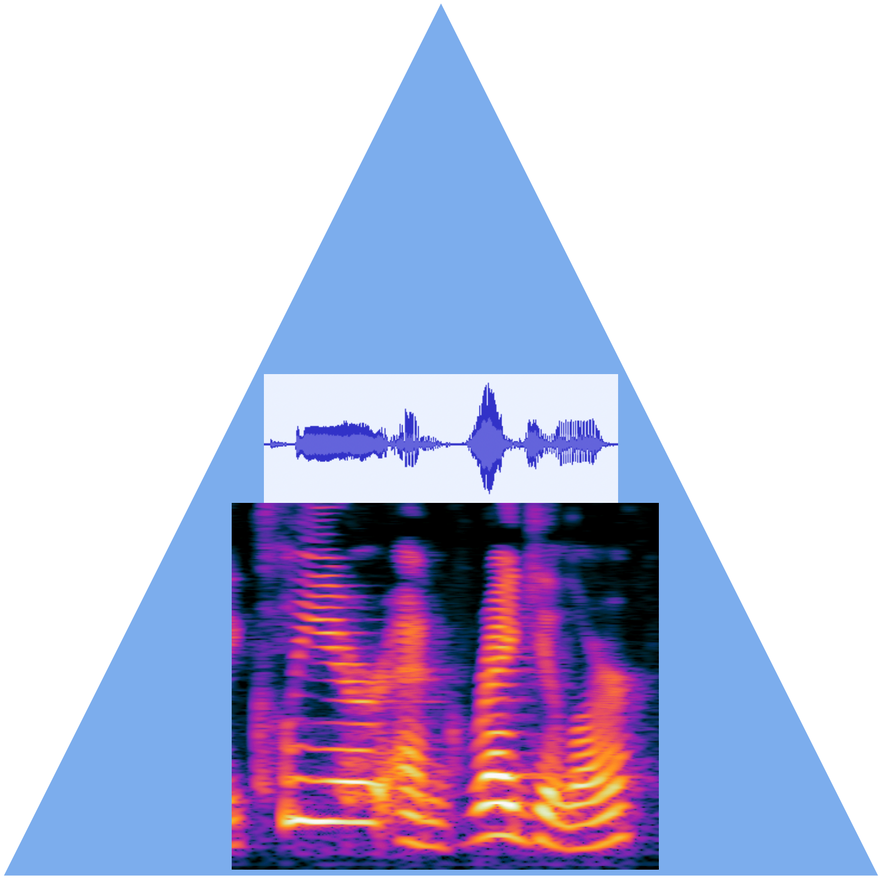
Lecture 15: Voice Conversion

Zhizheng Wu

Agenda

- Recap
- Voice conversion
- Cross-lingual voice conversion
- Singing voice conversion

