Voice Conversion

Voice Conversion

I. **Introduction to Voice Conversion**
   – Speech Synthesis Context (TTS)
   – Overview of Voice Conversion
II. **Spectrum Transformation in VC**
   – Gaussian Mixture Model
III. **Conversion Results**
   – Objective Metrics & Subjective Evaluations
   – Sound Samples
IV. **Summary & Conclusions**

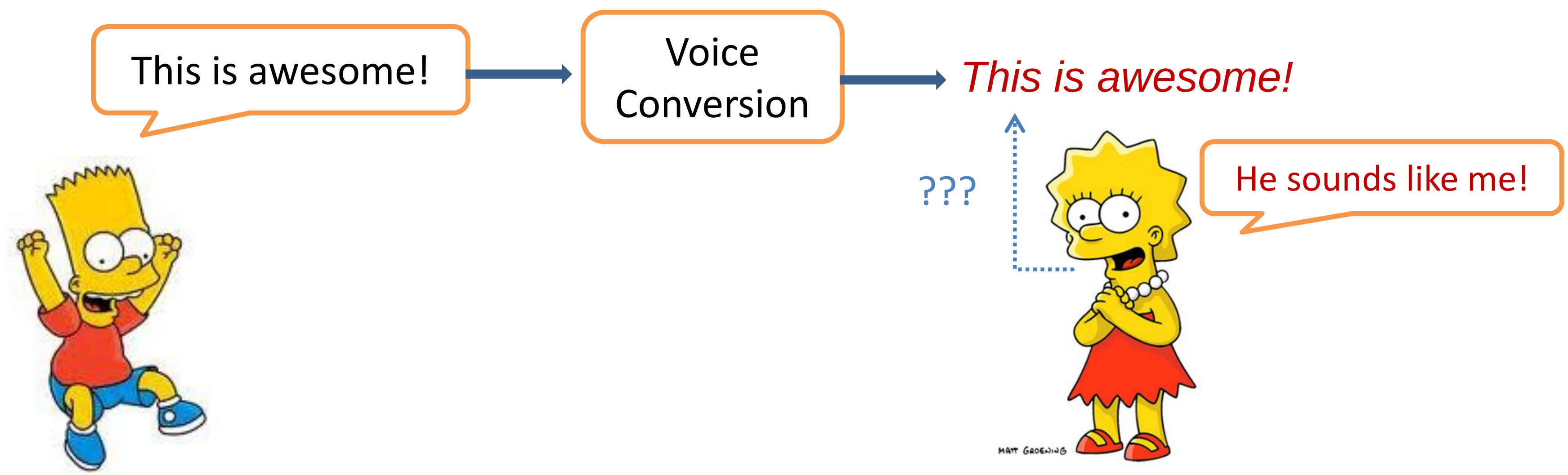
Voice Conversion

I. Introduction to Voice Conversion

- Speech Synthesis Context (TTS)
- Overview of Voice Conversion

Voice Conversion (VC)

- Transform the speech of a (source) speaker so that it sounds like the speech of a different (target) speaker.



source target

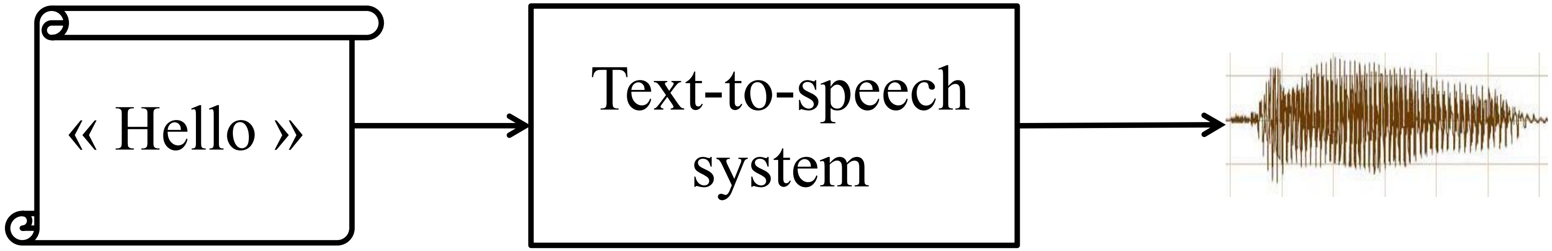
Context: Speech Synthesis

- **Increase in applications using speech technologies**
  - Cell phones, GPS, video gaming, customer service apps…

Information communicated through speech!

- **Text-to-Speech (TTS) Synthesis**
  - Generate speech from a given text

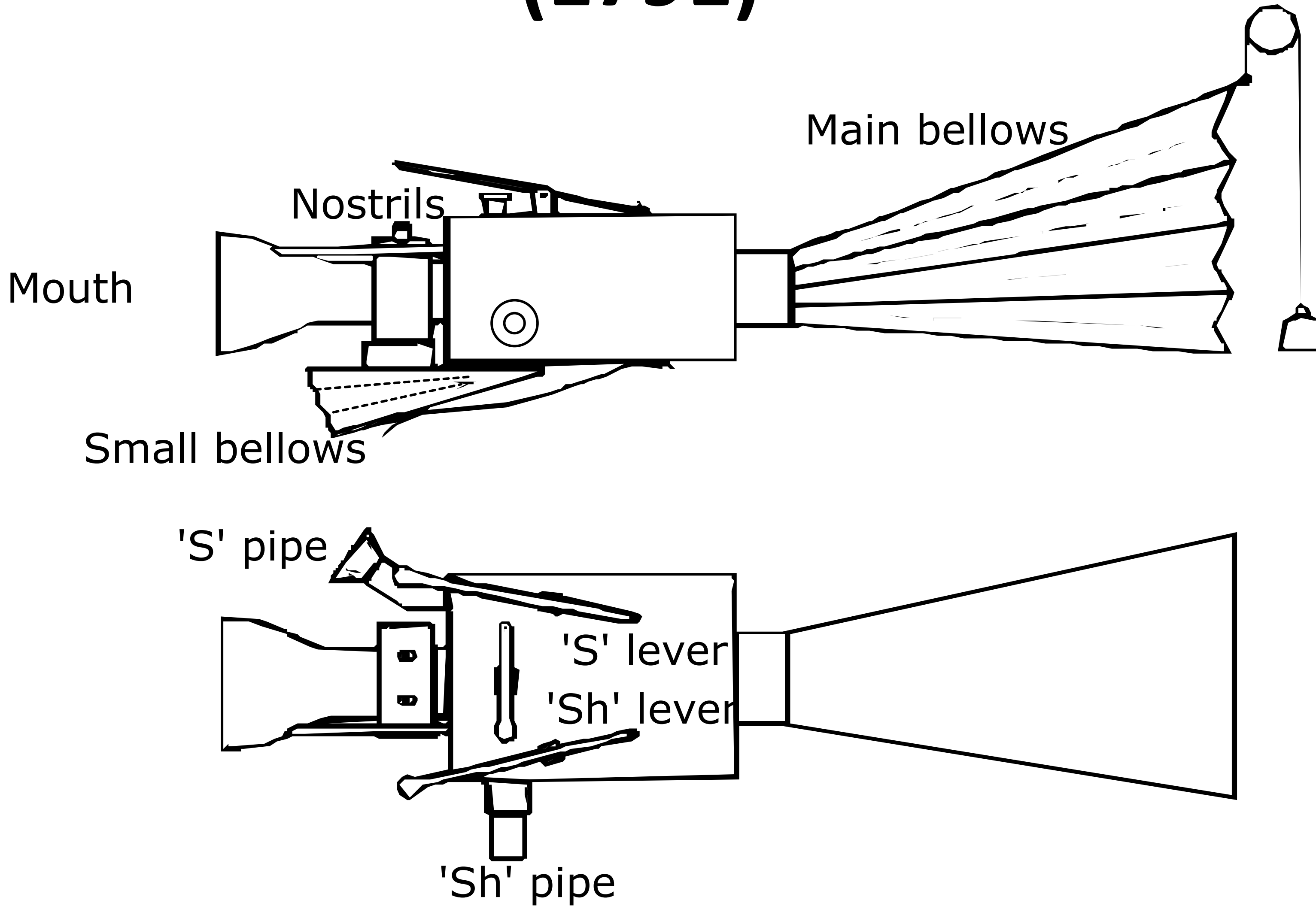
Speech Synthesis



GOAL :

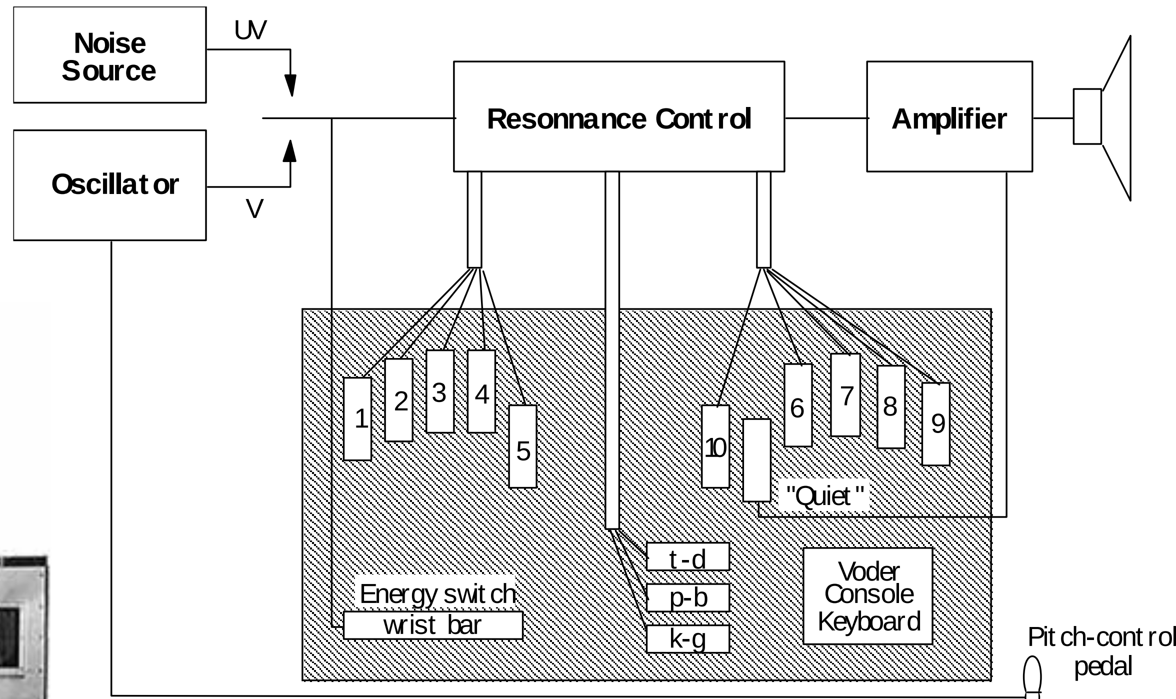
Produce the lecture of an unknown text typed by the user

Von Kempelen's talking machine (1791)

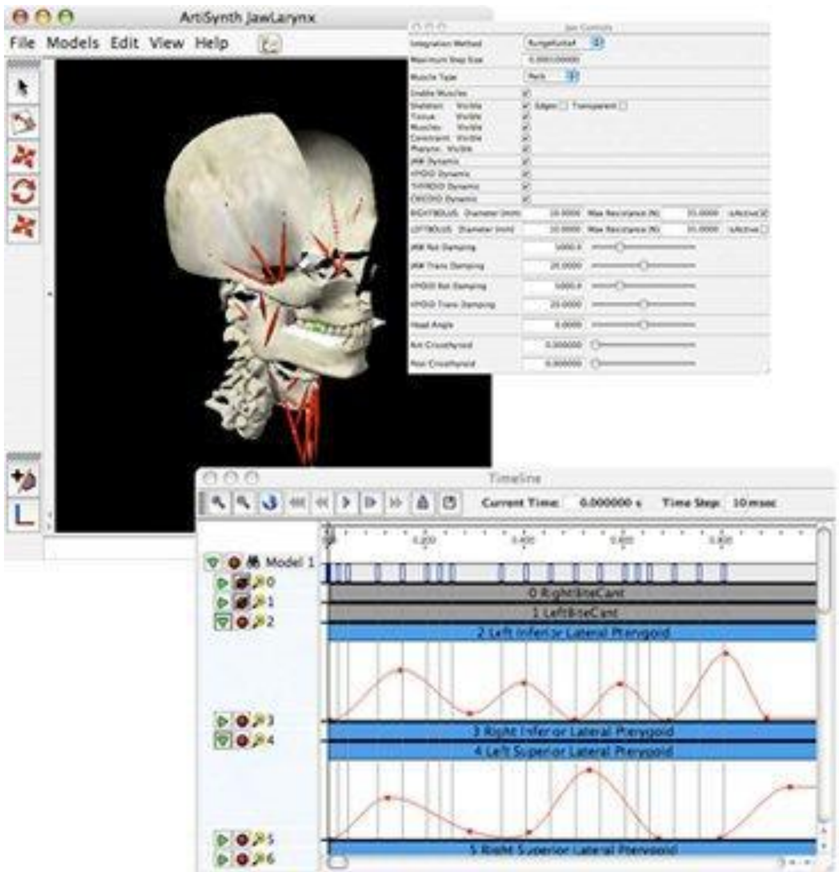


Omer Dudley's Voder

(Bell Labs, 1936)



