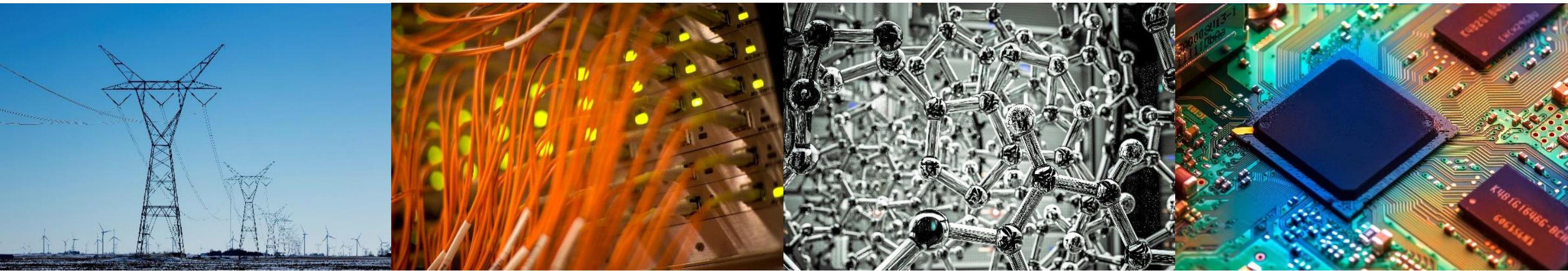


Current Techniques for Voice Conversion

John Harvill
Master Student



I ILLINOIS

Electrical & Computer Engineering

GRAINGER COLLEGE OF ENGINEERING

Outline

- What is voice conversion?
- Parallel vs. non-parallel data
- Parts of a voice conversion system
- Speaker-independent content features
- Speaker representation features
- Audio synthesis
- Examples of VC systems (AutoVC, Speaker-dependent wavenet, my recent work)

What is voice conversion?

- Voice conversion (VC) is the process of taking a speech utterance from one speaker and transforming it to sound as if it was uttered by a different speaker.
- VC is effectively a style transfer problem, where the content is whatever is being said and the style is the way in which it is said, i.e. who is saying it.

Parallel vs. non-parallel data

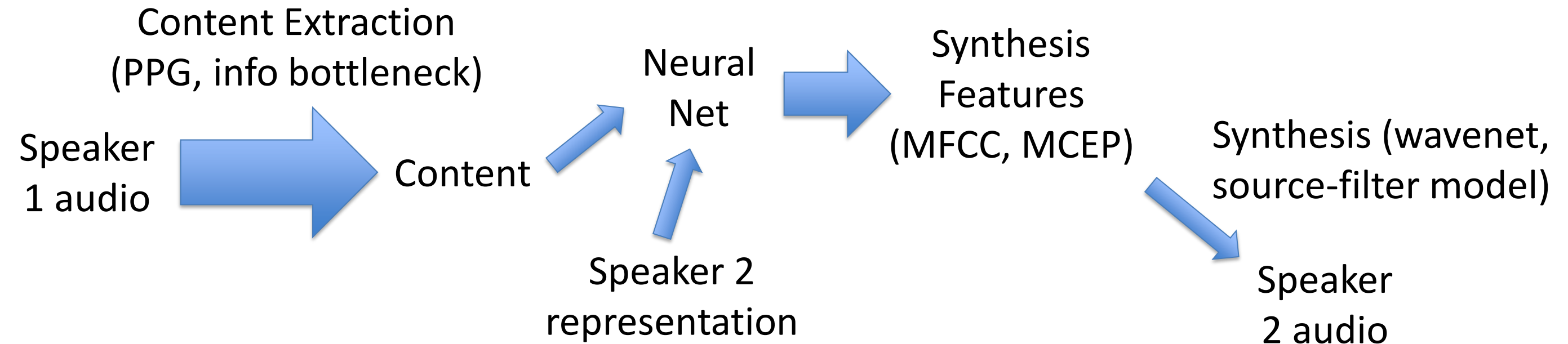
- Parallel data consists of the speakers uttering the same sentences. VC systems trained with parallel data are not ideal because parallel data is hard to obtain.
- Non-parallel data does not have the requirement that the speakers utter the same sentences. VC systems that use non-parallel data are preferred.