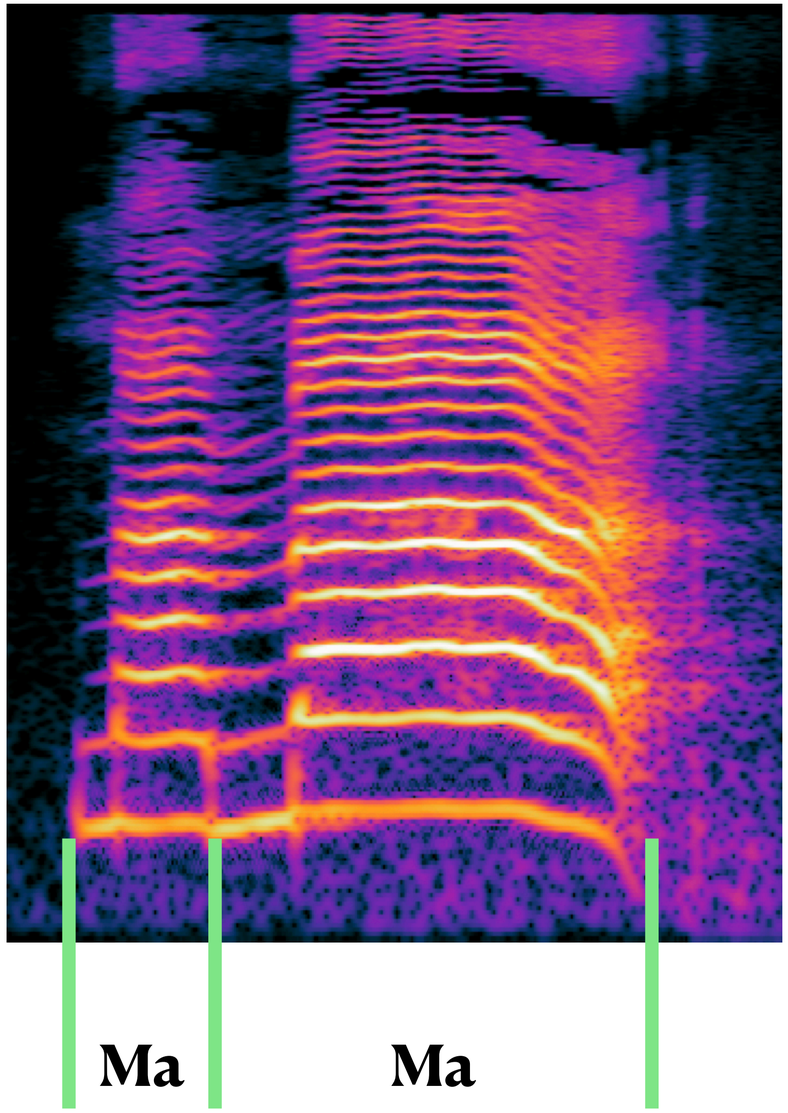
Content

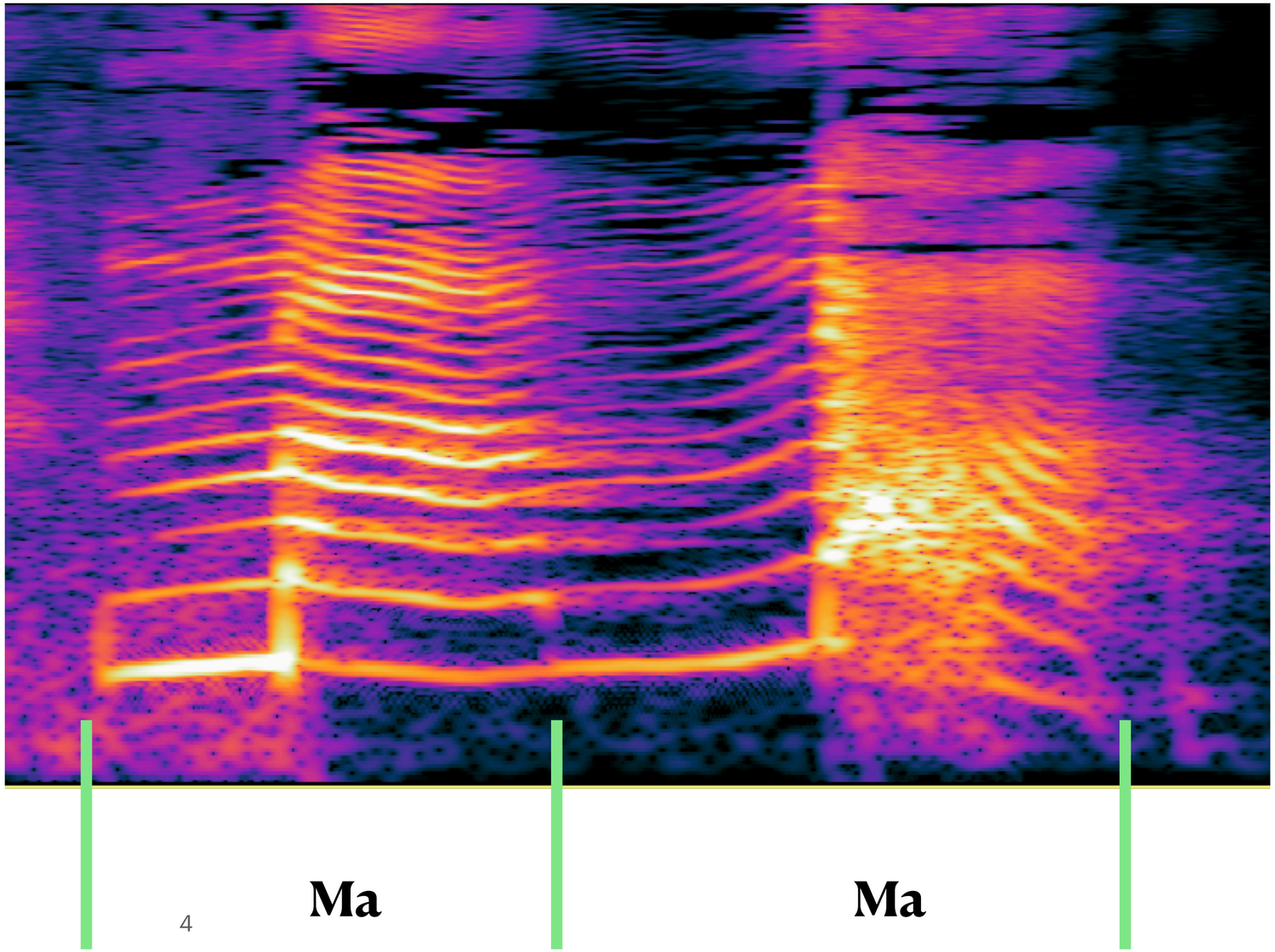


Timbre

Prosody

Speech representation





Timbre difference

- Each speaker has its unique speaker identity

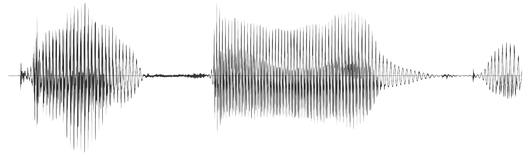
Text to speech

- Generate an audible audio given a sequence of text

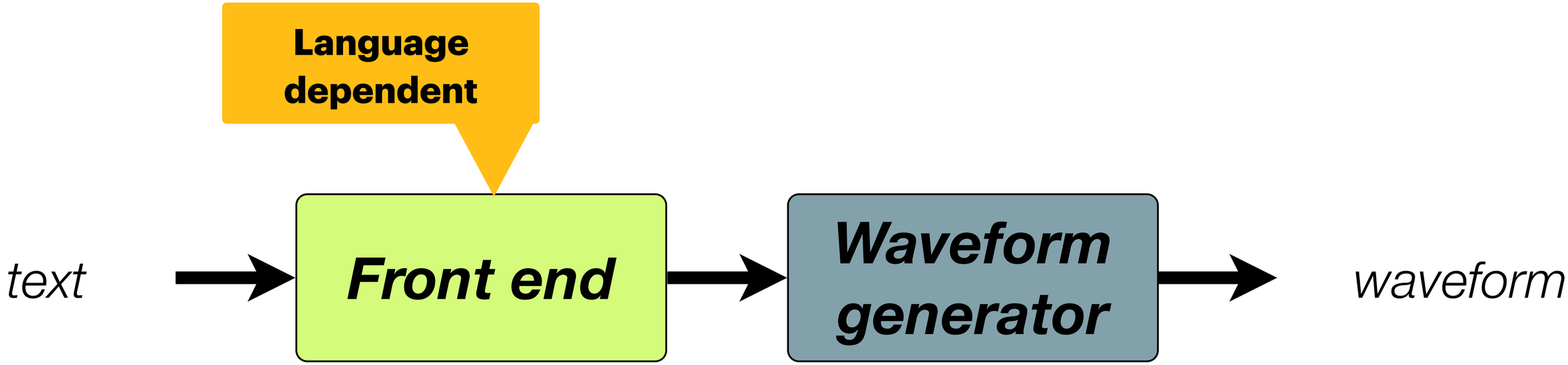
The end-to-end problem we want to solve

text → ***Text-to-Speech*** → *waveform*

`Author of the…`

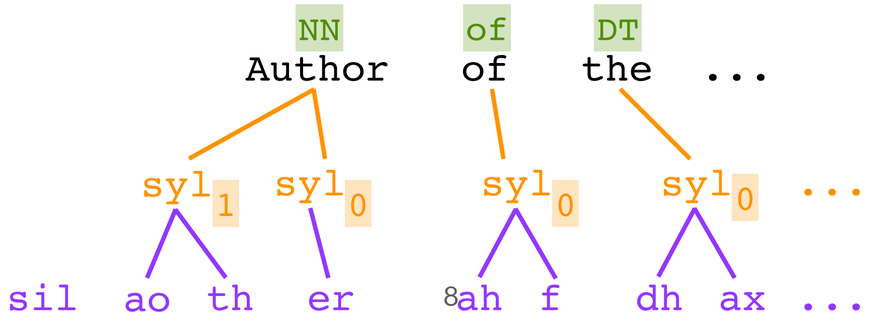


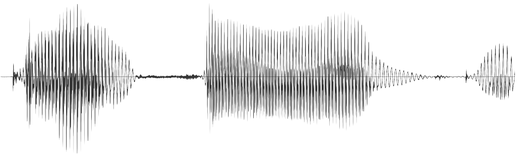