And other developments in articulatory synthesis



- Work by :
  - K. Stevens, G. Fant, P. Mermelstein, R. Carré (*GNUSpeech*), S. Maeda, J. Shroeter & M. Sondhi…
- More recently :
  - O. Engwall, S. Fels (*ArtiSynth*), Birkholz and Kröger, A. Alwan & S. Narayanan (MRI)…

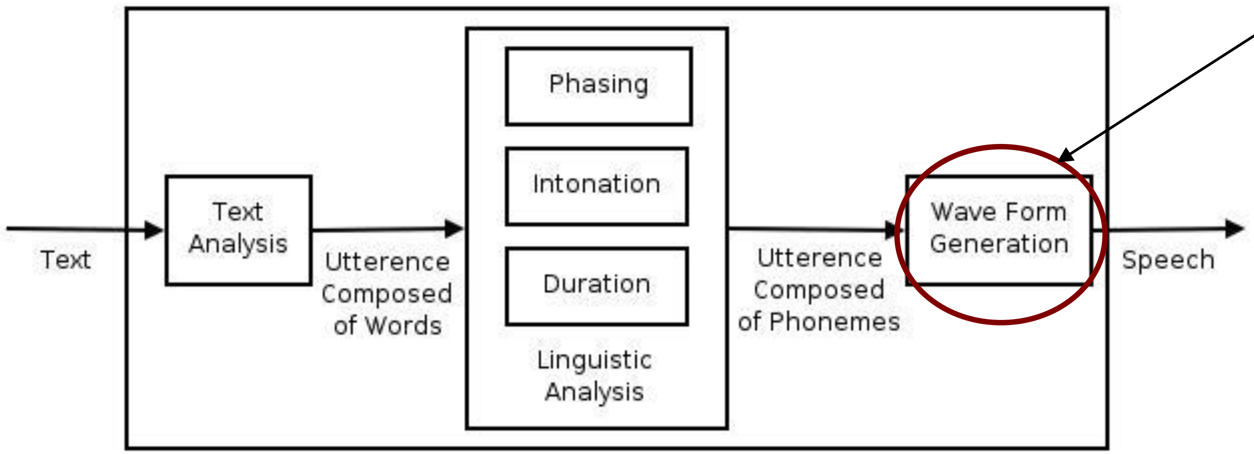
Text-to-Speech (TTS) Systems

- ## TTS Approaches

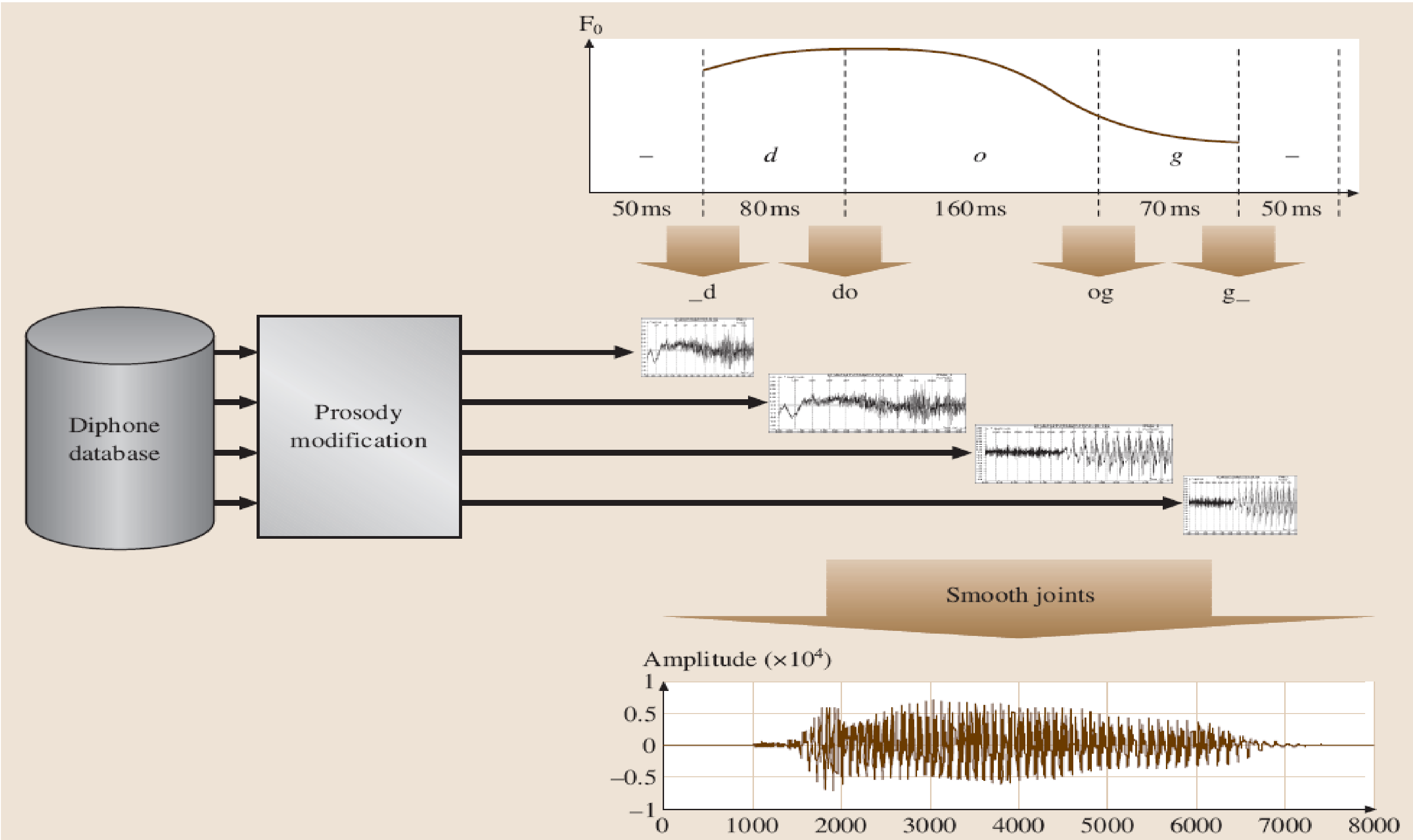
1. **Concatenative:** speech synthesized from recorded segments
   - Unit-Selection: parts of speech chosen from corpora & strung together
     High-quality synthesis, but need to record & process corpora

2. **Parametric:** speech generated from model parameters
   - HMM-based: speaker models built from speech using linguistic info
     Limited quality due to simplified speech modeling & statistical averaging

Concatenative or Paramateric?

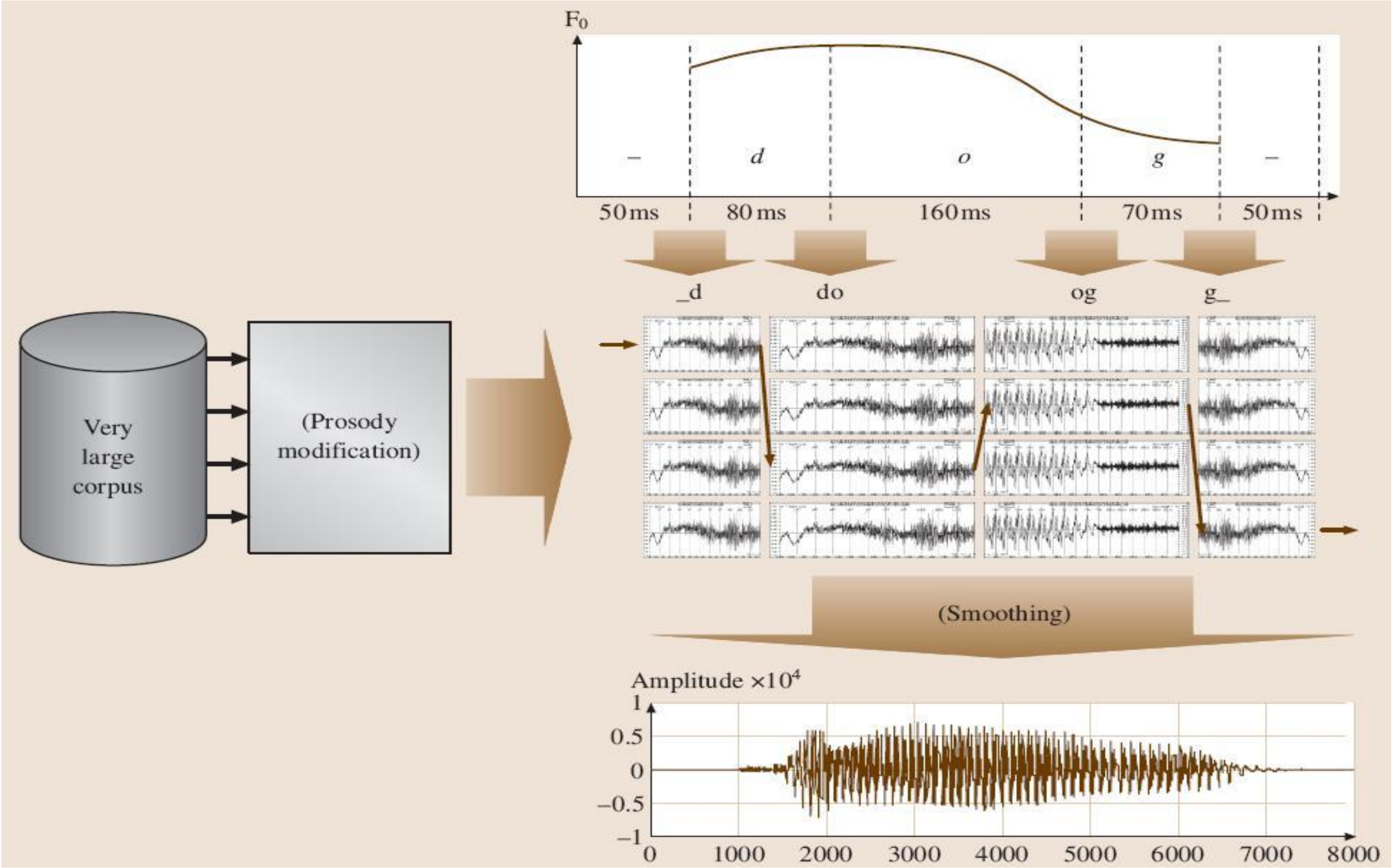


Diphone concatenation



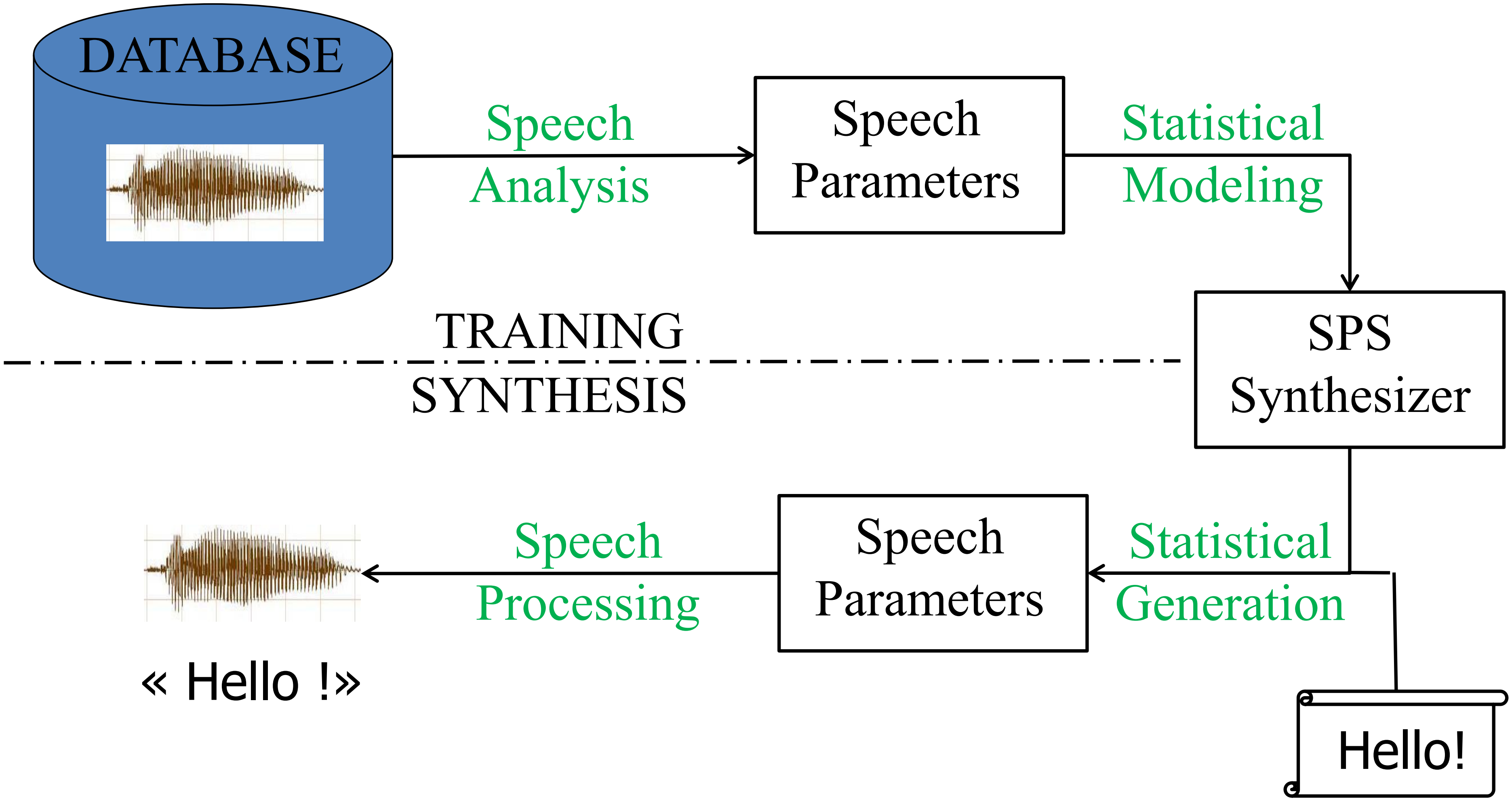
Intelligibility ✓ Naturalness ~ Mem/CPU/Voices ✓ Expressivity ✗

Unit selection



Intelligibility ✓ **Naturalness** ✓ **Mem/CPU/Voices** ~ **Expressivity** ~

Statistical Parametric Speech Synthesis



Voice Conversion: TTS Motivation

- Concatenative speech synthesis
  – High-quality speech
  – But, need to record & process a large corpora for each voice
- Voice Conversion
  – Create different voices by speech-to-speech transformation
  – Focus on acoustics of voices

What gives a *voice* an *identity*?