Parts of a voice conversion system

- The goal of VC is to extract a speaker-independent form of content from an utterance and synthesize audio with that content using a different style (speaker)
- Content extraction techniques include Phonetic Posteriorgrams (PPGs) and information bottlenecks in autoencoders

Parts of a voice conversion system

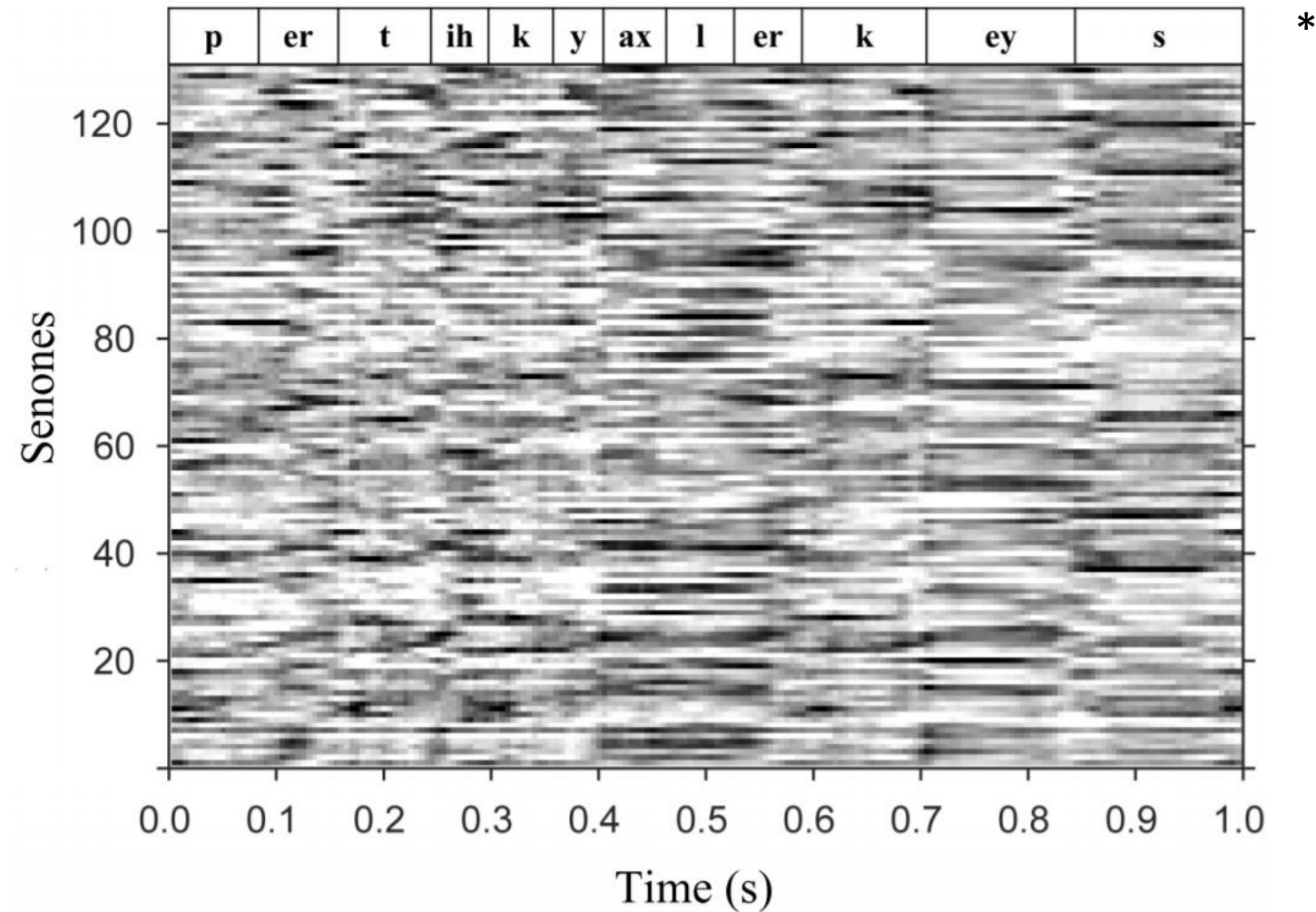
- Speaker representation techniques include one-hot vectors and speaker embeddings (zero-shot conversion)
- Audio synthesis techniques include wavenet vocoders and source-filter model