The two-stage pipeline



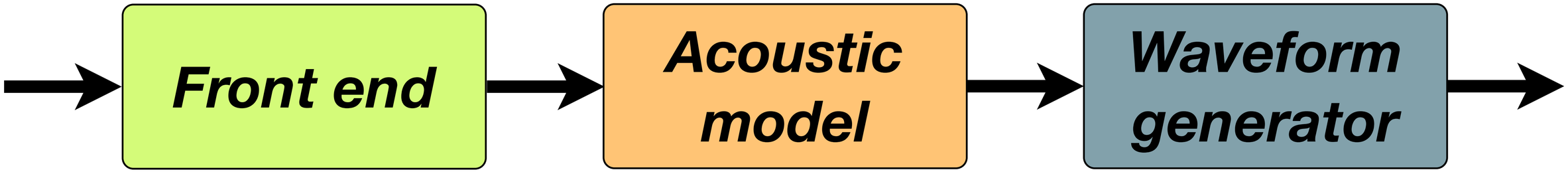
Author of the…

linguistic specification





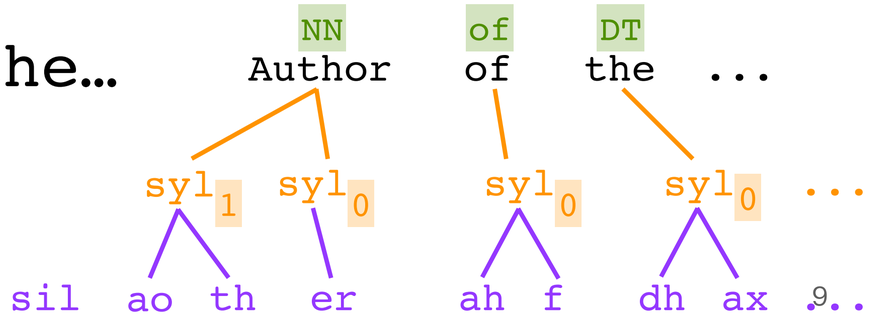
The three-stage pipeline



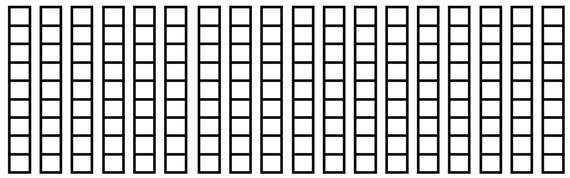
text

Author of the…

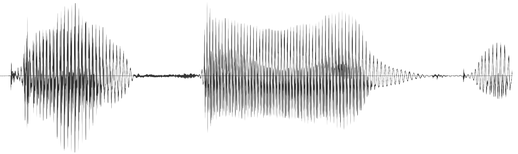
linguistic specification



acoustic features



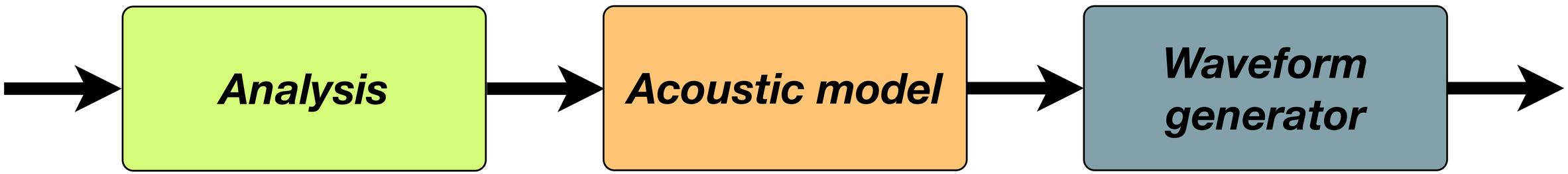
waveform



Voice conversion

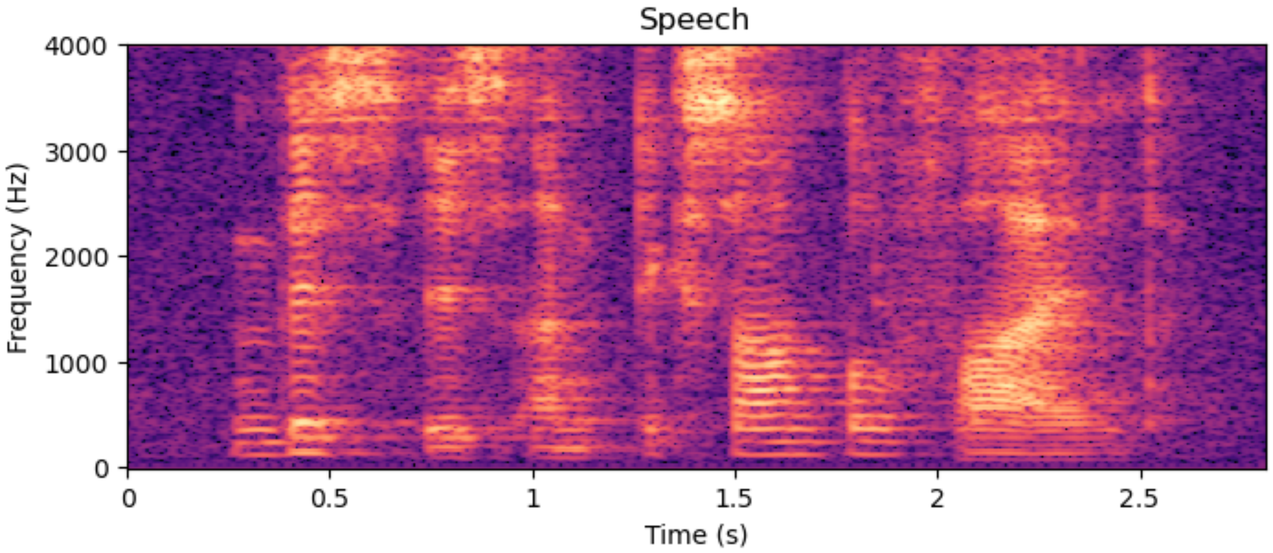
- Converting one speaker’s voice to sound like another speaker without changing language content

