is preserved?  Content

What is changed?  Many different aspects ...
Speaker

- The same sentence said by different people has different effect.
- Deep Fake: Fool humans / speaker verification system
- One simple way to achieve personalized TTS
- Singing

[Nachmani, et al., INTERSPEECH’19]
https://enk100.github.io/Unsupervised_Singing_Voice_Conversion/

[Deng, et al., ICASSP’20]
https://tencent-ailab.github.io/pitch-net/
Speaker

• Privacy Preserving
  [Srivastava, et al., arXiv’19]

(詳見獵人第八卷)
Speaking Style

• Emotion
  [Gao, et al., INTERSPEECH’19]

• Normal-to-Lombard
  [Seshadri, et al., ICASSP’19]

• Whisper-to-Normal
  [Patel, et al., SSW’19]

• Singers vocal technique conversion
  [Luo, et al., ICASSP’20]

‘lip thrill’ (彈唇) or ‘vibrato’ (顫音)

Source of audio:
https://shreyas253.github.io/SpStyl
eConv_CycleGAN/
Improving Intelligibility

• Improving the speech intelligibility
  • surgical patients who have had parts of their articulators removed
    [Biadsy, et al., INTERSPEECH’19][Chen et al., INTERSPEECH’19]

• Accent conversion
  • voice quality of a non-native speaker and the pronunciation patterns of a native speaker
  • Can be used in language learning
    [Zhao, et al., INTERSPEECH’19]
Data Augmentation

VC

Training Data x 2

[Keskin, et al., ICML workshop’19]

VC

VC

VC

Clean Speech

Noisy Speech

[Mimura, et al., ASRU 2017]
In real implementation ...

- **Voice Conversion**: Usually $T' = T$
  - Seq2seq is not needed

- **Vocoder**
  - Rule-based: Griffin-Lim algorithm
  - Deep Learning: WaveNet

Used in VC, TTS, Speech Separation, etc. (not today)
Categories

Parallel Data

Lack of training data:
• Model Pre-training [Huang, et al., arXiv’19]
• Synthesized data! [Biadsy, et al., INTERSPEECH’19]

Unparallel Data

How are you?

天気真好

How are you?

This is “audio style transfer”
• Borrowing techniques from image style transfer
Categories

Parallel Data

Unparallel Data

Direct Transformation

Feature Disentangle

Content Encoder

Speaker Encoder

phonetic information

speaker information
Feature Disentangle

Do you want to study PhD?

Content Encoder

Do you want to study PhD?

Speaker Encoder

Do you want to study PhD?

Decoder
Feature Disentangle

Do you want to study PhD?

Good bye

Do you want to study PhD?
Feature Disentangle

as close as possible (L1 or L2 distance)

input audio

Content Encoder

Speaker Encoder

Decoder

reconstructed

How can you make one encoder for content and one for speaker?
Using Speaker Information

Assume we know the speakers of training utterances

- One-hot vector for each speaker

[Hsu, et al., APSIPA’16]
Using Speaker Information

Assume we know the speakers of training utterances

- One-hot vector for each speaker
Pre-training Encoders

- Speech recognition

• One-hot vector for each speaker

Issue: difficult to consider new speakers

• Speaker embedding (i-vector, d-vector, x-vector ... )

[Sun, et al., ICME’16][Liu, et al., INTERSPEECH’18]

[Qian, et al., ICML’19][Liu, et al., INTERSPEECH’18]
Adversarial Training

[Chou, et al., INTERSPEECH’18]

Learn to fool the speaker classifier

Speaker classifier and encoder are learned iteratively
Designing network architecture

Content Encoder = instance normalization (remove speaker information)
Designing network architecture

\[ \text{IN} \quad = \text{instance normalization (remove speaker information)} \]
Designing network architecture

Each channel has zero mean and unit variance

Normalize for each channel

Content Encoder
Designing network architecture

Content Encoder

Speaker Encoder

Decoder

How are you?

How are you?

IN = instance normalization (remove speaker information)
Designing network architecture

How are you?

Encoder

Speaker Encoder

Content Encoder

IN

Decoder

AdaIN

IN

AdaIN

= instance normalization  (remove speaker information)

= adaptive instance normalization

(only influence speaker information)
$z'_i = \gamma \odot z_i + \beta$

Output of Speaker Encoder

Decoder

AdaIN = adaptive instance normalization

(only influence speaker information)
Designing network architecture

- Content Encoder
- Speaker Encoder

How are you?

Which speaker?