Speaker-independent content features

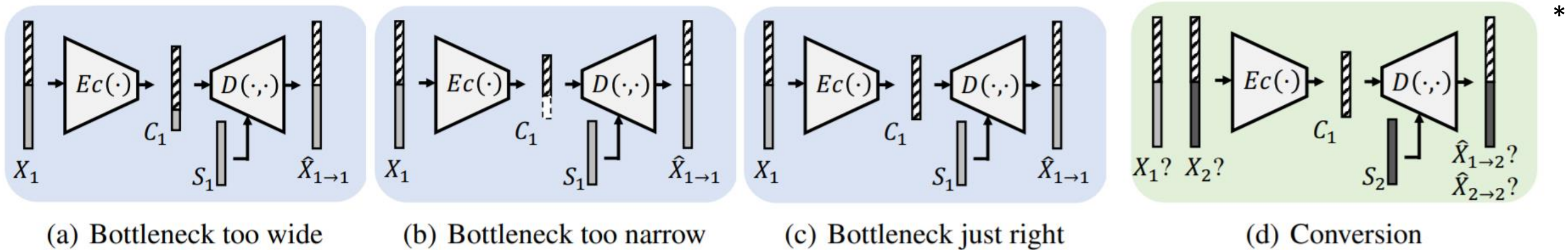
- Phonetic Posteriorgrams (PPGs) represent the probability of each phonetic class for each frame of the audio sample.* PPGs are speaker independent meaning that time-aligning the utterances of two different speakers saying the same thing would ideally result in identical PPGs.

Speaker-independent content features



Speaker-independent content features

- Information bottlenecks in autoencoders provide a convenient way to let a neural net determine the content.



- Note that the hidden content representation is not interpretable whereas a PPG is interpretable.

