# Statistical Methods for Speech Processing In Low Resource Environments

Hadas Ben Esti

# Statistical Methods for Speech Processing In Low Resource Environments

Research Thesis

As Partial Fulfillment of the Requirements for
the Degree Doctor of Philosophy

Hadas Ben Esti

Submitted to the Senate of the Technion—Israel Institute of Technology

Shevat 5776 Haifa January 2016

The Research Thesis Was Done Under The Supervision of Professor David Malah and Professor Koby Crammer in the Department of Electrical Engineering.

# List Of Publications

Some results in this thesis have been published as articles by the author and research collaborators in conferences and journals during the course of the author's doctoral research period, the most up-to-date versions of which being:

- H. Benisty and D. Malah, "Voice Conversion Using GMM with Enhanced Global Variance." *in Proc. INTERSPEECH*, 2011, pp 669-672.

- H. Benisty, D. Malah, and K. Crammer, "Modular global variance enhancement for voice conversion systems." *in Proc. EUSIPCO*, 2012, pp. 370-374.

- H. Benisty, D. Malah, and K. Crammer, "Non-parallel voice conversion using joint optimization of alignment by temporal context and spectral distortion." *in Proc. ICASSP*, 2014, pp. 7909-7913.

- H. Benisty, D. Malah, and K. Crammer, "Sequential voice conversion using grid-based approximation." *in Proc. IEEEI*, 2014.

- H. Benisty, D. Malah, and K. Crammer, "Grid-Based Approximation For Voice Conversion In Low Resource Environments", *EURASIP Journal on Audio, Speech, and Music Processing*, Jan. 2016.

- H. Benisty, K. Crammer, D. Malah, and I. Kats, "Discriminative Keyword Spotting For Low Resource Applications", *to be submitted after submission of a patent*, 2015.

# Acknowledgement

I would like to express my deepest gratitude to my supervisors: Prof. David Malah and Prof. Koby Crammer for their devoted supervision and guidance. I feel privileged to have learnt from them, both professionally and personally. I greatly appreciate their guidance, fruitful advices, encouragement to perfection, and mostly, for always pushing me forward, especially in times when I was about to give up.

Many thanks to the devoted Signal and Image Processing Lab (SIPL) staff: Nimrod Peleg, Yair Moshe, Ziva Avni and Avi Rozen who create a pleasant environment and a family-like atmosphere.

I would like to thank my friends from the Technion: Ronen, Miri, Itamar and Nurit. Thanks for all those professional and personal talks, and for making my time in the Technion fun.

I would like to thank my dearest and sweet husband, Eyal, who believed in me back in the beginning of the first degree and kept believing and supporting me till graduation of PhD. Your love is my strength. Special thanks for my beloved family, my parents: Aythan and Bruria, my bothers: Tomer and Allon, for your endless help, support and love. Thanks to my best and dearest friends: Noga, Tali, Ela, Michal and Lizu, for all your love and support. And finally, to my beloved and sweet children: Talia, Daniel and Reut, who daily grant me with endless love, happiness and joy.

The generous financial help of the Technion is gratefully acknowledged.
This research was funded in part by the Ministry of Israeli Economics grant 50360 and in part by an Israeli Science Foundation grant ISF- 1567/10.

# Contents

# List of Figures

# List of Tables

# Abstract

Many speech processing systems are based on statistical modeling of speech signals, thus requiring relatively large-scale data-sets for training. As technology advances, computational effort and memory footprint are less of a problem for such systems, while the amount of data available for training is still challenging in many limited-data applications such as under-documented languages, speech of children, and mobile applications, where most users are not willing to invest much time and effort in recording themselves. For this setup we address two major speech processing tasks: a voice conversion task, in which a sentence said by a source speaker is converted to sound as if said by a target speaker, and a keyword spotting (KWS) task of detecting whether a given keyword was said or not, in a speech utterance.

Common voice conversion systems are based on a Gaussian Mixture Model (GMM), thus requiring at least several dozens of recorded sentences for training. The trained conversion function is linear, often producing muffled synthesized signals due to over-smoothing of the converted spectral envelope. We present a method for voice conversion for low data-resource applications, where the conversion process is expressed as a sequential estimation problem of tracking the target spectrum based on the observed source spectrum. To improve the quality of the converted synthesized signals, we also present methods for enhancing the global variance of the converted signal.

Most voice conversion systems require a parallel training set, in which the two speakers say the same text. In this work we also address the non-parallel setup, where no assumptions are made regarding the uttered text of the training set. In this setup, in addition to training a conversion function, the source-target correspondence also needs to be evaluated. We present here a generalized version of an existing method, by using temporal context vectors to improve the source-target matching process and prove that it converges.

Standard KWS methods require medium-large phonetically segmented sets for training, and therefore are not adequate for limited-data environments. In this work we propose a new KWS method, suitable for this setup, based on discriminative classifiers for words and sentences. We present a new histogram representation for words, obtained with respect to a pre-trained Gaussian Mixture Model (GMM). Sentences are represented by a fixed-length global feature vector, extracted from the response curve obtained by the word classifier. Dataset for training the GMM can be easily obtained since no annotation or labeling is required.