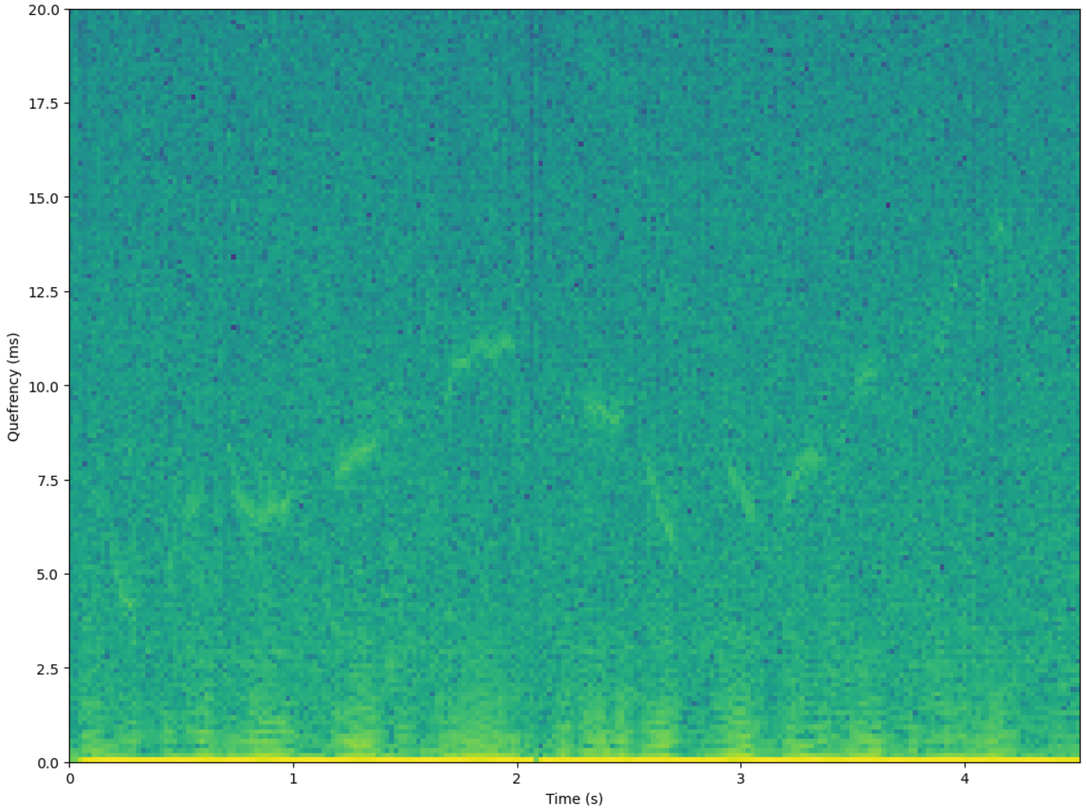
Voice conversion: three stages



Voice conversion: Analysis

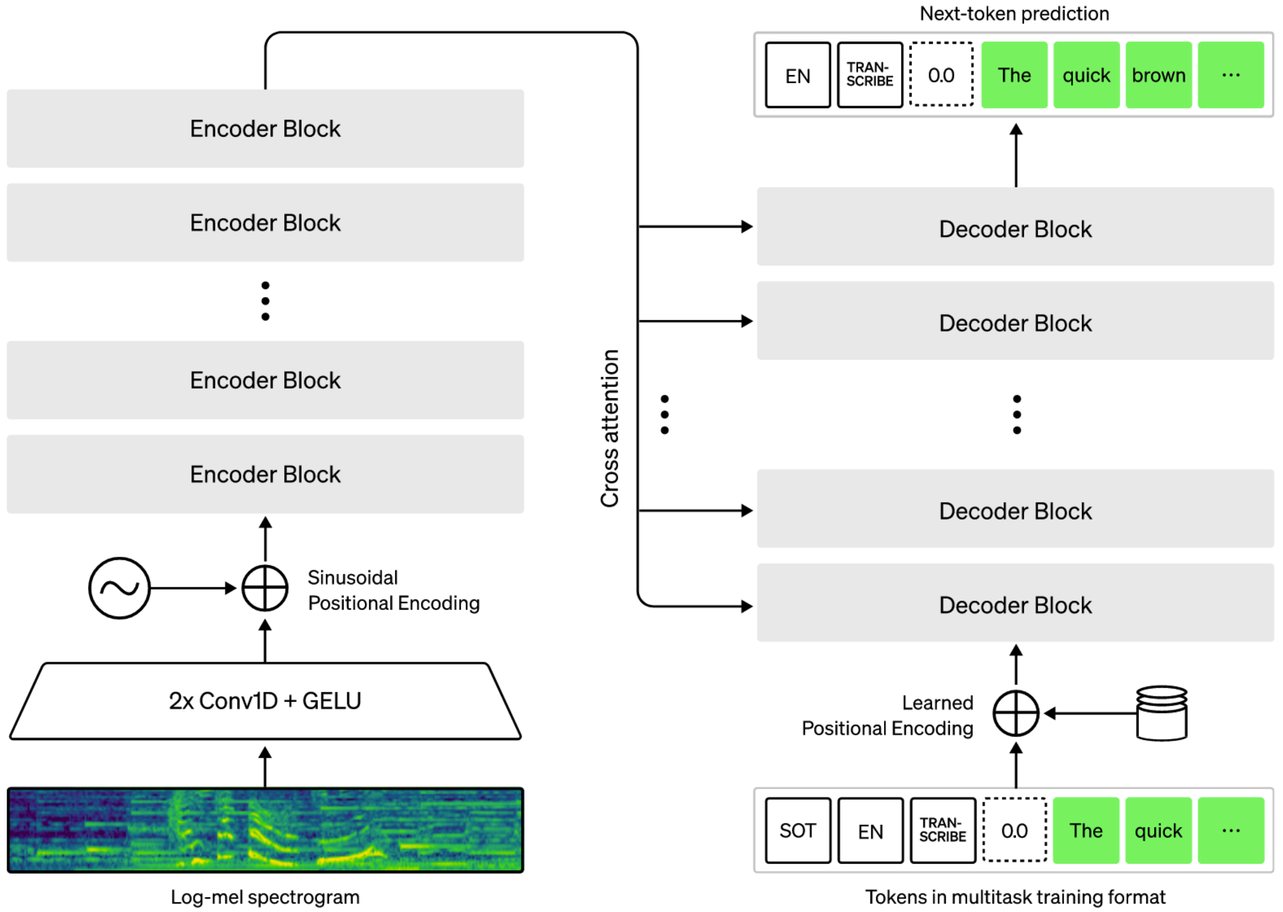
- Hand-crafted features





Voice conversion: Analysis

- Using pretrained model



Voice conversion: Acoustic model

Voice conversion: Acoustic model

Voice conversion: Acoustic model

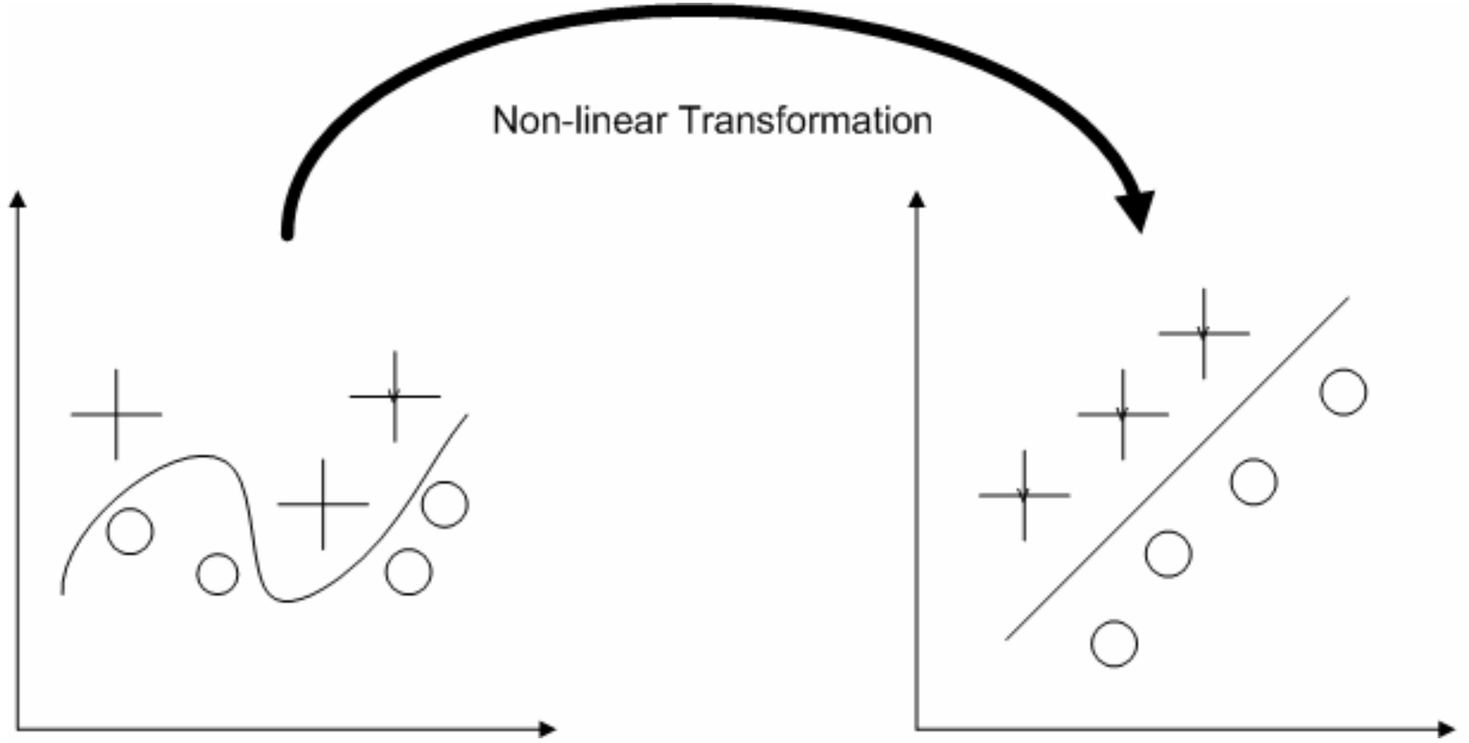
- Weighted linear transformation

$$X' = WX$$

$$where\ W_{n\times n} = \begin{bmatrix} w_0^0 & w_0^1 & \cdots & w_0^n \\ w_1^0 & w_1^1 & \cdots & w_1^n \\ \vdots & \vdots & \ddots & \vdots \\ w_n^0 & w_n^1 & \cdots & w_n^n \end{bmatrix},\ X_{n\times 1} = \begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}$$