- “Voice”→ notion of identity (*voice* rather than *speech*)

- Characterize speech based on different levels

1. Segmental
   - Pitch – fundamental frequency
   - Timbre – distinguishes between different types of sounds

2. Supra-Segmental
   - Prosody – intonation & rhythm of speech

Goals of Voice Conversion

1. Synthesize High-Quality Speech
   – Maintain quality of source speech (limit degradations)

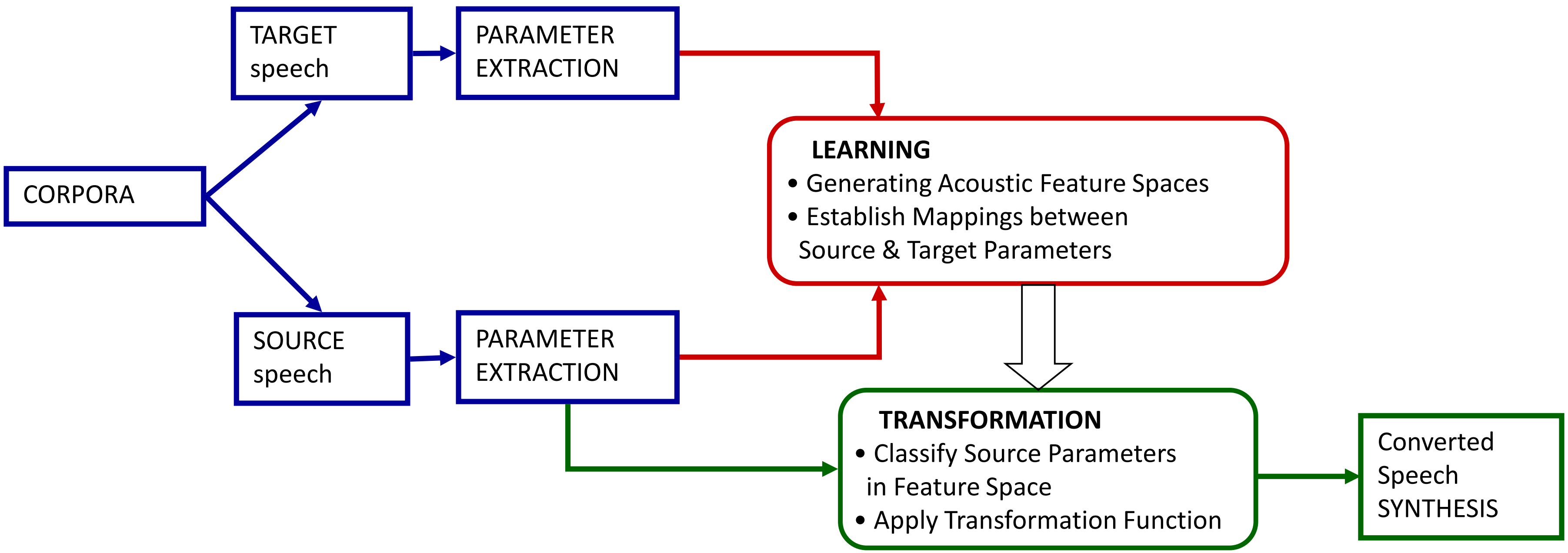
2. Capture Target Speaker Identity
   – Requires learning between source & target features

Difficult task!

– significant modifications of source speech needed that risk severely degrading speech quality…

Stages of Voice Conversion

1) Analysis, 2) Learning, 3) Transformation



- Key Parameters: the spectrum and prosody

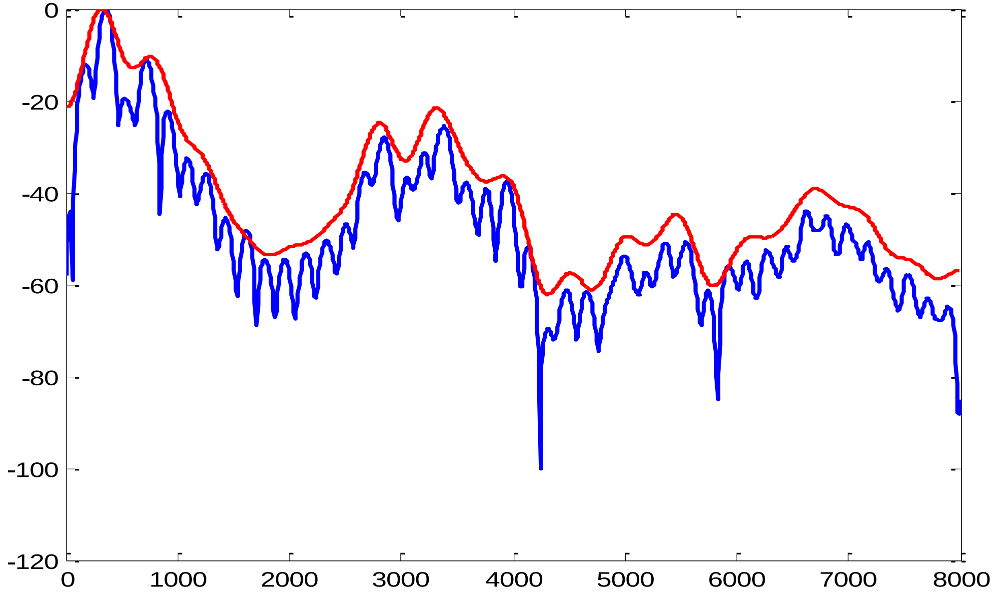
Voice Conversion

I. **Introduction to Voice Conversion**
   – Speech Synthesis Context (TTS)
   – Overview of Voice Conversion

II. **Spectrum Transformation in VC**
   – Gaussian Mixture Model

The Spectral Envelope

- Spectral Envelope: curve approximating the DFT magnitude



- Related to voice timbre, plays a key role in many speech applications:
  - Coding, Recognition, Synthesis, Voice transformation/conversion
- Voice Conversion: important for both speech quality and voice identity

Spectral Envelope Parameterization

- Two common methods

  1) **Cepstrum**

     - Discrete Cepstral Coefficients

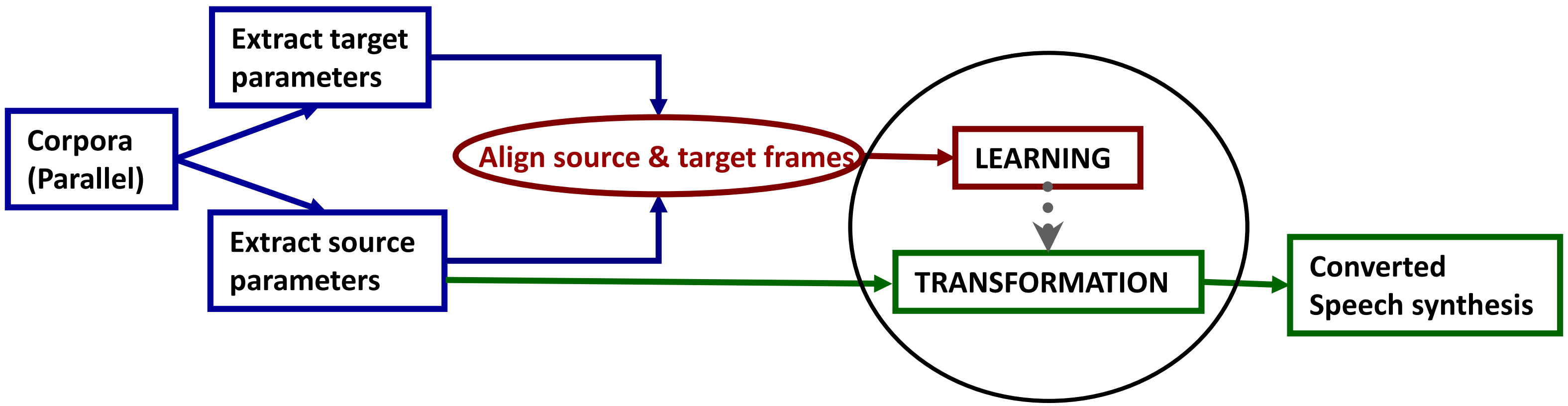
     - Mel-Frequency Cepstral Coefficients (MFCC)

       → change the frequency scale to reflect bands of human hearing

  2) **Linear Prediction (LP)**

     - Line Spectral Frequencies (LSF)

Standard Voice Conversion



Focus: Learning & Transforming the Spectral Envelope

- **Parallel corpora**: source & target utter same sentences
- Parameters are spectral features (e.g. vectors of cepstral coefficients)
- Alignment of speech frames in time

→ Standard: Gaussian Mixture Model

Voice Conversion

I. **Introduction to Voice Conversion**
   - Speech Synthesis Context (TTS)
   - Overview of Voice Conversion

II. **Spectrum Transformation in VC**
   - Standard: Gaussian Mixture Model
     - Formulation
     - Limitations
       1. Acoustic mappings between source & target parameters
       2. Over-smoothing of the spectral envelope

GMM-based VC

1. Start form Minimum Mean Square Estimation (MMSE)
2. Time alignment
3. To derive the transfer function of GMM based VC.

Mean-Square Estimation(1/4)

- 如用一個 constant c 去 estimate RV $\mathbf{y}$，以 MS estimation (i.e.，mean-square error 為最小之 estimation) 可如下推導

$$e = E\left\{(\mathbf{y}-c)^2\right\} = \int_{-\infty}^{\infty}(y-c)^2 f(y)dy$$

$$\frac{de}{dc} = -\int_{-\infty}^{\infty} 2(y-c) f(y)dy = 0$$

$$c = \int_{-\infty}^{\infty} yf(y)dy = E\{\mathbf{y}\}$$

Mean-Square Estimation(2/4)

- 現在考慮 nonlinear MS estimation 由一個 RV $\mathbf{x}$ 去估計另一個 RV $\mathbf{y}$

$$e = E_{xy}\left\{\left[\mathbf{y} - c(\mathbf{x})\right]^2\right\}$$

$$= \iint (y - c(x))^2 f(x, y)dxdy$$

$$= \int f(x)\left[\int (y - c(x))^2 f(y|x)dy\right]dx$$

$\because [\bullet]$ 為正，$f(x)$ 為正，所以只要 $[\bullet]$ 中之 $c(x)$ 使得$[\bullet]$為最小 for every given x，then e is minimum (i.e., 本來是 $\int f(x)[\bullet]dx$ 合起來考慮時要 minimum，但它等同於對每一 x, $[\bullet]$ 皆 minimum 即可)

Mean-Square Estimation(3/4)

$\therefore$ 要minimum $[\bullet]$ for each given x，而 $c(x)$ 為一deterministic

(constant) when x is given，$\therefore$ 由前面 case 和 $c(x)=E_y[\mathbf{y}\,|\,x]$，

再將 $\mathbf{x}$ 可改變考慮進去，上式變為 $c(\mathbf{x})=E_y[\mathbf{y}\,|\,\mathbf{x}]$

- 如 RVs $\mathbf{y}$ 和 $\mathbf{x}$ 為 independent，則 $E_y[\mathbf{y}\,|\,\mathbf{x}]=E_y[\mathbf{y}]=$ constant

Mean-Square Estimation(4/4)

1 mixture Gaussian, assume x_t and y_t are joint Gaussian, source x_t follow a Gaussian distribution.