Non-keyword recordings can be easily obtained, as opposed to speech including the keyword, which needs to be specifically provided for each keyword, so a highly biased training set is a reasonable scenario. To avoid biased classifiers, we use bagging predictors for training both word and sentence classifiers. According to our experiments, the proposed KWS system performs better than an HMM benchmark system for small training sets, and is more robust to highly variable signals, such as speech of children, and to noisy conditions - specifically, babble and car noise in a wide range of SNR values.

# Notation

| | |
|---|---|
| $A_l(m)$ | amplitude of the $l$-th sinusoid at frame $m$ |
| $f_l(m)$ | frequency of the $l$-th sinusoid at frame $m$ |
| $f_0(m)$ | fundamental frequency - pitch at frame $m$ |
| $\mathcal{F}$ | a conversion function |
| $L(t)$ | number of sinusoids at time $t$ |
| $L_1$ | amount of bagging predictors for word classification |
| $L_2$ | amount of bagging predictors for sentence classification |
| $\mathcal{M}$ | GMM parameters |
| $p\left(\mathbf{x}_t \vert \mathbf{y}_t^k\right)$ | likelihood probability of the source spectrum given the target spectra |
| $p\left(\mathbf{y}_t \vert \mathbf{x}_{1:t}\right)$ | posterior probability of the target spectrum given the source spectra |
| $\theta_l^0(m)$ | phase of the $l$-th sinusoid at frame $m$ |
| $\mathbf{u}_t$ | an indicator vector for the Gaussian in $\mathcal{M}$, leading to the maximal poterior at time $t$ |
| $\mathbf{v}$ | a histogram representation of a word |
| $x$ | scalar |
| $\mathbf{x}$ | spectral feature vector related to a source speaker |
| $\mathbf{X}_{1:T}$ | a sequence of feature vectors related to the source speaker |
| $\mathbf{y}$ | spectral feature vector related to a target speaker |
| $\mathbf{Y}_{1:T}$ | a sequence of feature vectors related to the target speaker |
| $\tilde{}Z_{1:T}$ | a sequence of converted and enhanced feature vectors |
| $\mathbf{z}_t$ | a posterior vector related to the time frame $t$ |

# Abbreviations

| | |
|---|---|
| AR | Auto-Regressive |
| ASR | Automatic Speech Recognition |
| AUC | Area Under the ROC Curve |
| CGMM | Constrained GMM |
| DTW | Dynamic Time Warping |
| EM | Expectation Maximization |
| Fant | Filtering and Adding Noise Tool |
| FFT | Fast Fourier Transform |
| GMM | Gaussian Mixture Model |
| GB | Grid-Based |
| GV | Global Variance |
| HM | Harmonic Model |
| HMM | Hidden Markov Model |
| HNM | Harmonic Plus Noise Model |
| INCA | Iterative combination of a Nearest Neighbor search step and a Conversion step Alignment method |
| JGMM | JGMM |
| KWS | Keyword Spotting |
| LSD | Log Spectral Distortion |
| LSF | Line Spectral Frequencies |
| LVCSR | Large Vocabulary Continuous Speech Vocabulary |
| MFCC | Mel Frequency Cepstrum Coefficients |
| MLP | Multi-Layer Perceptron |
| MUSHRA | Multi Stimulus test with Hidden Reference and Anchor |
| ND | Normalized Distortion |
| NGV | Normalized Global Variance |
| ROC | receiver Operating Characteristics |
| SM | Sinusoidal Model |
| SVM | Support Vector Machine |
| TC-INCA | Temporal Context INCA |
| TD-PSOLA | Time-Domain Pitch-Synchronous Overlap and Add |
| VQ | Vector Quantization |

# Chapter 1

# Introduction

Modern speech processing systems often require large scale resources in terms of training set size, computational effort and memory footprint, which present an engineering challenge when implemented in low resource environments. As technology evolves, the computational and memory abilities become less limiting but the training set size is still challenging. In mobile applications for example, typical users are willing to record themselves saying just few sentences, therefore providing a very small data set for training. Also, in cases of under-documented languages, large-scale data sets with or even without phonetic labelling are not available for training. In this work we concentrate on two speech processing tasks: a voice conversion task and a keyword spotting task.

Common voice conversion systems are based on modeling of the source and target spectra using a Gaussian Mixture Model (GMM), [2, 3], which requires several dozens of recorded sentences. In this method the trained conversion function is a linear function that over-smoothes the converted spectral envelopes, which results in muffled output signals, [4,5]. Recently, a different approach aiming to capture the temporal evolution of the spectral envelope was presented [6], where the Global Variance (GV) of the spectral features was considered in the trained statistical model. In this work we propose two methods for improving the quality of converted signals through enhancement of the global variance of the spectral features: 1) a GMM-based conversion method with a GV constraint (CGMM) 2) a modular enhancement block, independent of the conversion process and applied on converted signals as a post-processing block. Our subjective evaluations show that these enhancement methods significantly improve the quality of the synthesized outputs and also improve their similarity to the target signal (individuality), compared to the GMM-based conversion method.