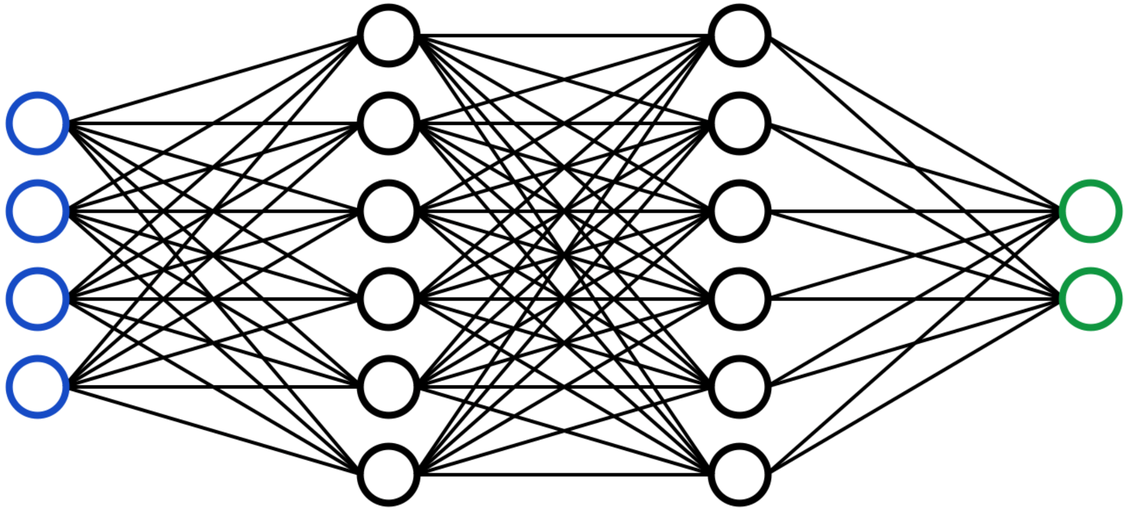
Voice conversion: Acoustic model

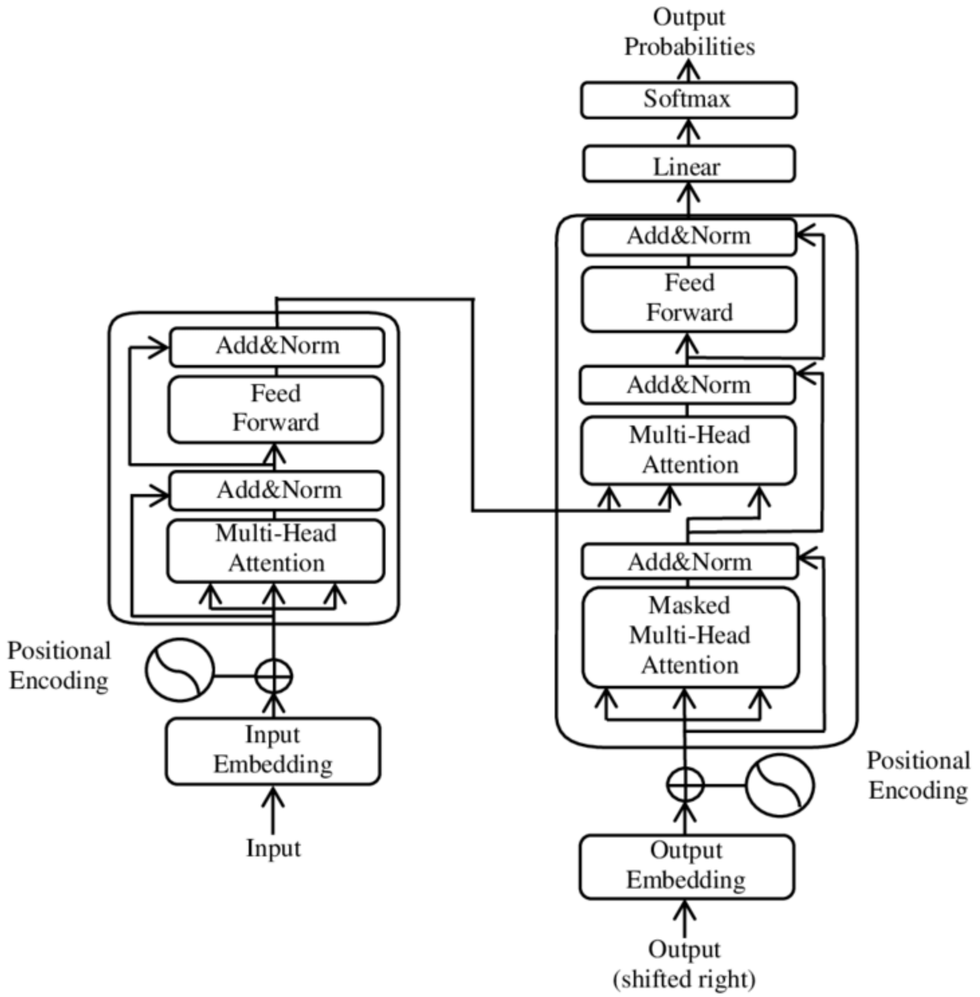
- Nonlinear transformation



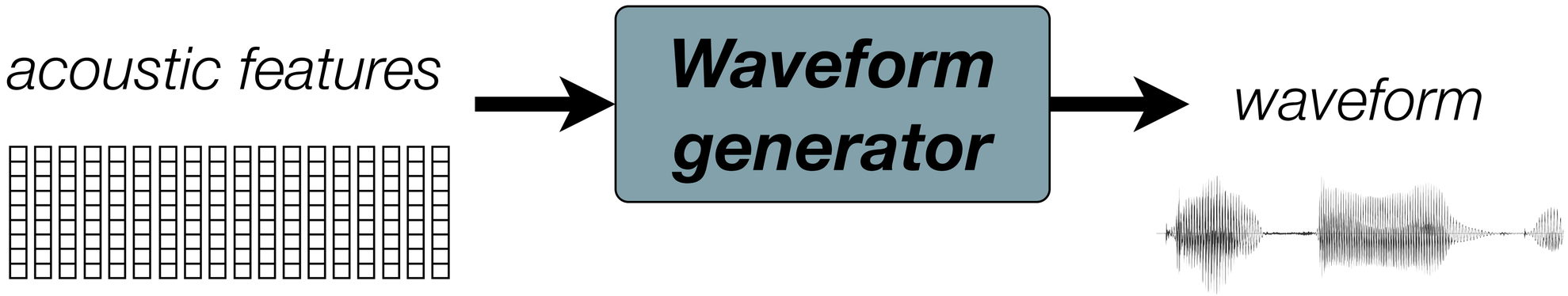
Voice conversion: Acoustic model

- Nonlinear transformation





Voice conversion: Waveform generator



Artifacts of voice conversion

- Short-time Fourier transform (e.g. constant frequency/time resolutions)
- Inaccuracy in pitch estimation

- Smoothing effect due to statistical averaging
- Inaccuracy in statistical modeling

- Distortion introduced by vocoder/neural vocoder
- Upsampling
- Phase: 1) phase discontinuity 2) minimum phase vocoding

Cross-lingual voice conversion: Example

Voice dubbing in a different language

- The original movie actor may not speak different languages
- A native voice actor is needed
- However the voice timber between the native voice actor and the original movie actor is different

What is Singing Voice Conversion (SVC)?

Professional Singer1

Professional Singer2

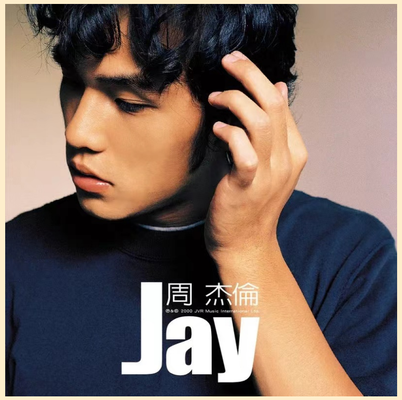
Inter-singer Conversion

Speaker

Singer

Cross-domain Conversion

Amateur Singer

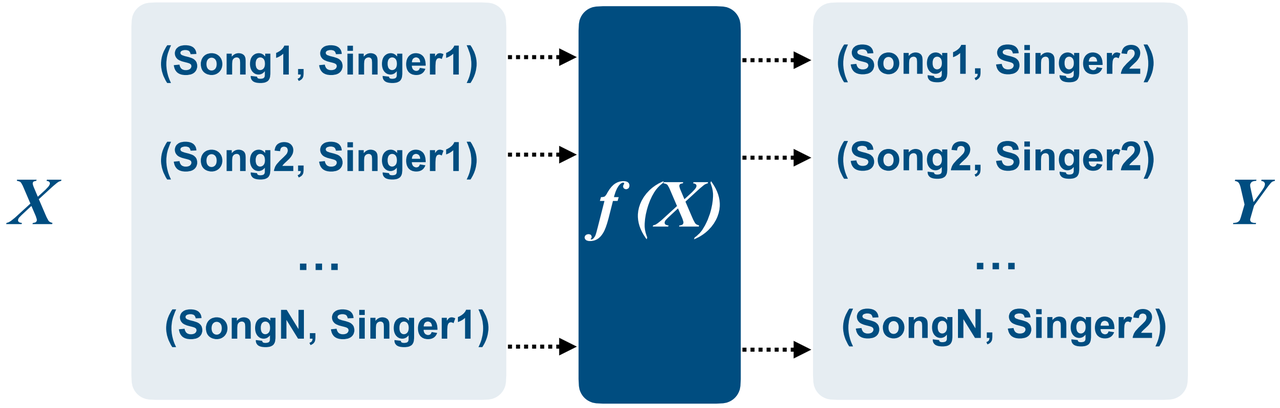


Professional Singer

Intra-singer Conversion

Parallel Singing Voice Conversion

Professional Singer1 → **Professional Singer2**



Parallel corpus is hard to collect!