By using MMSE, conversion function is

$$\hat{y}_t = F(x_t) = E[y_t \mid x_t] = v + \Gamma \Sigma_{xx}^{-1}(x_t - \mu_x)$$

where $v = \mu_y$, and $\Gamma = \Sigma_{xy}$

Gaussian Mixture Model (GMM) for VC

- **Origins:**
  - Evolved from "fuzzy" Vector Quantization (*i.e.* VQ with "soft" classification)
  - Originally proposed by [Stylianou et al; 98]
  - Joint learning of GMM (most common) by [Kain et al; 98]
- **Underlying principle:**
  - Exploit joint statistics exhibited by aligned source & target frames
- **Methodology:**
  - Represent distributions of spectral feature vectors as mix of Q Gaussians
  - Transformation function then based on MMSE criterion

Preliminaries of GMM

- We assume that the dataset *X* has been generated by a *parametric* distribution *p(X)*.
- Estimation of the parameters of *p* is known as *density estimation*.
- We consider Gaussian distribution.

Typical parameters (1)

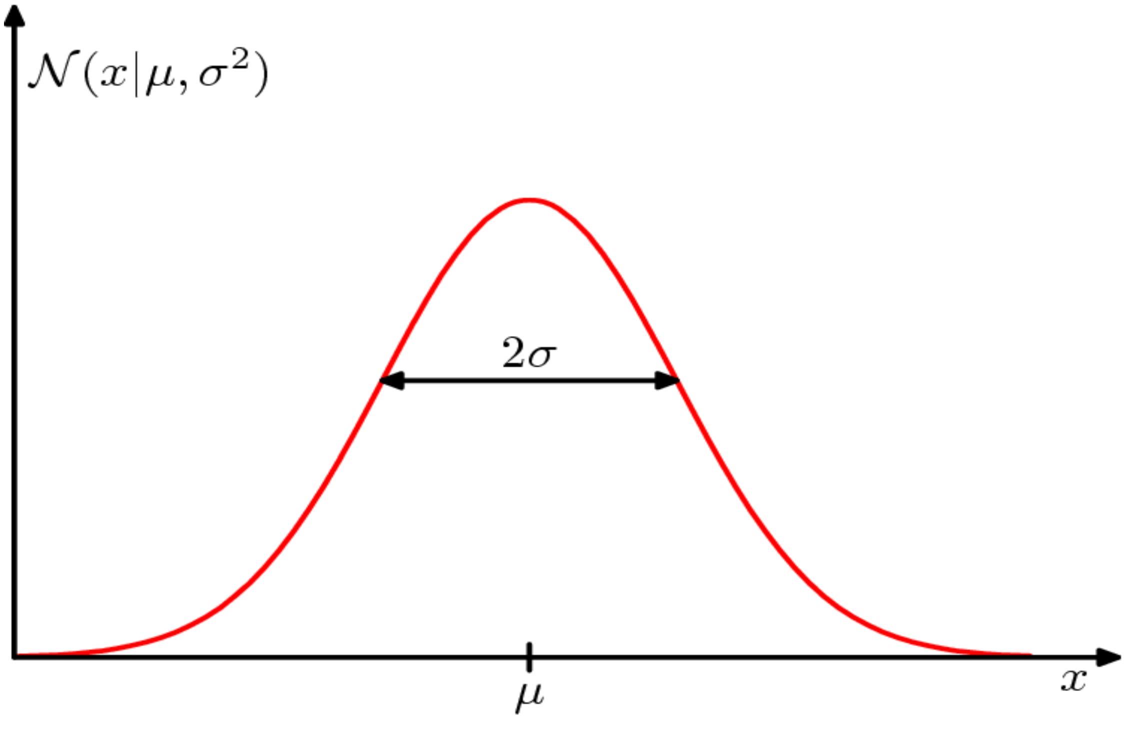
- *Mean* (μ): average value of $p(X)$, also called expectation.
- *Variance* (σ): provides a measure of variability in $p(X)$ around the mean.

Typical parameters (2)

- *Covariance*: measures how much two variables vary together.
- *Covariance matrix*: collection of covariances between all dimensions.
  - Diagonal of the covariance matrix contains the variances of each attribute.

One-dimensional Gaussian

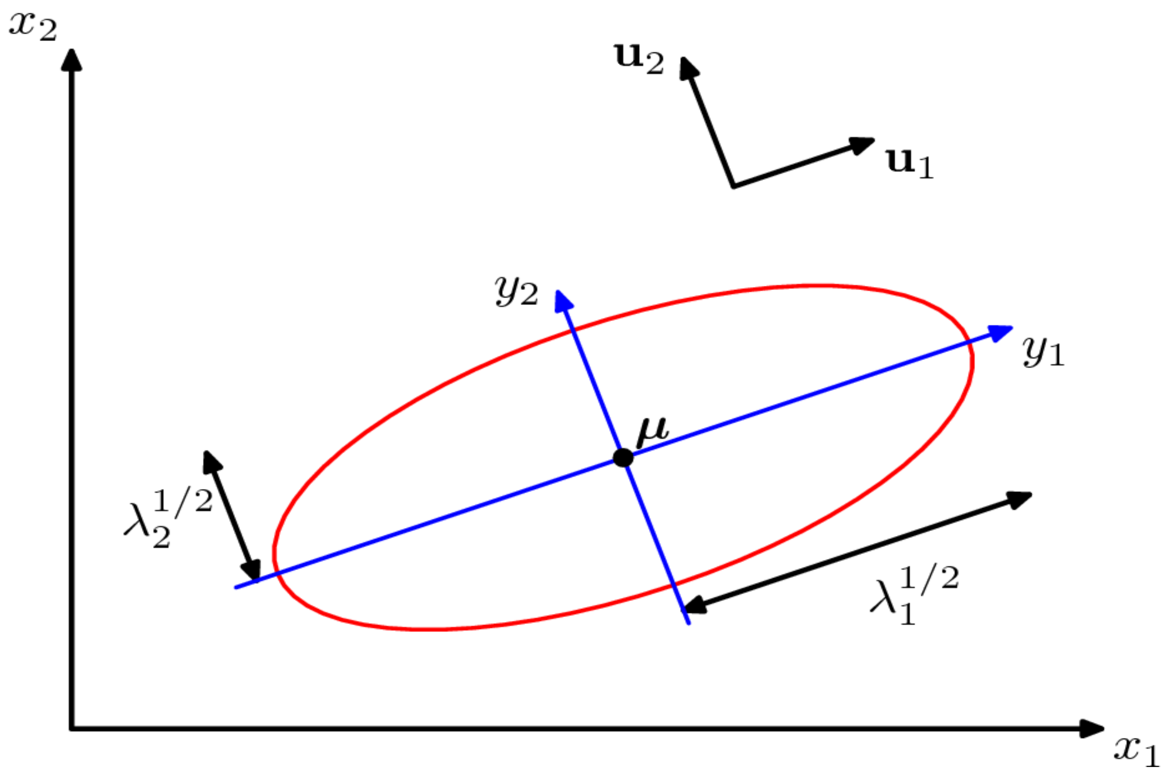
$$\text{Normal}(x \mid \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{ -\frac{1}{2\sigma^2}(x-\mu)^2 \right\}$$



- Parameters to be estimated are the mean (μ) and variance (σ)

Multivariate Gaussian (1)

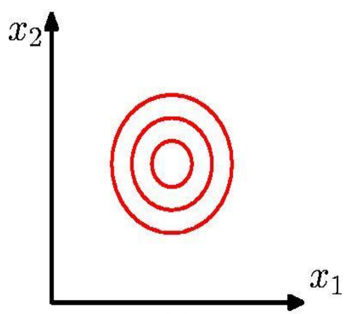
$$\text{Normal}(\mathbf{x} \mid \mu, \Sigma) = \frac{1}{(2\pi)^2} \frac{1}{\det(\Sigma)^{1/2}} \exp\left\{ -\frac{1}{2} (\mathbf{x} - \mu)^T \Sigma (\mathbf{x} - \mu) \right\}$$



- In multivariate case we have covariance matrix instead of variance

Multivariate Gaussian (2)

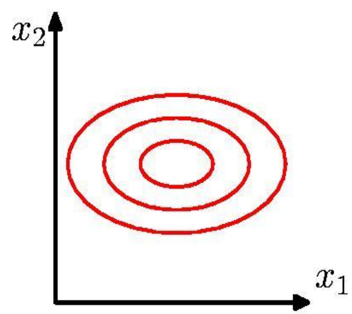
Single



(c)

$$\Sigma = \begin{pmatrix} \sigma^2 & 0 \\ 0 & \sigma^2 \end{pmatrix}$$

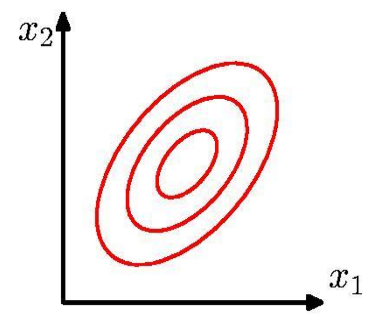
Diagonal



(b)

$$\Sigma = \begin{pmatrix} {\sigma_1}^2 & 0 \\ 0 & {\sigma_2}^2 \end{pmatrix}$$

Full covariance



(a)

$$\Sigma = \begin{pmatrix} {\sigma_{11}}^2 & {\sigma_{12}}^2 \\ {\sigma_{12}}^2 & {\sigma_{22}}^2 \end{pmatrix}$$

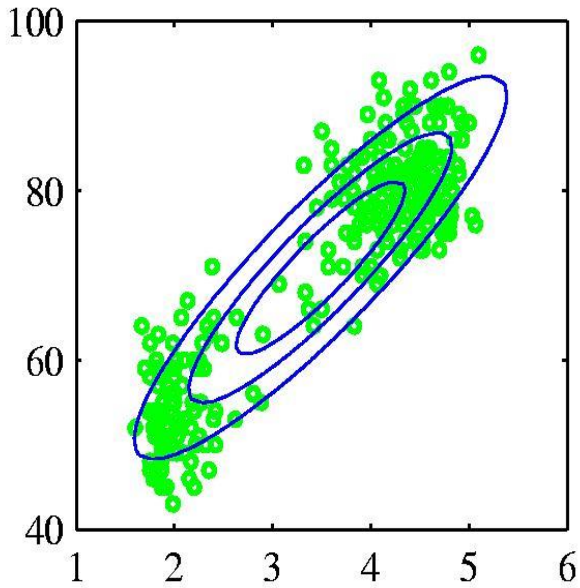
Complete data log likelihood:

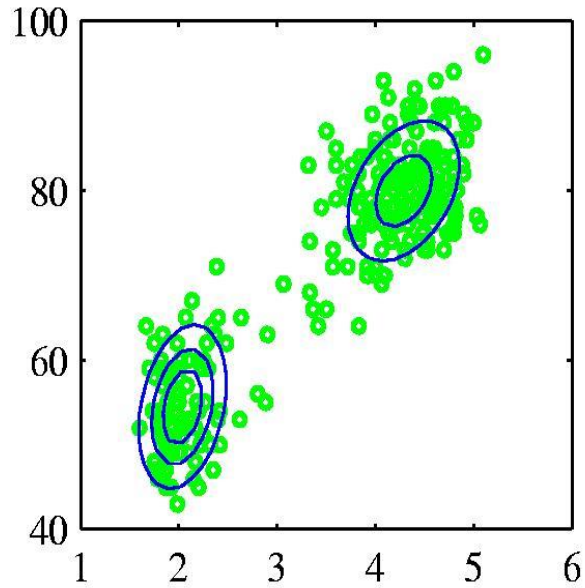
$$\ln p(X) = \ln \prod_{n=1}^{N} \mathrm{Normal}(\mathbf{x}_n \mid \mu, \Sigma)$$

Maximum Likelihood (ML) parameter estimation

- Maximize the log likelihood formulation
- Setting the gradient of the complete data log likelihood to zero we can find the closed form solution.
  - Which in the case of mean, is the sample average.