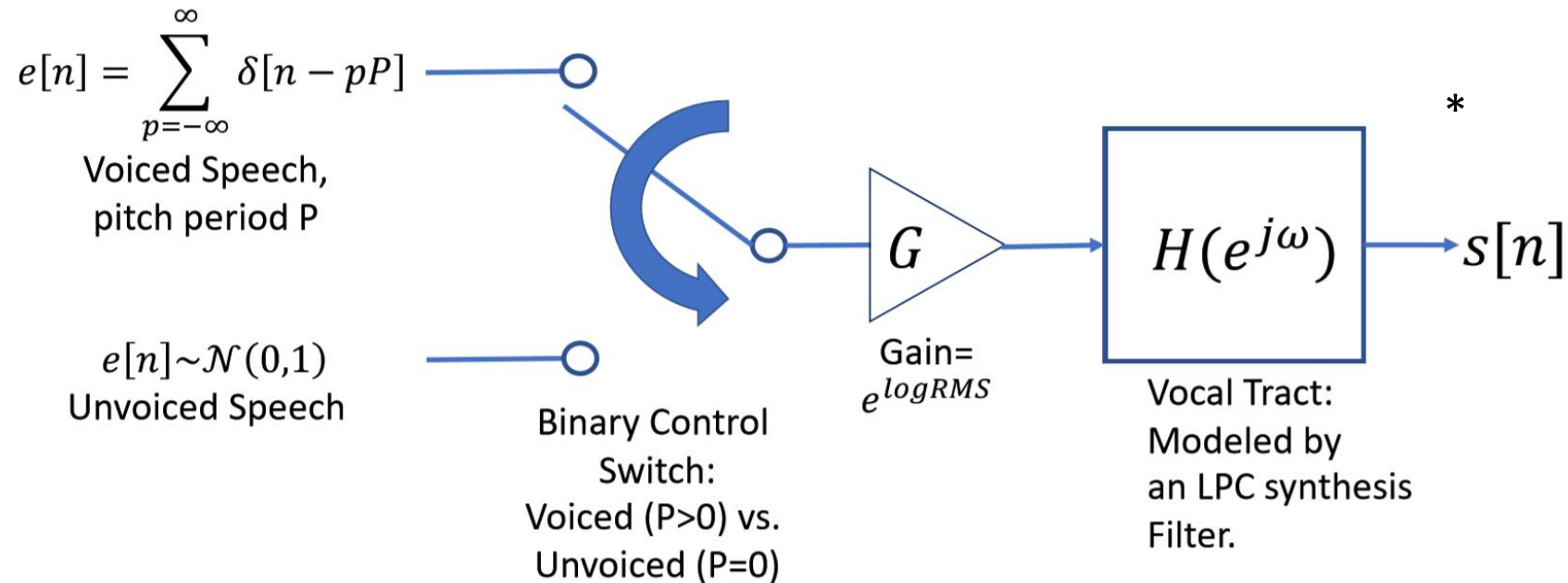
Speaker representation features

- One-hot vectors are used when conversion is only intended to be done between speakers in the training dataset.
- Speaker embeddings are used when conversion is intended to be done between any speaker. Speaker embeddings give a generalized representation of the speaker such that speakers don't need to be in the training dataset to be used for conversion.

Audio synthesis

- The output of the voice conversion neural network can be a set of synthesis features such as MFCCs or Mel-cepstral coefficients which are then converted to an audio signal. Other systems directly train to directly output an audio signal.
- Wavenet can be directly tied into a network or can be conditioned on MFCCs to produce audio. Mel-cepstral coefficients and excitation signals can be used to produce audio with a source-filter model.

Source-filter model



- Mel log spectral approximation (MLSA) filter used in practice due to low sensitivity in the filter coefficients (calculated from mel cepstrum coefficients)

Source-filter model

- Mel Cepstrum

$$H(z) = \exp \sum_{m=0}^M c_{\alpha}(m) \tilde{z}^{-m} \quad *$$

$$\tilde{z}^{-1} = \frac{z^{-1} - \alpha}{1 - \alpha z^{-1}}$$

- Newton-Raphson method

Source-filter model

- MLSA filter

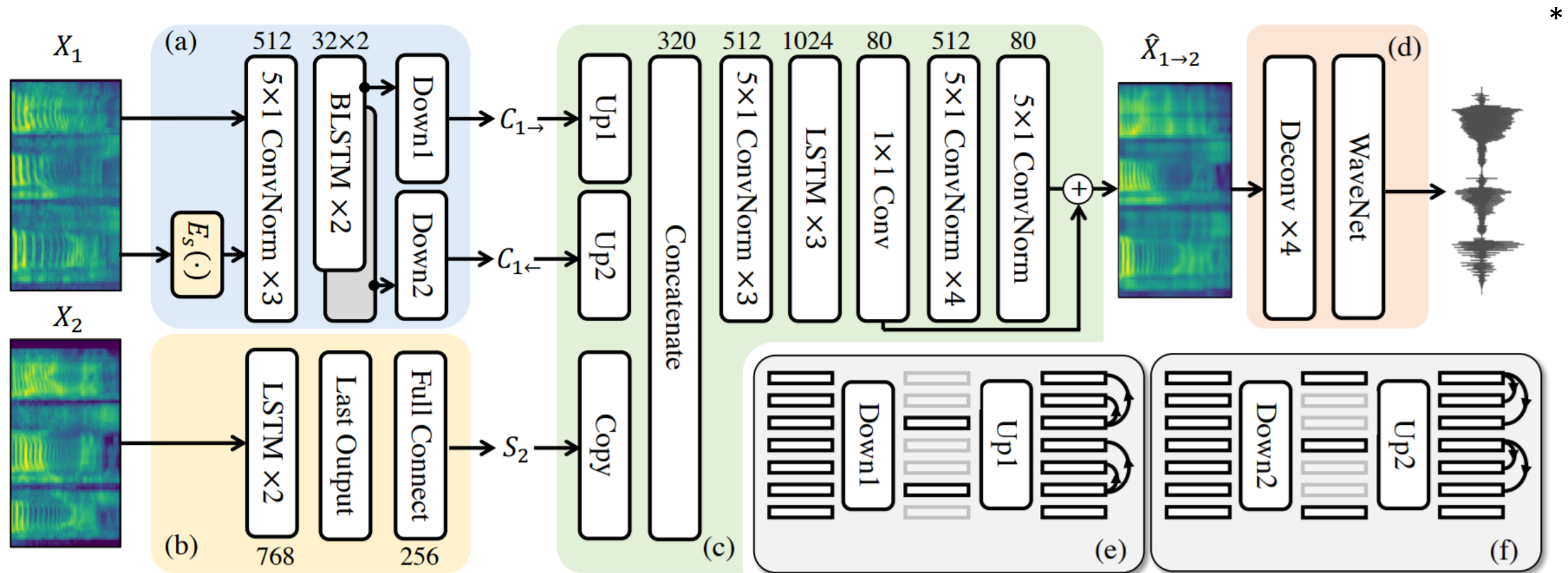
$$F(z) = \sum_{m=1}^M b(m) \Phi_m(z) \quad *$$