Non-Parallel Singing Voice Conversion

Professional Singer1

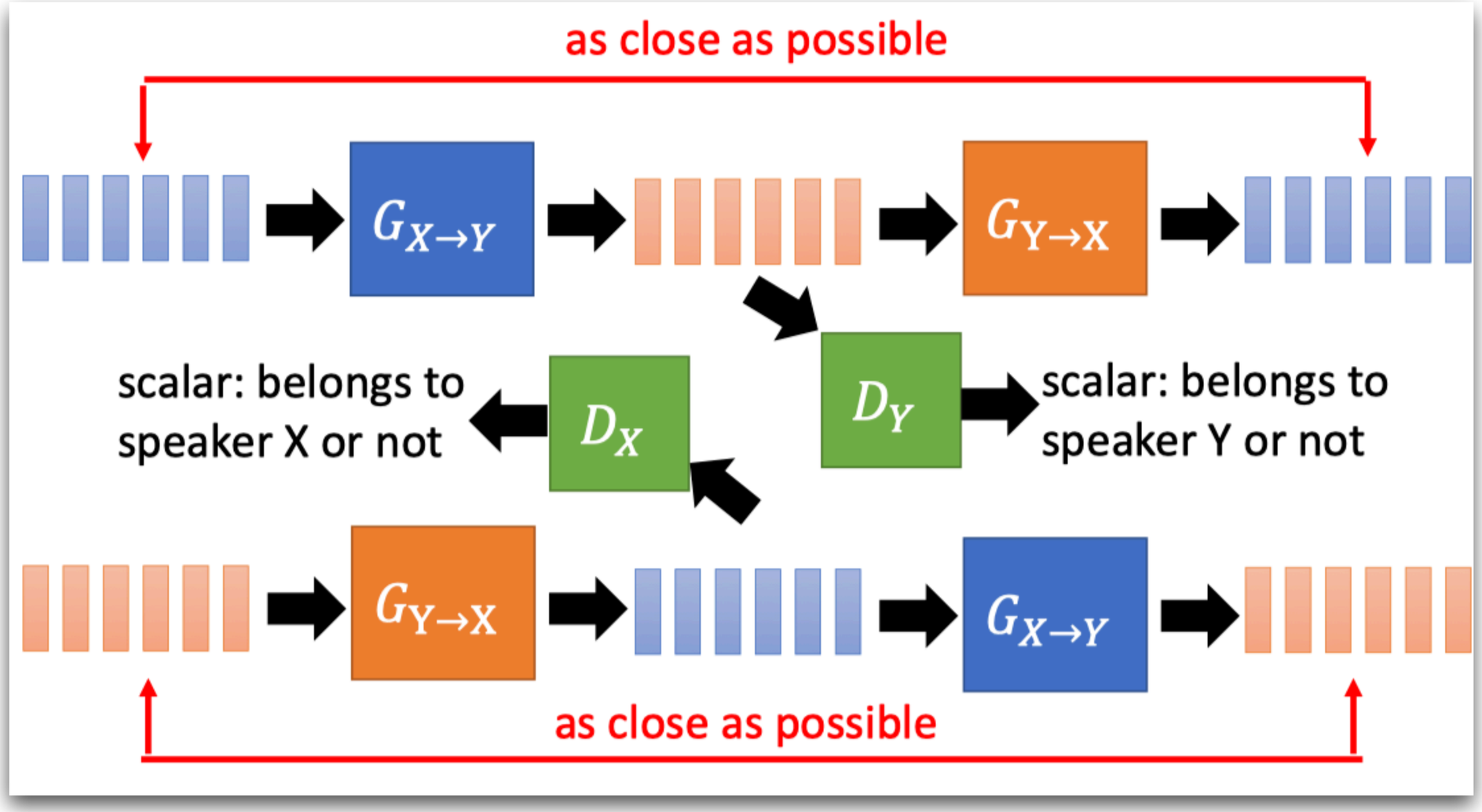
Professional Singer2

X

Singer1's Songs Singer2's Songs

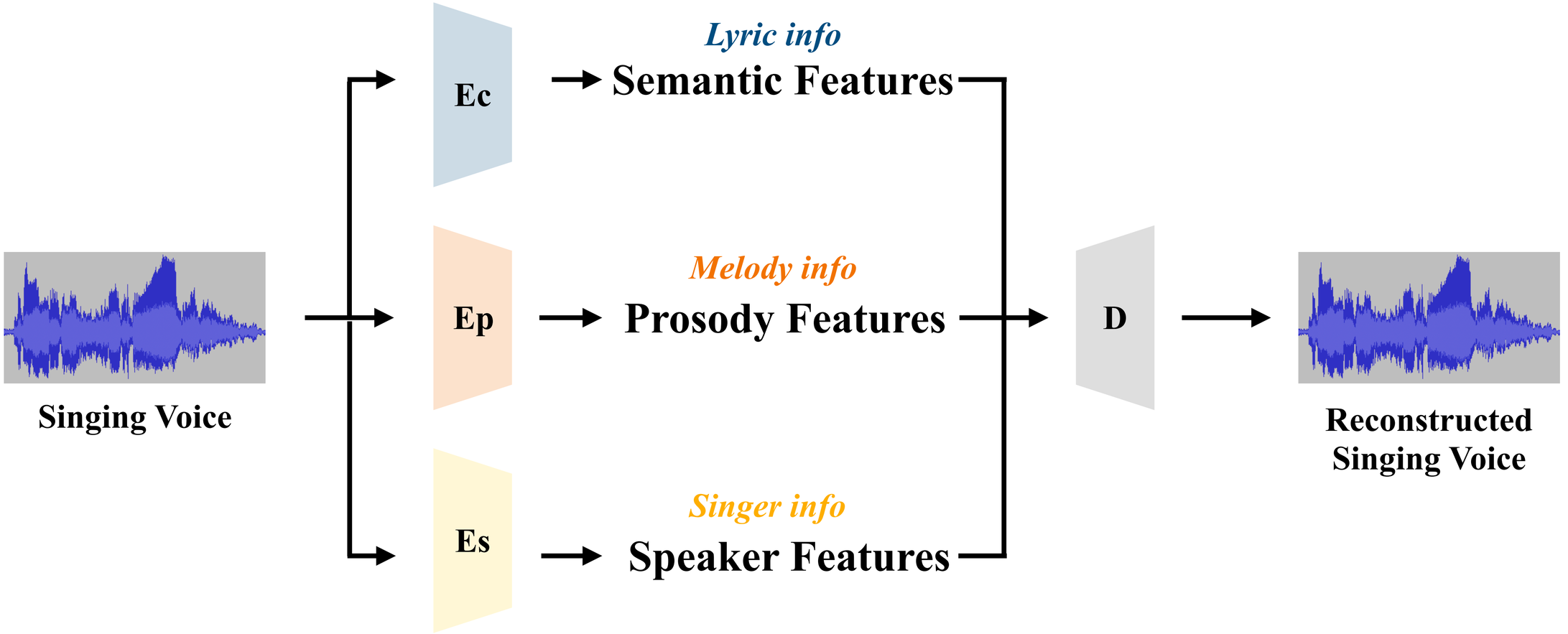
How to decouple the singer identity?

Non-Parallel SVC: GAN School



Credit: Voice Conversion, Hung-yi Lee.

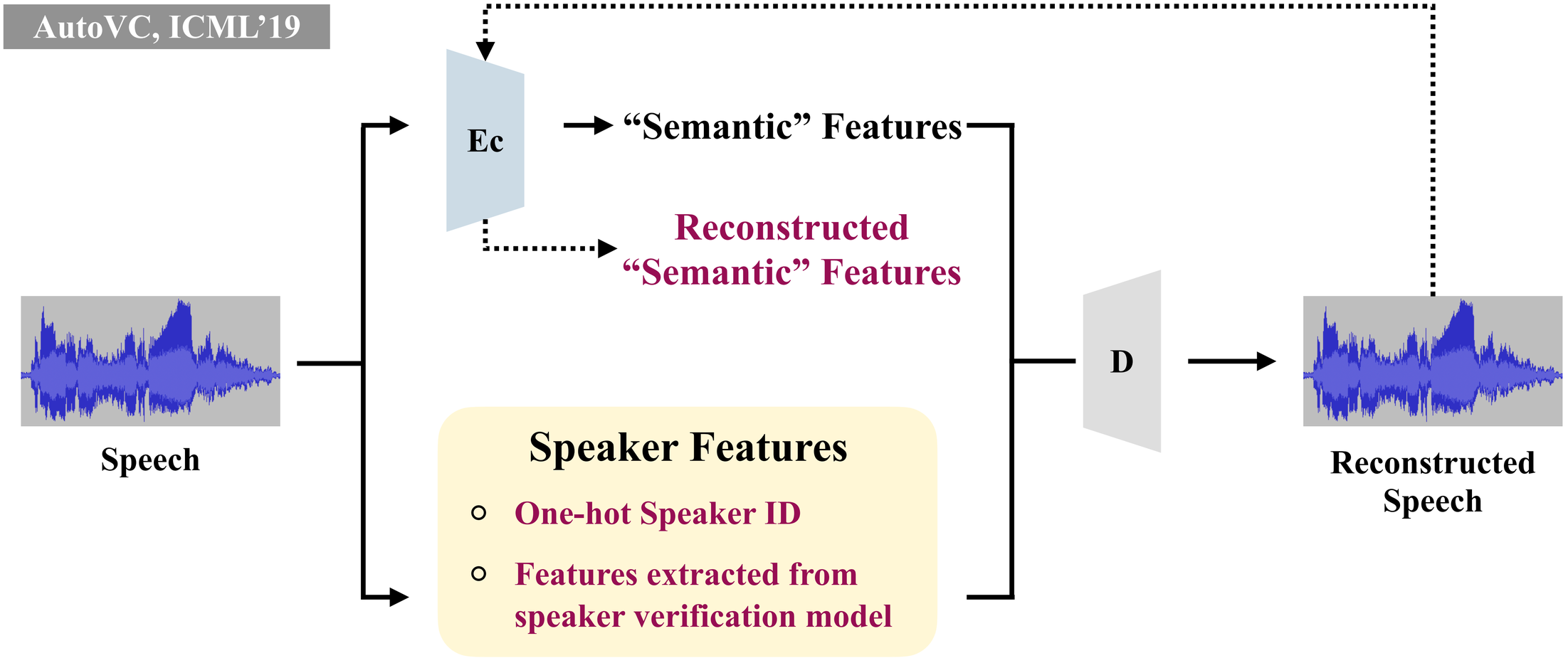
Non-Parallel SVC: Auto-Encoder School



- How to ensure the disentanglement of different features?
- How to ensure there is enough information of each features?