When one Gaussian is not enough

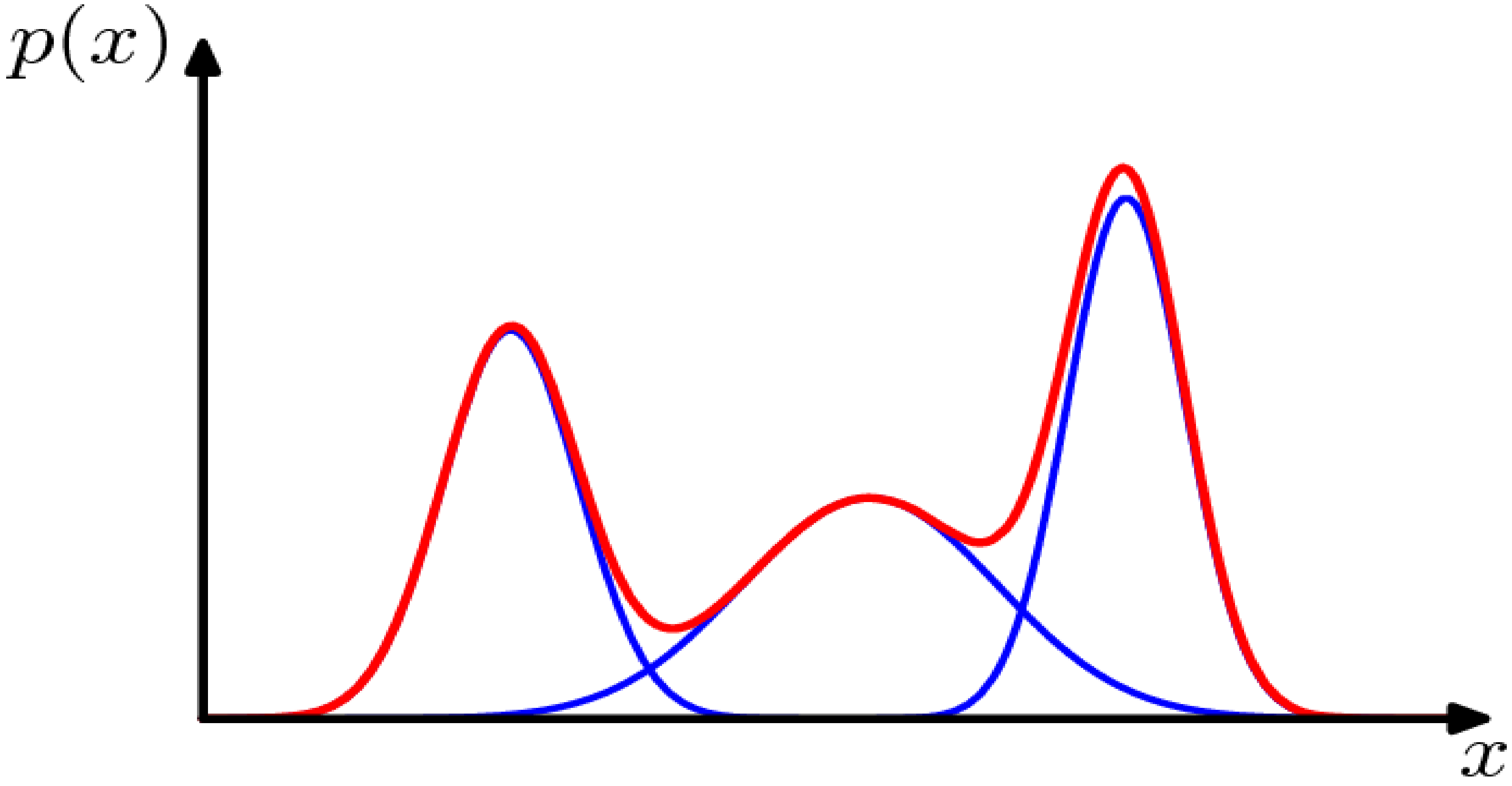




- Real world datasets are rarely unimodal!

Mixtures of Gaussians

$$p(\mathbf{x}) = \sum_{k=1}^{M} \pi_k \mathrm{Normal}(\mathbf{x} \mid \mu_k, \Sigma_k)$$



Mixtures of Gaussians (2)

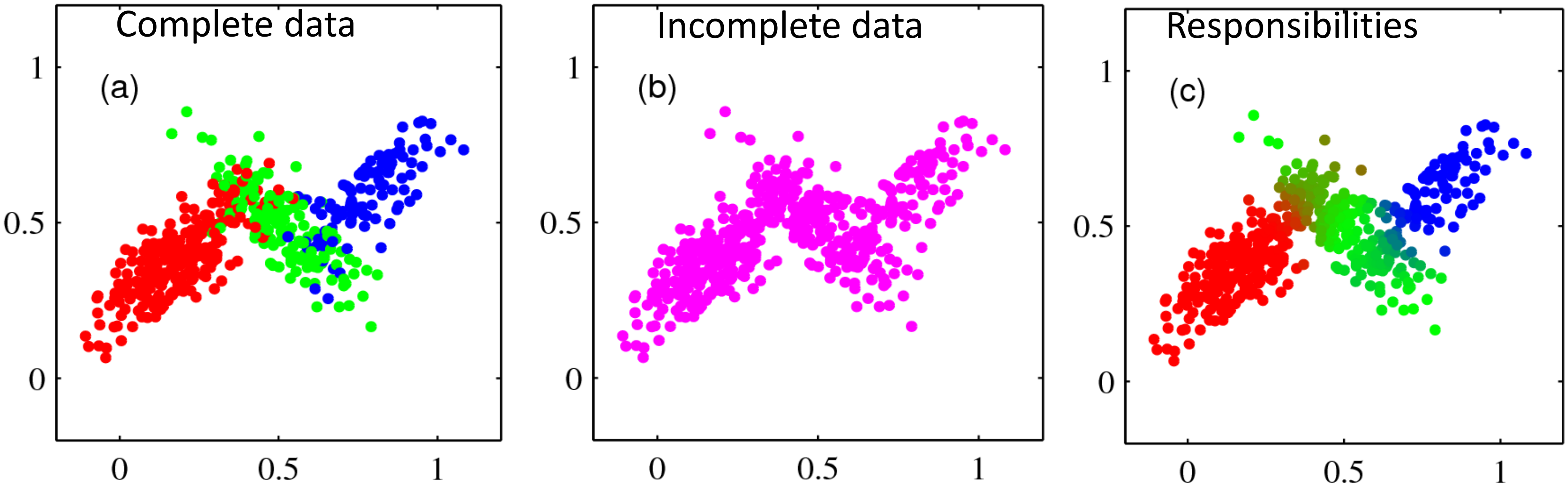
- In addition to mean and covariance parameters (now *M* times), we have mixing coefficients π_k.

Following properties hold for the mixing coefficients:

$$\sum_{k=1}^{M} \pi_k = 1 \qquad 0 \leq \pi_k \leq 1$$

It can be seen as the prior probability of the component *k*

Responsibilities (1)



- Component labels (red, green and blue) cannot be observed.
- We have to calculate approximations (responsibilities).

Responsibilities (2)

- Responsibility describes, how probably observation vector $\boldsymbol{x}$ is from component k.
- In clustering, responsibilities take values 0 and 1, and thus, it defines the hard partitioning.

Responsibilities (3)

■We can express the marginal density *p*(*x*) as:

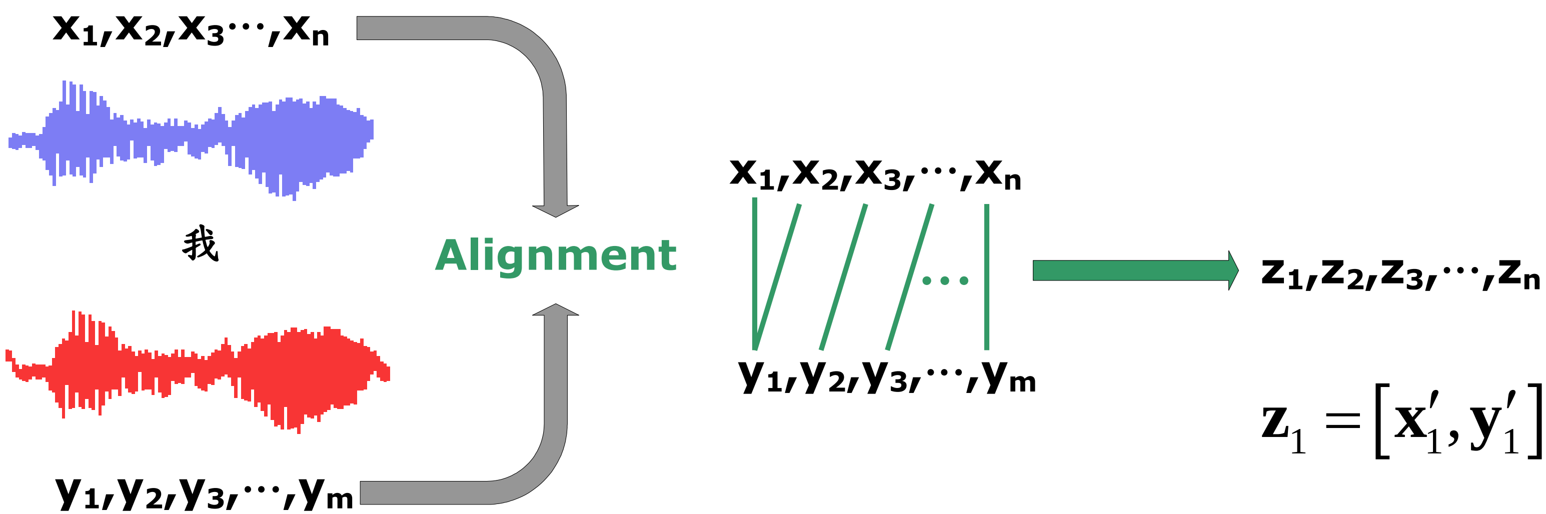
$$p(\mathbf{x}) = \sum_{k=1}^{M} p(k)\, p(\mathbf{x} \mid k)$$

■From this, we can find the responsibility of the kth component of *x* using Bayesian theorem:

$$\begin{aligned}\gamma_k(\mathbf{x}) &= p(k \mid \mathbf{x}) \\ &= \frac{p(\mathbf{x})\, p(\mathbf{x} \mid k)}{\sum_l p(l)\, p(\mathbf{x} \mid l)} \\ &= \frac{\pi_k \,\mathrm{Normal}(\mathbf{x} \mid \mu_k, \Sigma_k)}{\sum_l \pi_l \,\mathrm{Normal}(\mathbf{x} \mid \mu_l, \Sigma_l)}\end{aligned}$$

GMM-based Method

- Vector sequence of source speech $\mathbf{X} = \{\mathbf{x}_1, \mathbf{x}_2, \cdots, \mathbf{x}_n\}$
- Vector sequence of target speech $\mathbf{Y} = \{\mathbf{y}_1, \mathbf{y}_2, \cdots, \mathbf{y}_m\}$



$$\mathbf{z}_1 = \left[\mathbf{x}_1', \mathbf{y}_1'\right]'$$

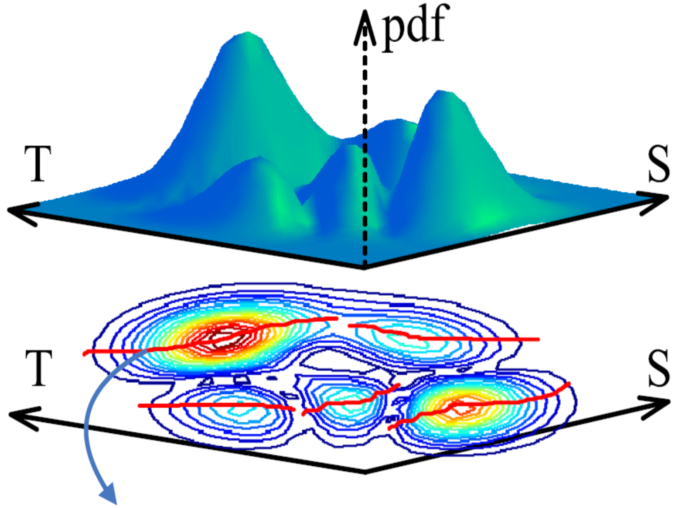
- Vector sequence of aligned source-target speech $\mathbf{Z} = \{\mathbf{z}_1, \mathbf{z}_2, \cdots, \mathbf{z}_n\}$

GMM-based Method

- Conversion function for a mixture

$$p(\mathbf{y}|\mathbf{x}) = \frac{1}{(2\pi)^{d/2}\det^{1/2}\left(\mathbf{\Sigma}^{\mathbf{YY}} - \mathbf{\Sigma}^{\mathbf{YX}}\left(\mathbf{\Sigma}^{\mathbf{XX}}\right)^{-1}\mathbf{\Sigma}^{\mathbf{XY}}\right)}\exp\left(-\frac{1}{2}\mathbf{U}\right)$$

$$\mathbf{U} = \left(\mathbf{y} - \left(\boldsymbol{\mu}^{\mathbf{Y}} + \mathbf{\Sigma}^{\mathbf{YX}}\left(\mathbf{\Sigma}^{\mathbf{XX}}\right)^{-1}\left(\mathbf{x} - \boldsymbol{\mu}^{\mathbf{X}}\right)\right)\right)' \left[\mathbf{\Sigma}^{\mathbf{YY}} - \mathbf{\Sigma}^{\mathbf{YX}}\left(\mathbf{\Sigma}^{\mathbf{XX}}\right)^{-1}\mathbf{\Sigma}^{\mathbf{XY}}\right]^{-1}\left(\mathbf{y} - \left(\boldsymbol{\mu}^{\mathbf{Y}} + \mathbf{\Sigma}^{\mathbf{YX}}\left(\mathbf{\Sigma}^{\mathbf{XX}}\right)^{-1}\left(\mathbf{x} - \boldsymbol{\mu}^{\mathbf{X}}\right)\right)\right)$$