$$\Phi_m(z) = \begin{cases} 1, & m = 0 \\ \frac{(1 - \alpha^2)z^{-1}}{1 - \alpha z^{-1}} \tilde{z}^{-(m-1)}, & m \geq 1 \end{cases}$$

$$b(m) = \begin{cases} c_\alpha(M), & m = M \\ c_\alpha(m) - \alpha b(m+1), & 0 \leq m < M \end{cases}$$

Examples of VC systems

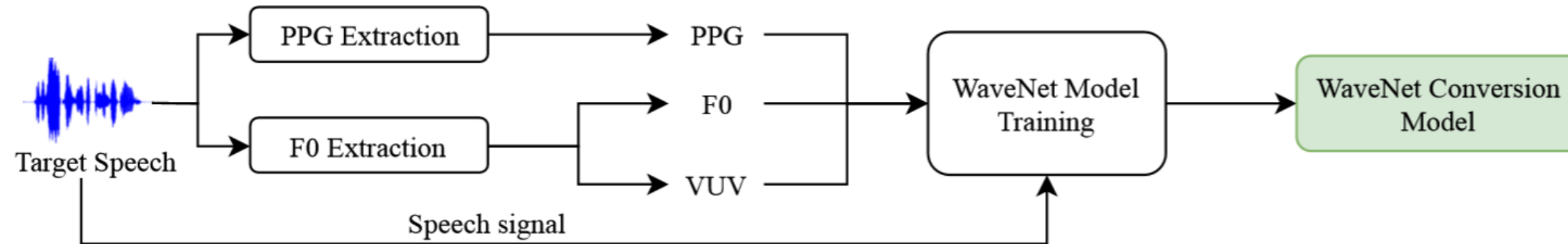
- AutoVC



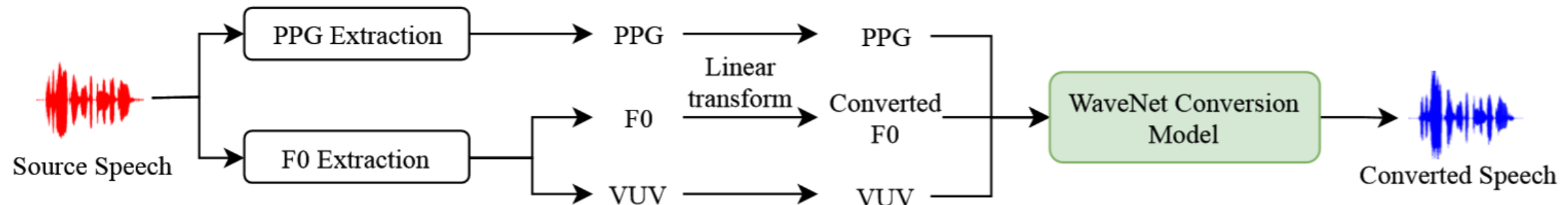
Examples of VC systems

- Speaker-dependent wavenet

(a) WaveNet conversion model training



(b) Run-time conversion



Note on Linear Transform

- The linear transform* used to convert F0 between speakers is given as:

$$f0'_y = \exp((\log f0_x - \mu_x) \frac{\sigma_y}{\sigma_x} + \mu_y)$$

- μ_x and σ_x are the mean and variance of the input audio, and μ_y and σ_y are the mean and variance of the target speaker over the training data