Auto-Encoder VC: The Early Researches

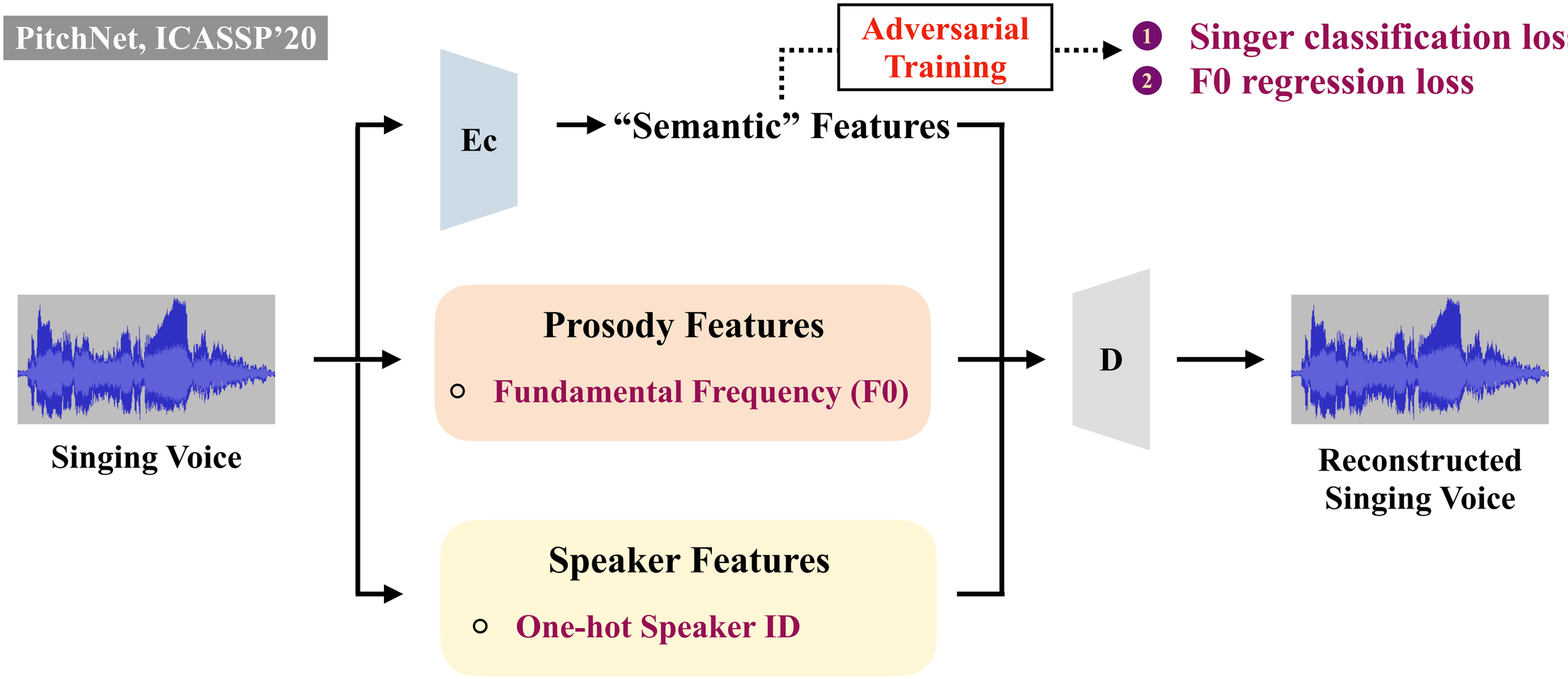
AutoVC, ICML'19



AutoVC: "To carefully design the dimension of the *semantic* features"

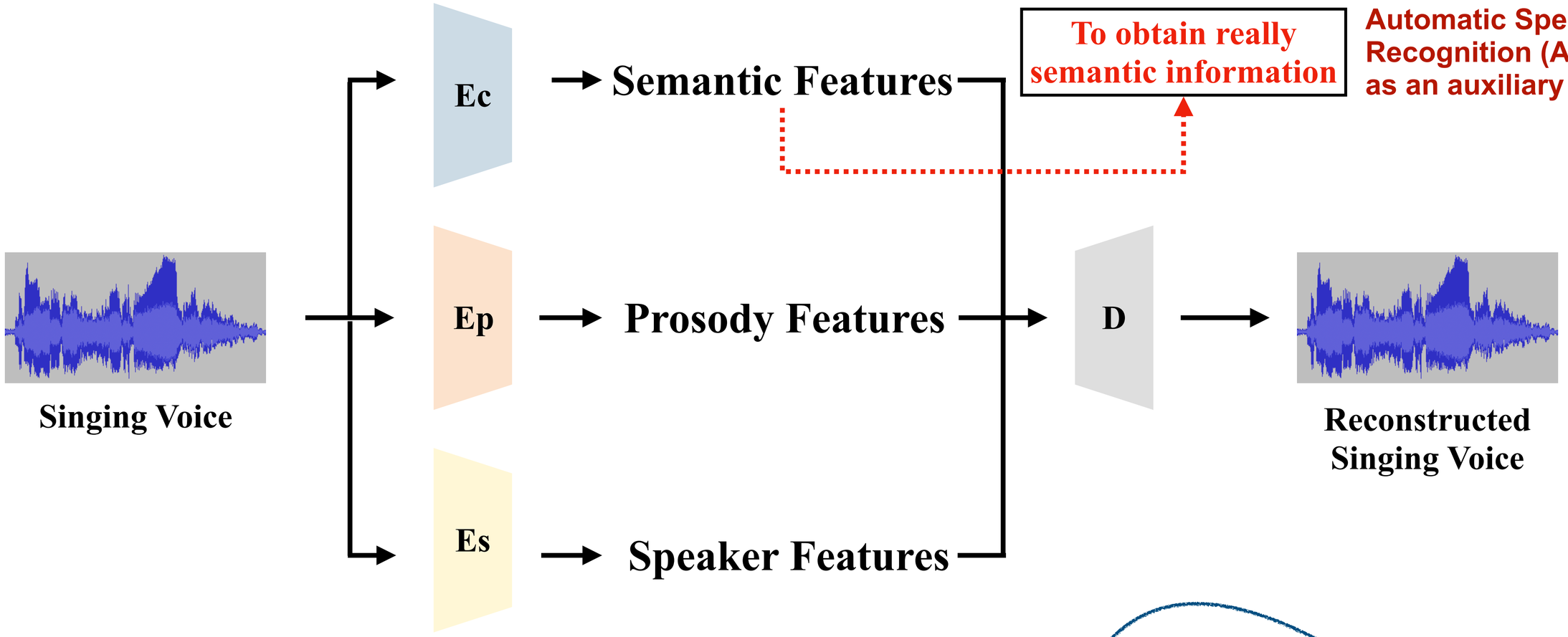
Auto-Encoder SVC: The Early Researches

PitchNet, ICASSP'20



PitchNet: "Adopt adversarial training to disentangle better"

Non-Parallel SVC: Auto-Encoder School

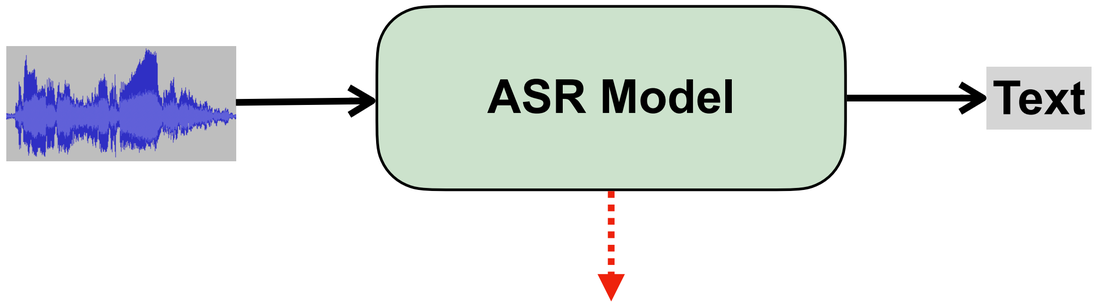


- How to ensure the disentanglement of different features?
- How to ensure there is enough information of each features?

🎯 Solved to some extent

🤔 How to address?

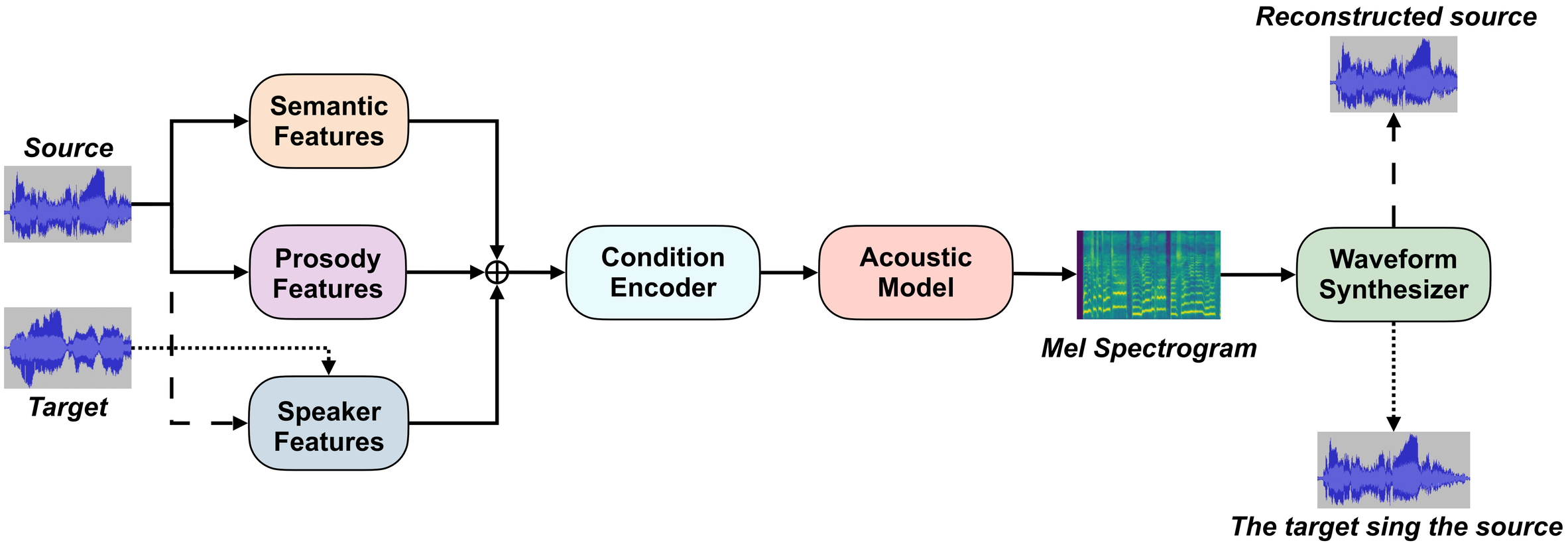
Non-Parallel VC/SVC — a.k.a Recognition & Synthesis VC/SVC



🤔 Why do we use the *dense semantic features* instead of the *symbolic text*?