- GMM-based conversion function

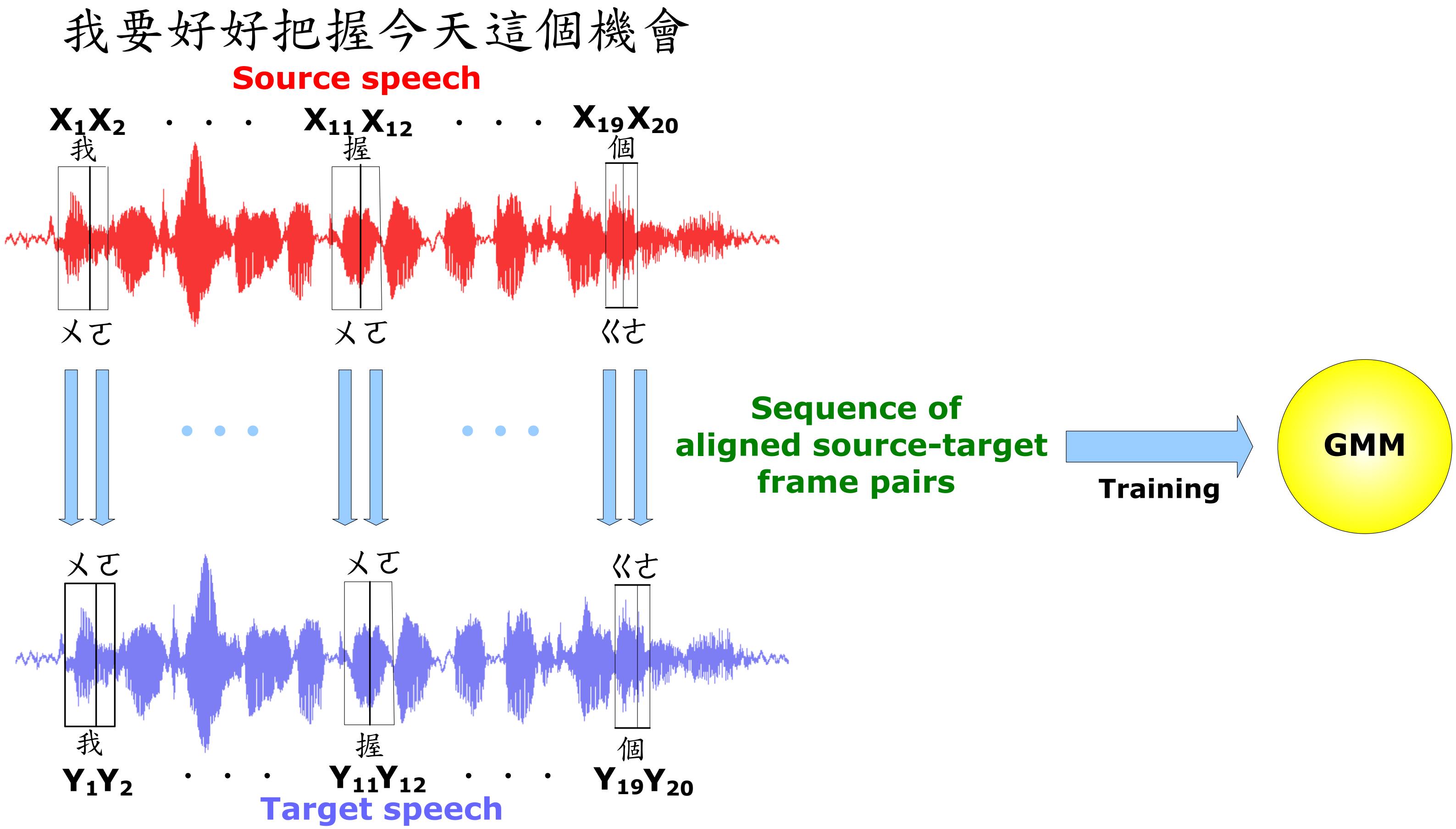
$$\tilde{\mathbf{y}}_t = F(\mathbf{x}_t) = E[\mathbf{y}_t|\mathbf{x}_t] = \sum_{m=1}^{M} p(m|\mathbf{x}_t)\left[\boldsymbol{\mu}_m^{\mathbf{Y}} + \mathbf{\Sigma}_m^{\mathbf{YX}}\left(\mathbf{\Sigma}_m^{\mathbf{XX}}\right)^{-1}\left(\mathbf{x}_t - \boldsymbol{\mu}_m^{\mathbf{X}}\right)\right]$$

– Posterior probability

$$p(m|\mathbf{x}_t) = \frac{w_m N\left(\mathbf{x}_t;\boldsymbol{\mu}_m^{\mathbf{X}},\mathbf{\Sigma}_m^{\mathbf{XX}}\right)}{\sum_{k=1}^{M} w_k N\left(\mathbf{x}_t;\boldsymbol{\mu}_k^{\mathbf{X}},\mathbf{\Sigma}_k^{\mathbf{XX}}\right)}$$

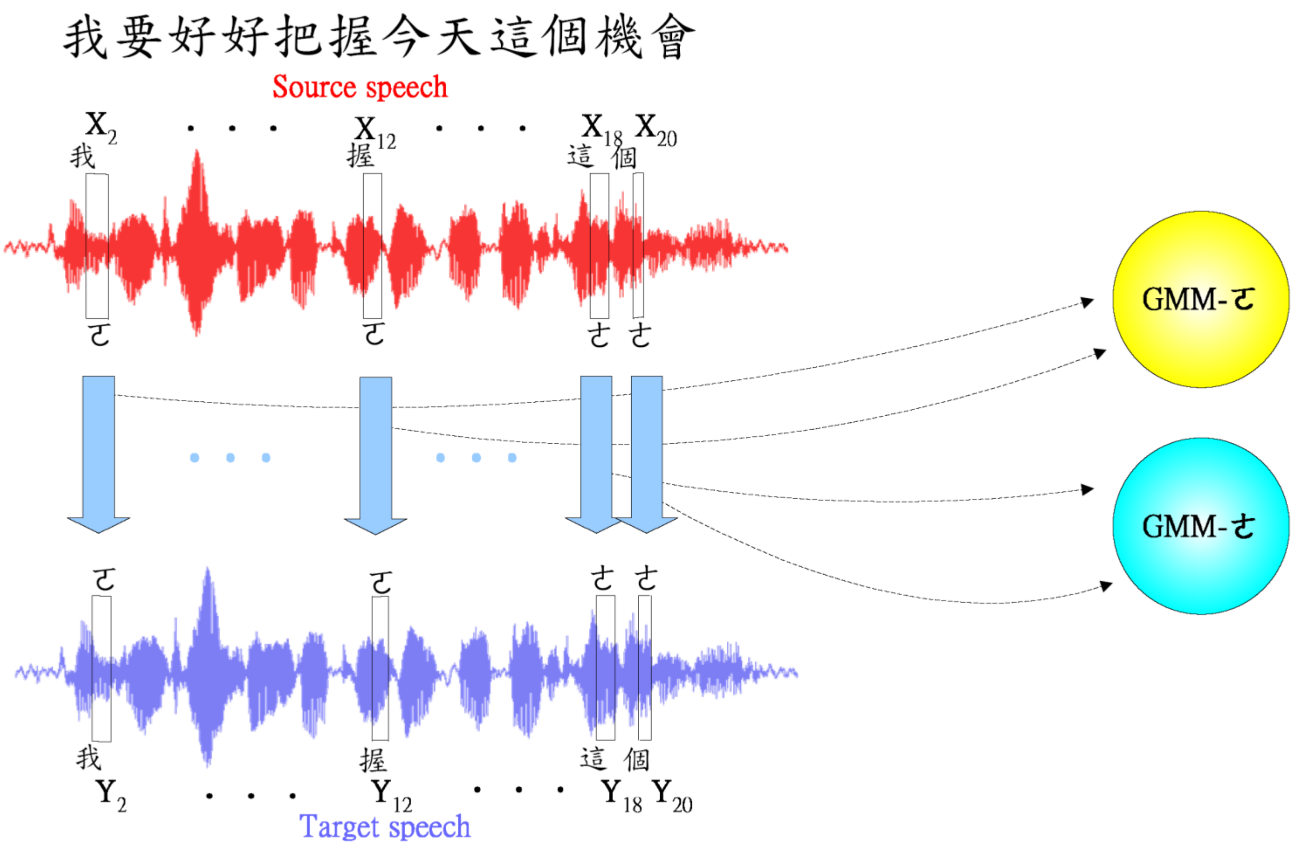
GMM-based Method

- One conversion function for **all sub-syllable**



GMM-based Method

- One conversion function for **each sub-syllable**
  - 38 context independent final
  - 112 right context dependent initial



GMM-based Spectral Transformation

1) Align N spectral feature vectors in time. (discrete cepstral coeffs)

$$\text{source}: X = \{x_1, \ldots, x_N\}, \quad \text{target}: Y = \{y_1, \ldots, y_N\}, \quad \text{joint}: Z = (X, Y)$$

2) Represent PDF of vectors as mixture of Q multivariate Gaussians

$$p(z) = \sum_{q=1}^{Q} \alpha_q N(z; \mu_q, \Sigma_q), \quad \sum_{q=1}^{Q} \alpha_q = 1, \quad \alpha_q \geq 0$$

Learn $\{\alpha_q, \mu_q, \Sigma_q, q = 1:Q\}$ from Expectation Maximization (EM) on Z

3) Transform source vectors using weighted mixture of Maximum Likelihood (ML) estimator for each component.

$$\hat{y}_n(x_n) = \sum_{q=1}^{Q} w_q^x(x_n)\left[\mu_q^y - \Sigma_q^{yx}\left(\Sigma_q^{xx}\right)^{-1}(x_n - \mu_q^x)\right]$$

$w_q^x(x_n)$: probability source frame belongs to acoustic class described by component q (calculated in Decoding)

GMM-Transformation Steps

1) Source frame $x_n \rightarrow$ want to estimate target vector: $\hat{y}_n$

2) Classify $x_n \rightarrow$ calculate $w_q^x(x_n)$

$w_q^x(x_n)$: probability source frame belongs to acoustic class described by component *q* (Decoding step)

3) Apply transformation function:

$$\hat{y}_n(x_n) = \underbrace{\sum_{q=1}^{Q} w_q^x(x_n)}_{\text{weighted sum}} \underbrace{\left[\mu_q^y - \Sigma_q^{yx}\left(\Sigma_q^{xx}\right)^{-1}(x_n - \mu_q^x)\right]}_{\text{ML estimator for class}}$$

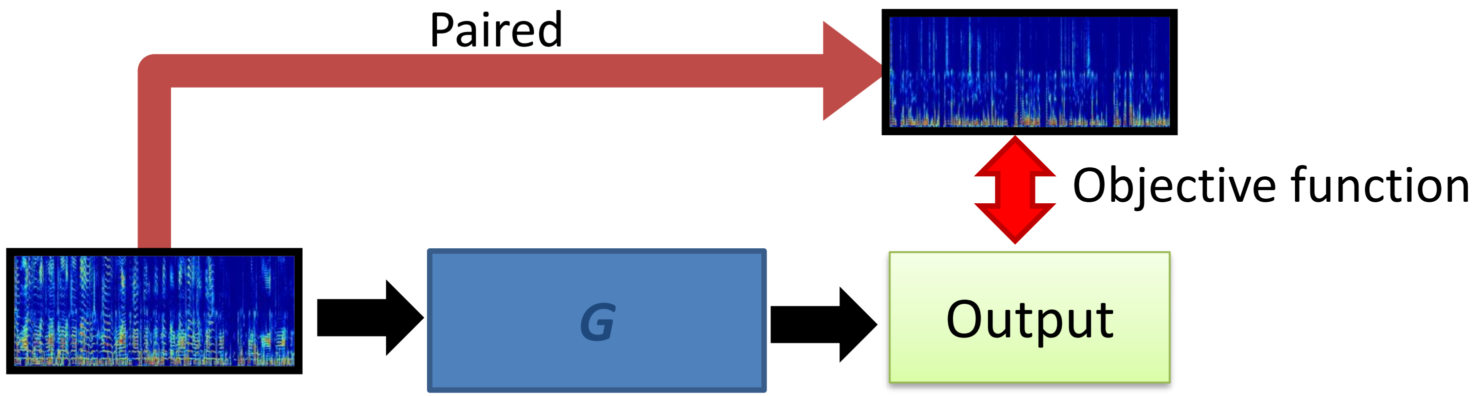
Conversion Examples

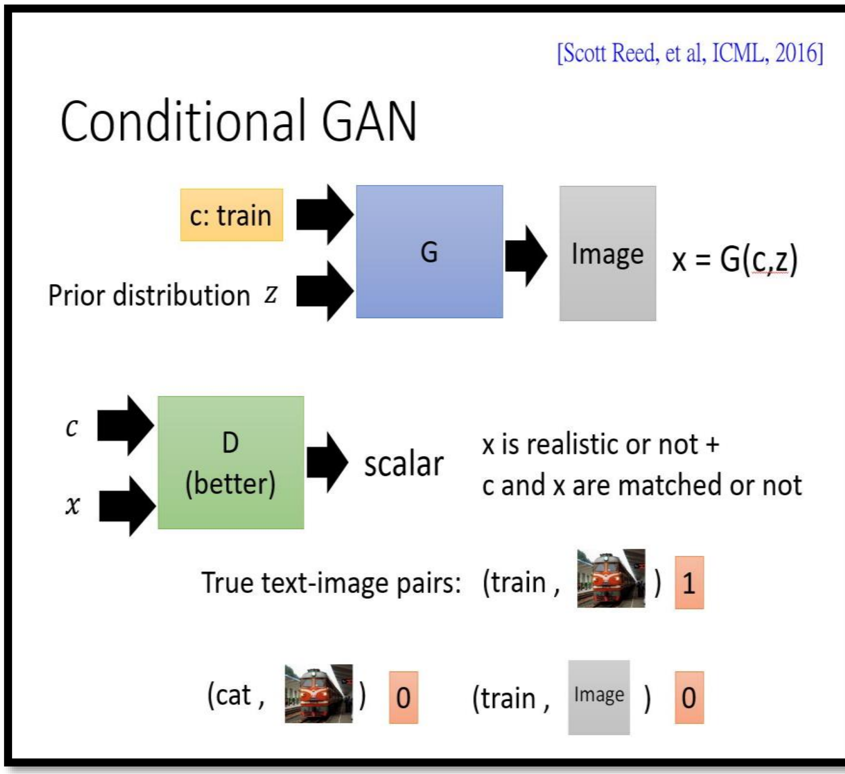
	Source	Target	GMM	DFWA	DFWE
slt → clb (FF)					
bdl → clb (MF)					