<table>
<thead>
<tr>
<th></th>
<th>With IN</th>
<th>Without IN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acc.</td>
<td>0.375</td>
<td>0.658</td>
</tr>
</tbody>
</table>

Training from VCTK
Designing network architecture

How are you?

Content Encoder

Speaker Encoder

Training from VCTK

For more results
[Chou, et al., INTERSPEECH 2019]

Unseen Speaker Utterances

male
female
**Issues**

**Training**

A is reading the sentence of B

The Same Speakers

Content Encoder

Decoder

Discriminator

Which speakers?

reconstructed

**Testing**

A is reading the sentence of B

Different Speakers

Content Encoder

Decoder

Low Quality

Hello

A

B

Hello
2nd Stage Training

Cheat discriminator
Help speaker classifier

Extra Criterion for Training

real or generated?
which speaker?

Different Speakers

[Chou, et al., INTERSPEECH’18]
[Liu, et al., INTERSPEECH’19]
2nd Stage Training

Only learn the patcher in the 2^{nd} stage

Different Speakers

Extra Criterion for Training

real or generated?

which speaker?

Discriminator

Speaker Classifier

content encoder

decoder

patcher

Hello

A

B
Categories

- Parallel Data
  - Training without parallel data
  - Using CycleGAN

- Unparallel Data

- Feature Disentangle

- Direct Transformation
Cycle GAN

Speaker X

$G_{X \rightarrow Y}$

Speaker Y

Become similar to speaker Y

$D_Y$

Input audio belongs to speaker Y?

Speaker Y

Speaker X

[Kaneko, et al., ICASSP'19]
Cycle GAN

Speaker X

ignore input

\(G_{X \rightarrow Y}\)

Become similar to speaker Y

Not what we want!

\(D_Y\)

scalar

Input audio belongs to speaker Y?

Speaker Y
Cycle GAN

as close as possible (L1 or L2 distance)

Cycle consistency

Input audio belongs to speaker Y?
Cycle GAN

\[ G_{X \rightarrow Y} \rightarrow D_X \rightarrow G_{Y \rightarrow X} \]

\[ D_Y \rightarrow G_Y \rightarrow X \rightarrow G_X \rightarrow Y \]

as close as possible

scalar: belongs to speaker X or not

scalar: belongs to speaker Y or not
StarGAN

[Kaneko, et al., INTERSPEECH’19]

For CycleGAN:
If there are N speakers, you need N x (N-1) generators.
StarGAN

Each speaker is represented as a vector.

Audio of speaker \( s_i \)

Speaker \( s_j \)

Audio of speaker \( s_j \)

Speaker \( s_i \)

Scalar: belongs to input speaker or not
The classifier is ignored here.
Blow

Flow-based model for VC

Figure 1: Blow schema featuring its block structure (left), steps of flow (center), and coupling network with hyperconvolution module (right).

Ref for flow-based model: https://youtu.be/uXY18nzdSsM
Concluding Remarks

- Parallel Data
  - Unparallel Data
  - Direct Transformation
  - Feature Disentangle
Reference


• [Seshadri, et al., ICASSP’19] Shreyas Seshadri, Lauri Juvela, Junichi Yamagishi, Okko Räsänen, Paavo Alku, Cycle-consistent Adversarial Networks for Non-parallel Vocal Effort Based Speaking Style Conversion, ICASSP, 2019
Reference

Reference


Reference

- [Chen et al., INTERSPEECH’19] Li-Wei Chen, Hung-Yi Lee, Yu Tsao, Generative adversarial networks for unpaired voice transformation on impaired speech, INTERSPEECH, 2019
- [Zhao, et al., INTERSPEECH’19] Guanlong Zhao, Shaojin Ding, Ricardo Gutierrez-Osuna, Foreign Accent Conversion by Synthesizing Speech from Phonetic Posteriorgrams, INTERSPEECH, 2019
- [Qian, et al., ICML’19] Kaizhi Qian, Yang Zhang, Shiyu Chang, Xuesong Yang, Mark Hasegawa-Johnson, AUTOVC: Zero-Shot Voice Style Transfer with Only Autoencoder Loss, ICML, 2019
Reference

- [Sun, et al., ICME’16] Lifa Sun, Kun Li, Hao Wang, Shiyin Kang, Helen Meng, Phonetic posteriorgrams for many-to-one voice conversion without parallel data training, ICME, 2016

- [Liu, et al., INTERSPEECH’18] Songxiang Liu, Jinghua Zhong, Lifa Sun, Xixin Wu, Xunying Liu, Helen Meng, Voice Conversion Across Arbitrary Speakers Based on a Single Target-Speaker Utterance, INTERSPEECH, 2018