1. There are errors for the recognized symbolic text.
2. It takes more time to obtain the symbolic text than just extracting dense features.
3. There are more acoustic information (such as pronunciation) in the dense features, which is better for improving the intelligibility of the synthesized voice.

Modern Singing Voice Conversion Pipeline

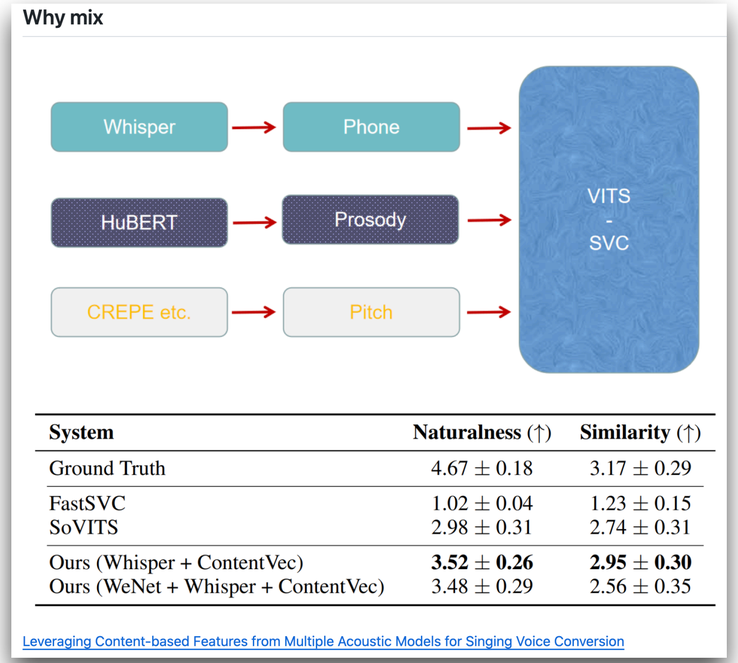


Amphion SVC: Supported Model Architectures

- **Semantic Features Extractor**
  - WeNet, Whisper, ContentVec
  - Joint Usage of Diverse Semantic Features Extractors
- **Prosody Features**
  - F0 and energy
- **Speaker Features**
  - One-hot Speaker ID
  - Features of Pretrained SV model
- **Acoustic Model**
  - Diffusion-based
  - Transformer-based
  - VAE- and Flow-based
- **Waveform Synthesizer**
  - GAN-based
  - Diffusion-based

AI Singer Demo and Impact

- **Make Taylor Swift sing Mandarin song!**

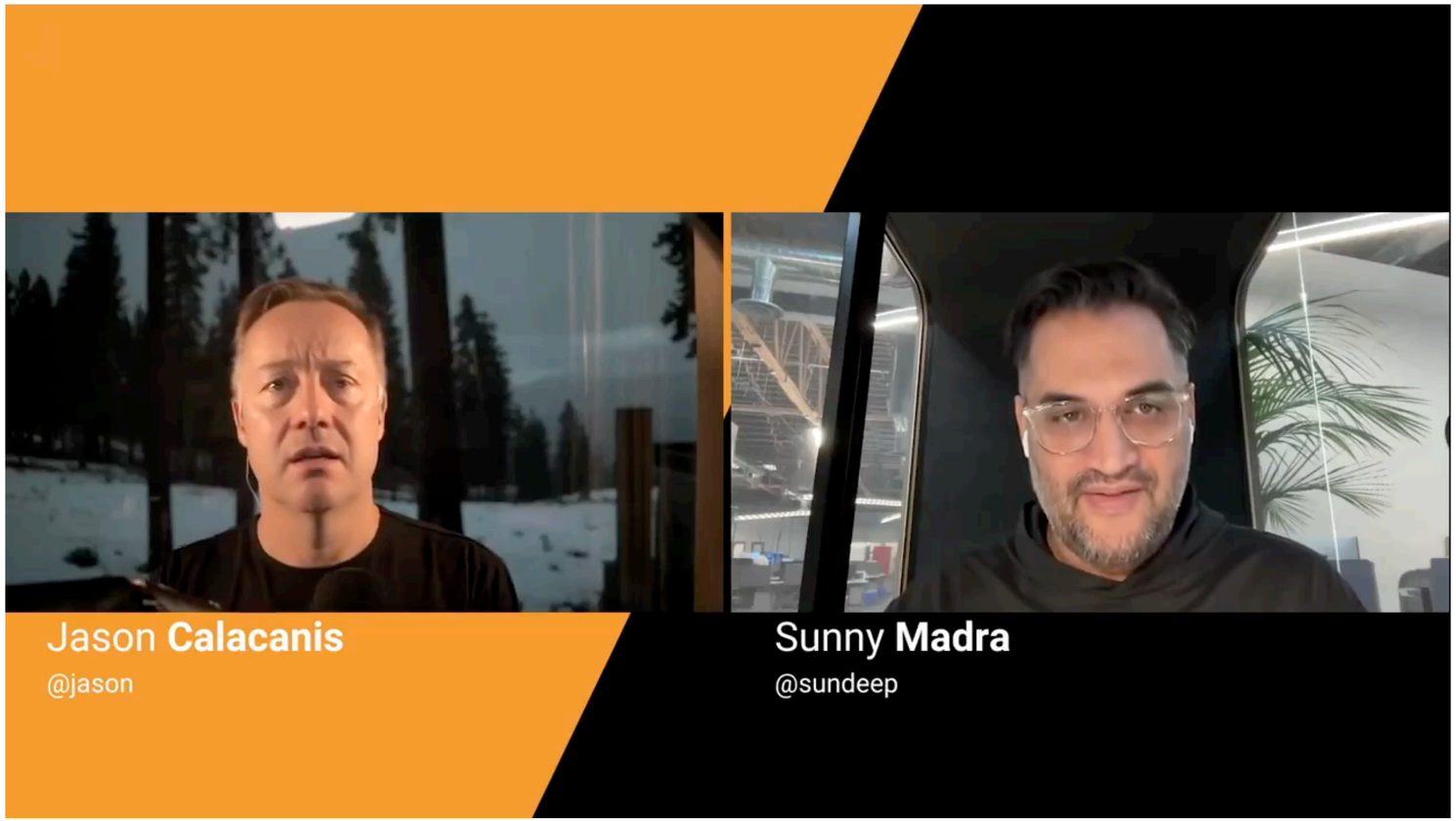


System	Naturalness (↑)	Similarity (↑)
Ground Truth	4.67 ± 0.18	3.17 ± 0.29
FastSVC	1.02 ± 0.04	1.23 ± 0.15
SoVITS	2.98 ± 0.31	2.74 ± 0.31
Ours (Whisper + ContentVec)	**3.52 ± 0.26**	**2.95 ± 0.30**
Ours (WeNet + Whisper + ContentVec)	3.48 ± 0.29	2.56 ± 0.35

Leveraging Content-based Features from Multiple Acoustic Models for Singing Voice Conversion

- **Our idea of using multiple content features has been borrowed and integrated into So-VITS-SVC 5.0 (Github over 2k stars)**

AI Singer Demo and Impact



- **Highly positive comments from the market**

Readings

- Interspeech 2022 TTS tutorial
  - https://github.com/tts-tutorial/interspeech2022/blob/main/INTERSPEECH_Tutorial_VC.pdf
- Singing Voice Conversion
  - https://www.zhangxueyao.com/data/SVC/tutorial.html