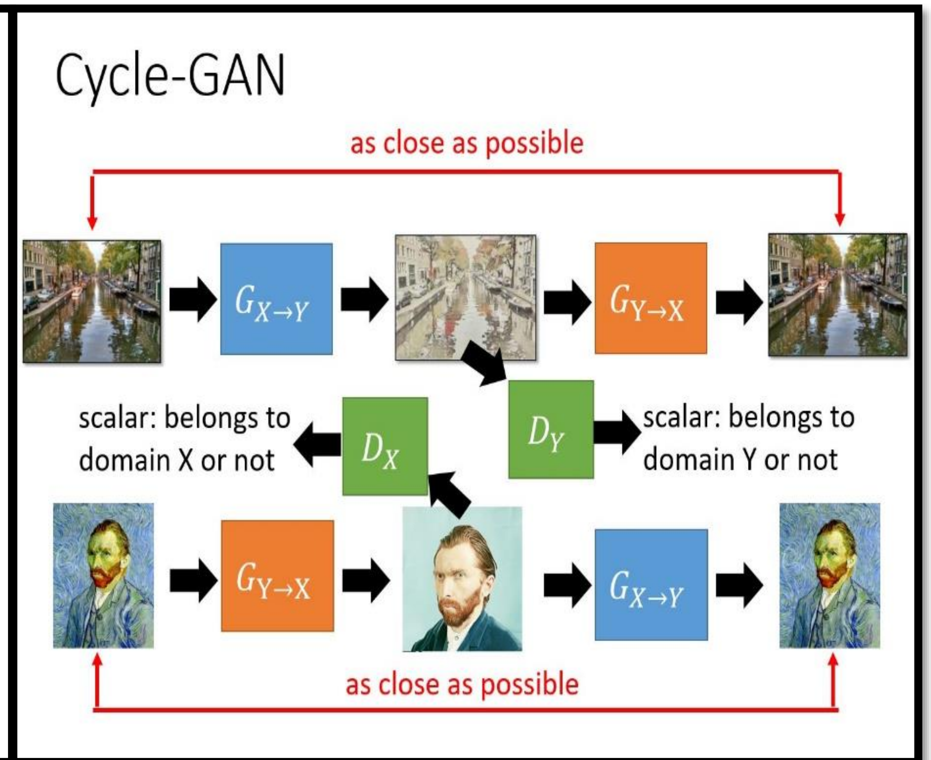
Target analysis-synthesis with converted spectral envelopes

- GMM-based suffer "loss of presence"

Speech Signal Generation (Regression Task)

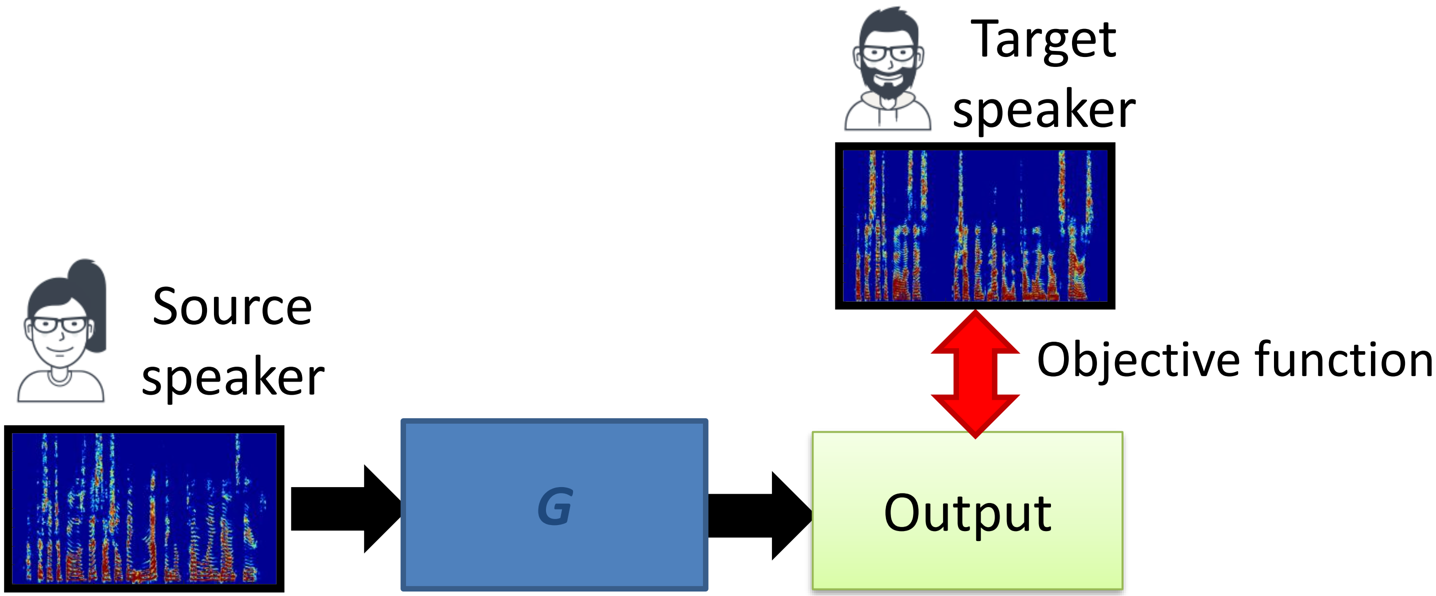






Voice Conversion

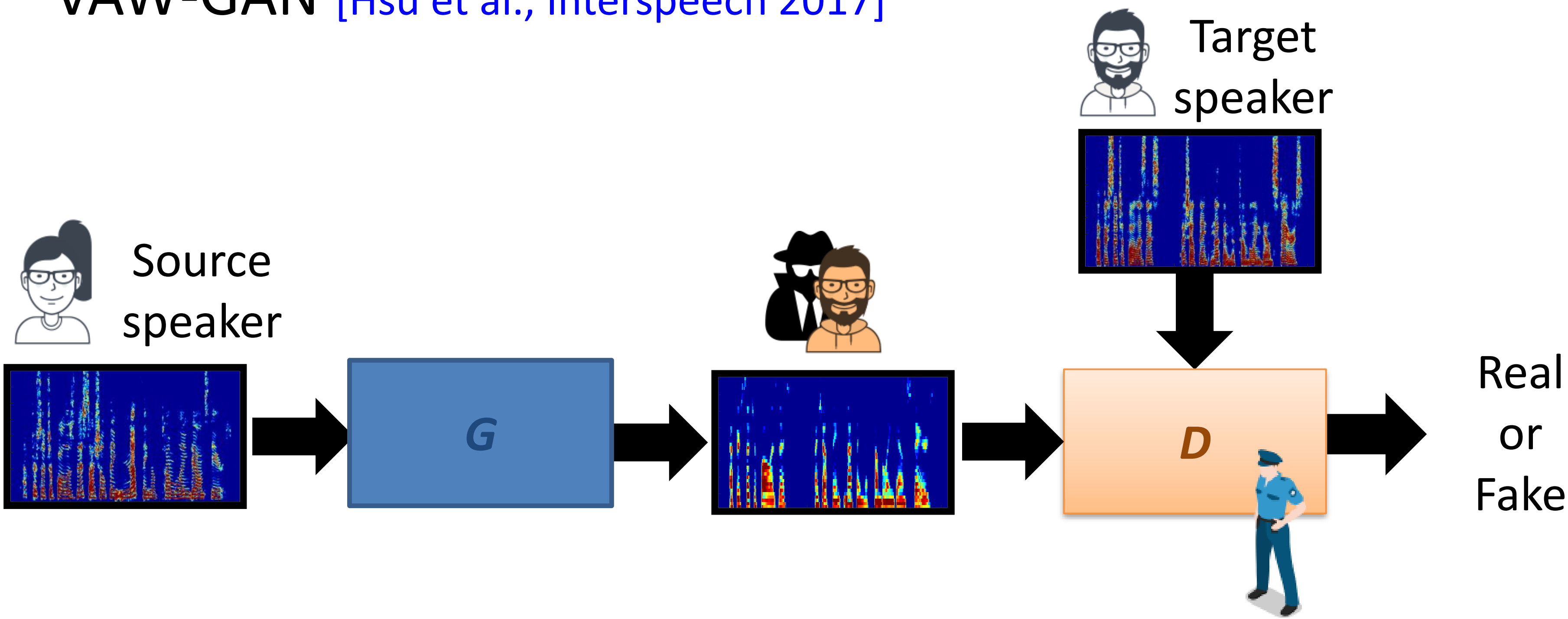
- Convert (transform) speech from source to target



➢ Conventional VC approaches include Gaussian mixture model (GMM) [Toda et al., TASLP 2007], non-negative matrix factorization (NMF) [Wu et al., TASLP 2014; Fu et al., TBME 2017], locally linear embedding (LLE) [Wu et al., Interspeech 2016], restricted Boltzmann machine (RBM) [Chen et al., TASLP 2014], feed forward NN [Desai et al., TASLP 2010], recurrent NN (RNN) [Nakashika et al., Interspeech 2014].

Voice Conversion

- VAW-GAN [Hsu et al., Interspeech 2017]



➢ Conventional MMSE approaches often encounter the "over-smoothing" issue.
➢ GAN is used a new objective function to estimate ***G***.
➢ The goal is to increase the naturalness, clarity, similarity of converted speech.

$$\boxed{V(G,D) = V_{GAN}(G,D) + \lambda V_{VAE}(\boldsymbol{x}|\boldsymbol{y})}$$

Voice Conversion (VAW-GAN)

- Objective and subjective evaluations

Fig. 14: The spectral envelopes.

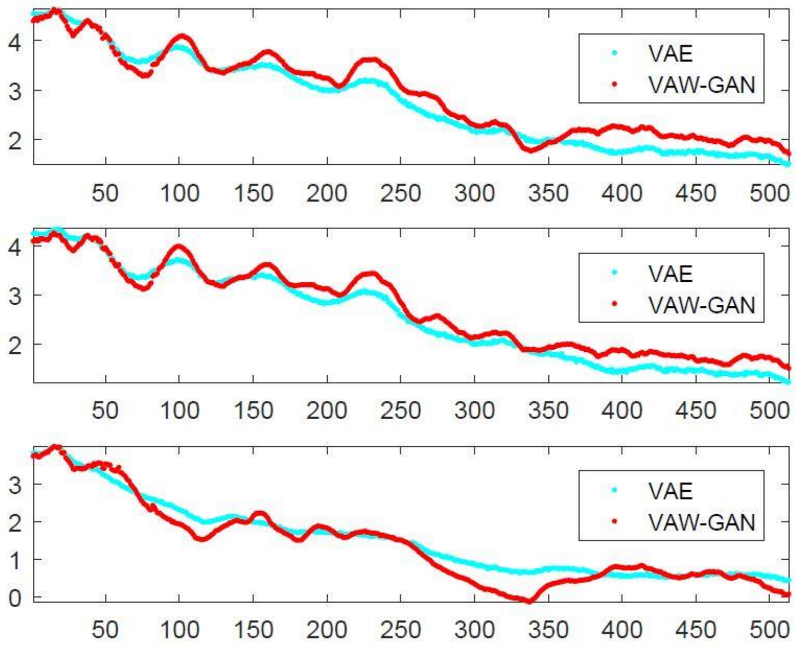
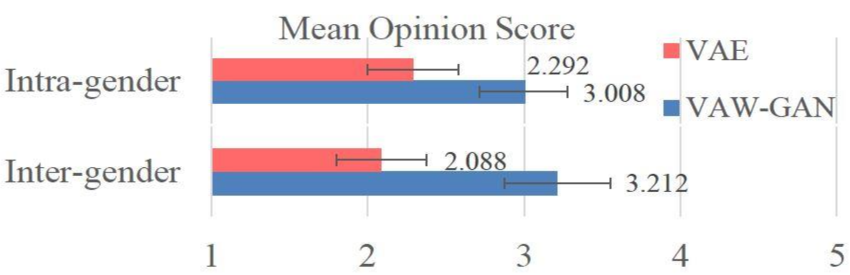