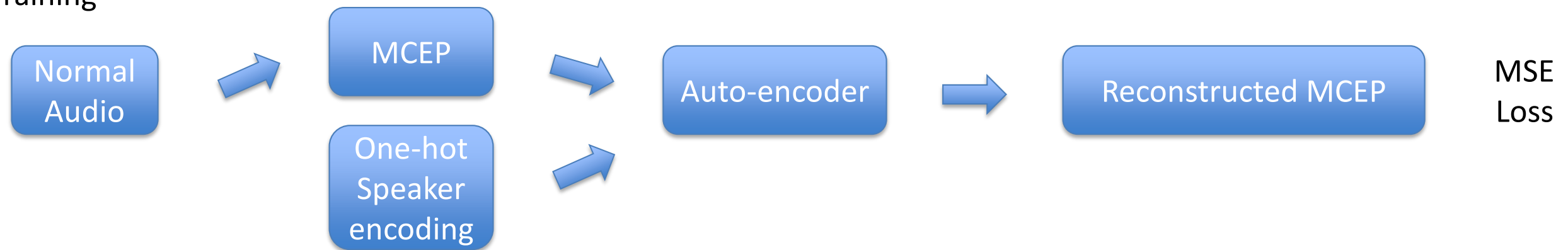
Examples of VC systems

- Speaker and speech type conversion using source-filter model and autoencoders
- Goal: Convert normal speech from speaker A to whispered speech of speaker B
- Future work: Improve quality of conversion of normal speech from speaker A to normal speech of speaker B, conversion of whispered speech of speaker A to whispered speech of speaker B

Examples of VC systems

- Speaker and speech type conversion using source-filter model and autoencoders

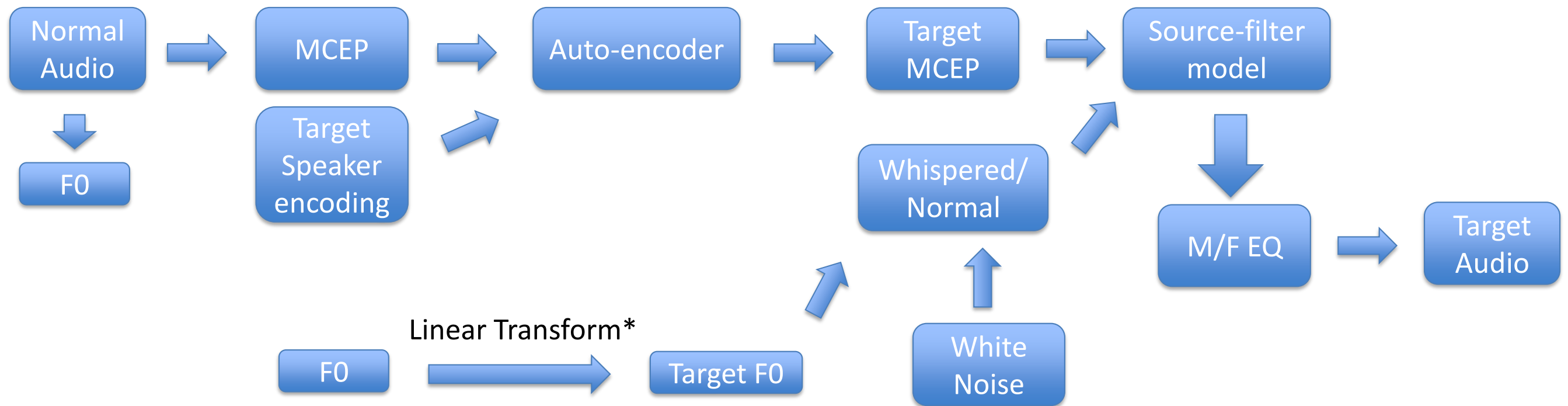
Training



Examples of VC systems

- Speaker and speech type conversion using source-filter model and autoencoders

Conversion



Benefits of recent work and audio examples

- Computationally cheap
- Parallel data not required
- Whispered data not required

Source (male, normal)



Target (female, whispered)



Source (male, whispered)



Target (female, normal)



References

- Tian, Xiaohai & Chng, Eng & Li, Haizhou. (2019). A Speaker-Dependent WaveNet for Voice Conversion with Non-Parallel Data. 201-205. 10.21437/Interspeech.2019-1514.
- Qian, Kaizhi & Zhang, Yang & Chang, Shiyu & Yang, Xuesong & Hasegawa-Johnson, Mark. (2019). AutoVC: Zero-Shot Voice Style Transfer with Only Autoencoder Loss.
- Sun, Lifa & Li, Kun & Wang, Hao & Kang, Shiyin & Meng, Helen. (2016). Phonetic posteriorgrams for many-to-one voice conversion without parallel data training. 1-6. 10.1109/ICME.2016.7552917.