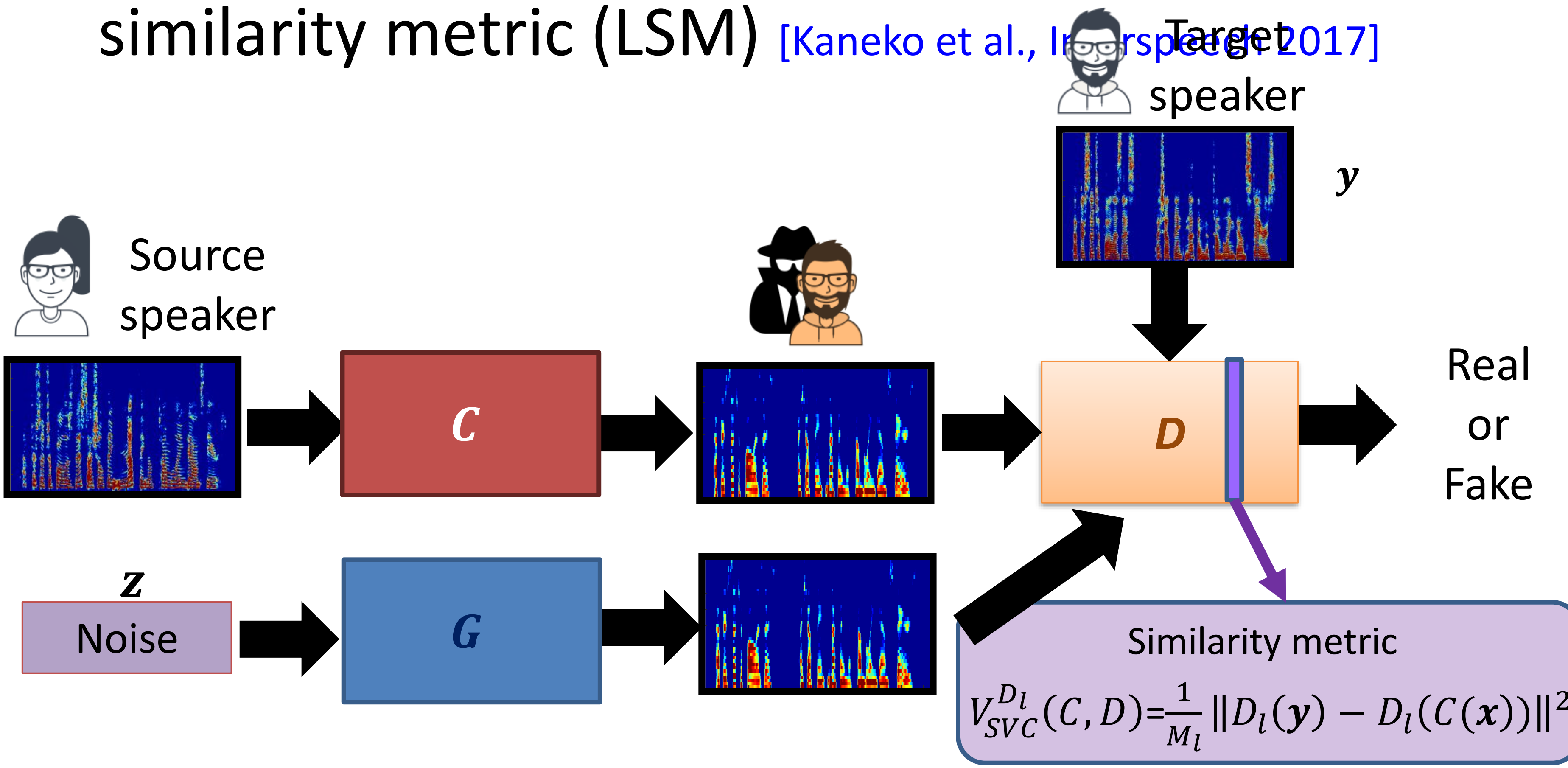
Fig. 15: MOS on naturalness.



VAW-GAN outperforms VAE in terms of objective and subjective evaluations with generating more structured speech.

Voice Conversion

- Sequence-to-sequence VC with learned similarity metric (LSM) [Kaneko et al., Interspeech 2017]

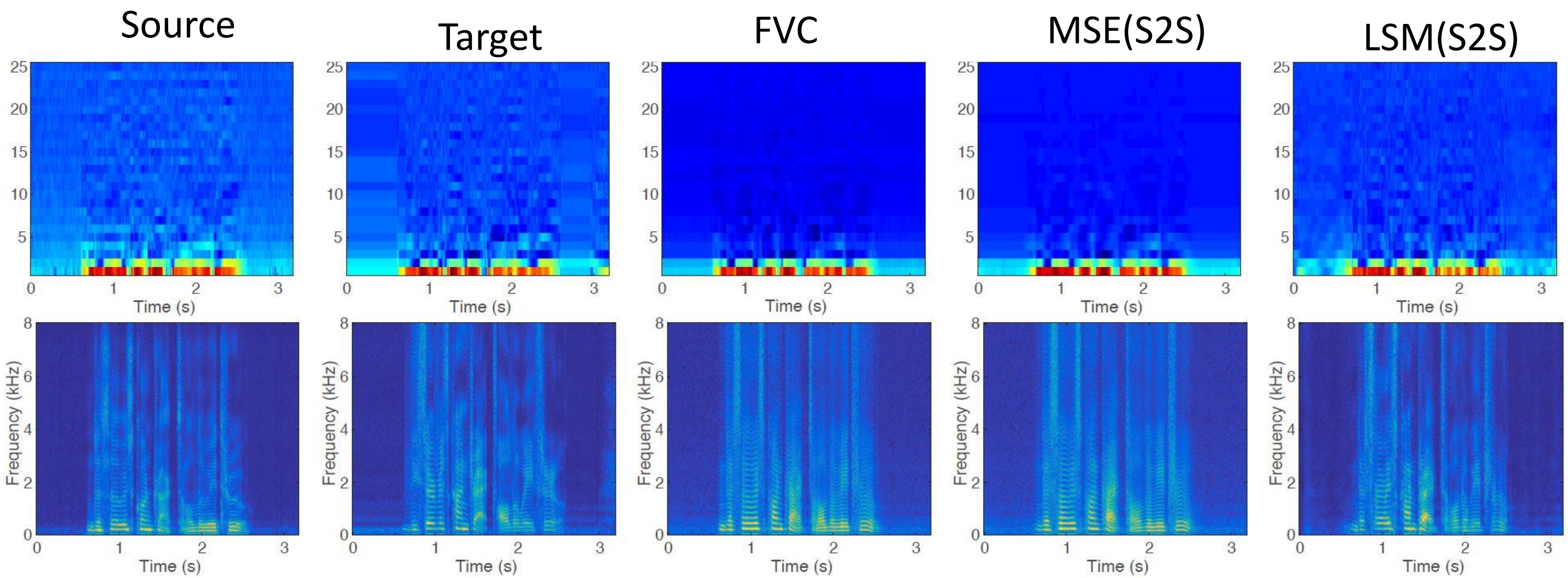


$$\boxed{V(C,G,D)=V_{SVC}^{D_l}(C,D)+V_{GAN}(C,G,D)}$$

Voice Conversion (LSM)

- Spectrogram analysis

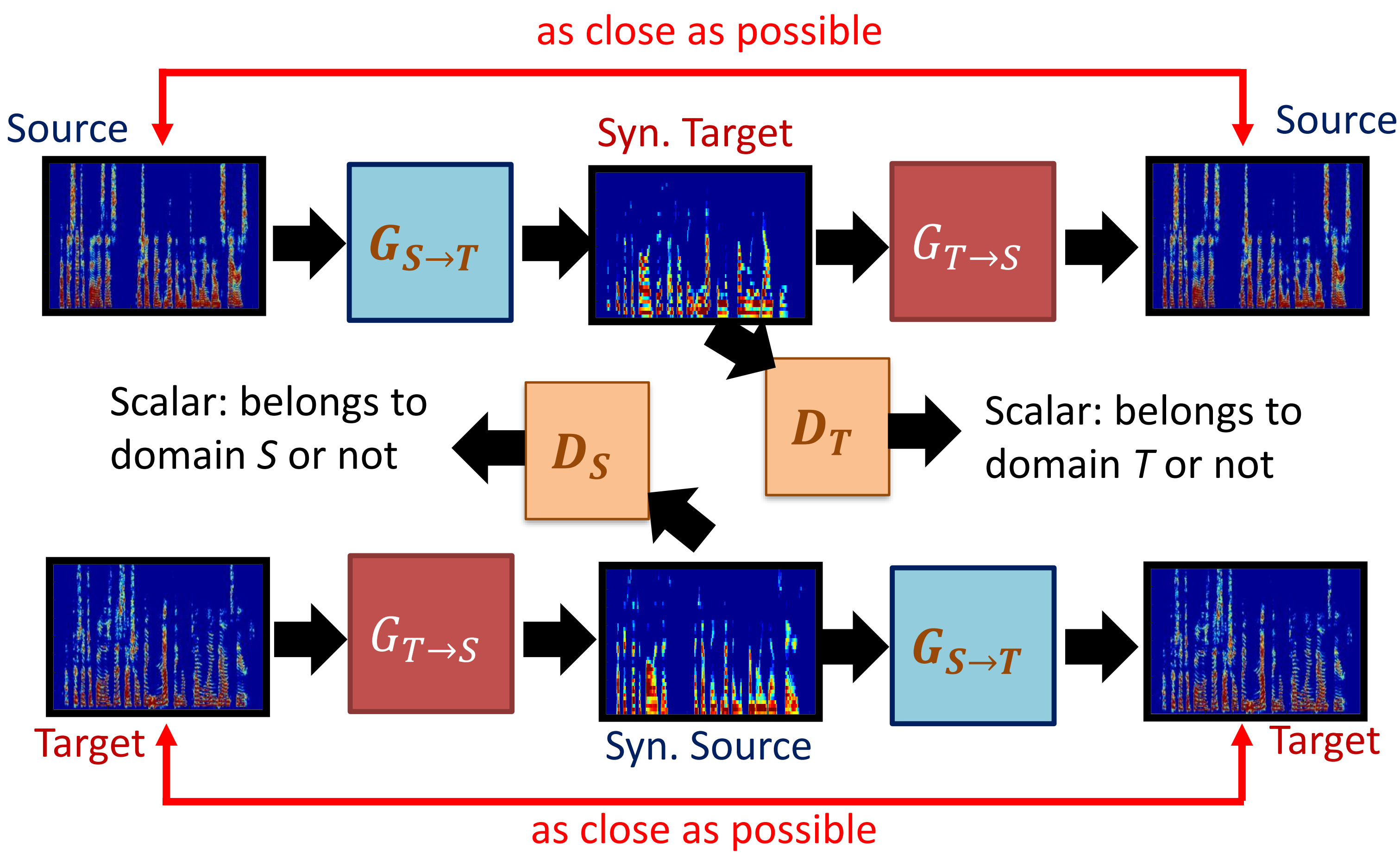
Fig. 16: Comparison of MCCs (upper) and STFT spectrograms (lower).



The spectral textures of LSM are more similar to the target ones.

Voice Conversion

- CycleGAN-VC [Kaneko et al., arXiv 2017]

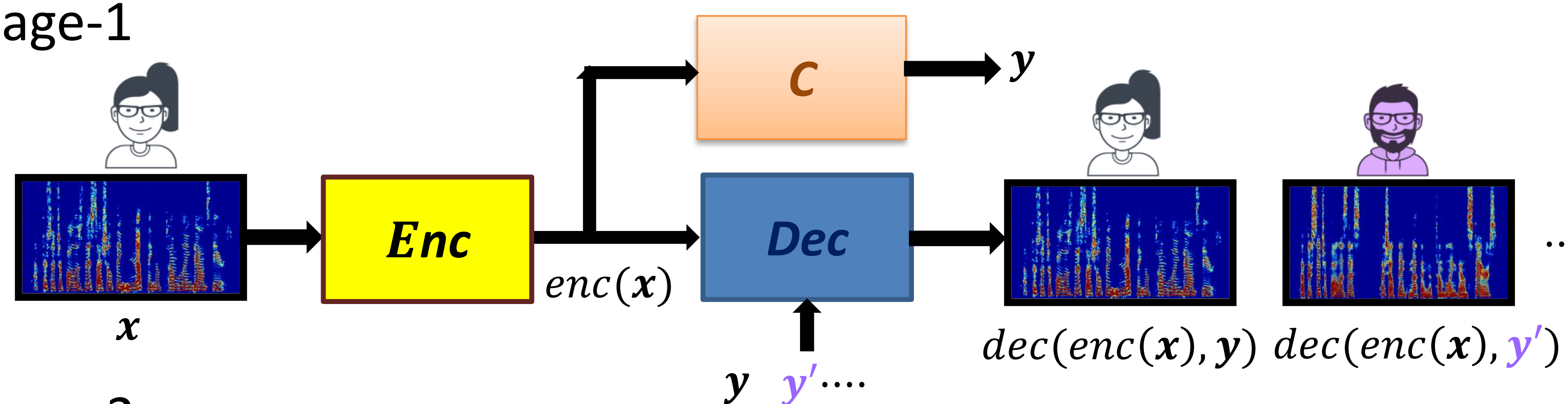


$$
\begin{aligned}
V_{Full} = &\ V_{GAN}(G_{X\to Y}, D_Y) + V_{GAN}(G_{X\to Y}, D_Y) \\
&+ \lambda\, V_{Cyc}(G_{X\to Y}, G_{Y\to X})
\end{aligned}
$$

Voice Conversion

- Multi-target VC [Chou et al., arxiv 2018]

➢ Stage-1



➢ Stage-2