Most voice conversion methods, including the GMM-based conversion methods mentioned above, are trained using parallel sentences where the source and target speakers are recorded saying the same text. This requirement of having dozens of parallel sentences for training is rather limiting in general, and not feasible in case of mobile users wanting their own voice to be the source/target speaker.

In this work we present a method for voice conversion for low resource environments, called a Grid-Based (GB) voice conversion, which can be successfully trained using just 5-10 sentences. The conversion process is expressed as a sequential estimation problem of tracking the target spectrum based on the observed source spectrum. The converted spectra are sequentially evaluated as a discrete sum of the target training vectors, used as grid-points. In an extreme setup of using just 10 sentences for training, objective evaluations show that the GB method leads to lower spectral distance and to higher variability than the GMM-based conversion method, at the same time.

To improve the perceived quality of the synthesized output signals, we applied our GV enhancement block to converted signals obtained by the GB method and by the GMM conversion. In our subjective evaluations comparing the output signals three systems: enhanced GB, enhanced GMM and CGMM, the Enhanced GB system was marked highest in terms of individuality, and comparable to the enhanced GMM system, in terms of quality (CGMM was rated as best).

We also address a non-parallel setup, where no assumptions are made regarding the text uttered by the source and target speakers. In such setup, in addition to training a conversion function, the source and target correspondence is evaluated using the training data alone, based on a statistical model, [7], or on a nearest neighbor search [1]. In the nearest neighbor approach called Iterative combination of a Nearest Neighbor search step and a Conversion step Alignment method (INCA), a source-target matching and a conversion function are iteratively evaluated. The matching process is applied using single feature vectors, so it is often that two a source vector and a target vector are matched, even though they do not relate to the same phonetic context, or even to the same phoneme, which degrades the performance of the conversion function since it is relies on these matched pairs, [1].

In our proposed approach, a generalized version of INCA called Temporal Context-INCA (TC-INCA), sequences of vectors are matched, and therefore their temporal context during the matching process is considered, instead of matching feature vectors one-by-one. Additionally, we show that the generalized iterative process (and therefore also its particular case - INCA), is in fact an alternating minimization procedure which minimizes a joint cost function including the matching and conversion functions. Our experiments results show that TC-INCA raises the accuracy of matching and, and a consequence, improves the quality and individuality of the synthesized output signals, compared to INCA.

Standard keyword spotting methods use Hidden Markov Models (HMMs) to statistically model subword units [8,9]. They require phonetically labelled recordings for training and therefore are not applicable in a limited-data setup. Other approaches avoid using phonetic labeling for training, by representing the searched keyword as a template signal and compare it against a similar representation of a given speech utterance. A posterior representation, with respect to various statistical models have been proposed for creating keyword templates, [8, 10–12]. The posterior representation of the template and test signals do not match in length, since the natural rate of speech varies with speakers and context. Therefore many of these methods use Dynamic Time Warping (DTW), which imposes a challenging computational load.

Many spotting approaches, including those mentioned above are generative methods, i.e., they aim to model the generation process of the speech signals including the keyword. Inference is made by measuring the correspondence of a given test utterance to the trained model. The main criticism against these approaches is that they are not trained directly to minimize detection errors. In recent years, some approaches based on discriminative classification have been proposed. These methods use machine learning techniques for training of optimal classifiers between speech signals including keywords and not including it. Keshet et al. proposed a new feature representation for speech utterances based on the estimated duration of phonemes and transition times [13]. This method is trained using phonetically segmented data at a medium size such as TIMIT, which consists of about 4 hours of recorded speech. Recently, two methods dealing with very small training sets, without phonetic labeling, have been proposed using features extracted from the time-frequency representation of speech signals [14, 15].

In this work we present a novel discriminative method for keyword spotting, which can be trained using small and unlabeled data sets, and therefore suitable for a low resource environment. Our method is based on two classifiers: an isolated word classifier trained using samples of the keyword and samples of non-keywords speech, and a sentence classifier trained using sentences including the keyword, and sentences not including it.

We propose a new fixed length representation for isolated words based on histograms, obtained with respect to a pre-trained GMM, which captures the structure of the spectral feature vectors. We use Expectation Maximization (EM) [16], an unsupervised method, for training the GMM. Therefore even in the case of an under-documented speech, a large amount of data can be easily obtained since no labelling is required for training. Moreover, the GMM is used for representing the entire spectral structure of the spoken language and therefore does not need to be retrained for detection of other keywords. Consequently, its training can be performed off-line on a distant server beforehand. Given a sequence of spectral feature vectors related to a word, we extract the posterior probabilities for each time frame with respect to the GMM. A histogram is obtained by counting the amount of times each Gaussian component in the mixture leads to the highest posterior. We train a binary classifier for words, using histograms related to utterances of the keyword and histograms related to utterances of other words.

We also propose a fixed length representation for sentences: given a sequence of spectral feature vectors extracted from a sentence, we apply a sliding window to produce sequences of histograms, as described above. Applying the word classifier on each histogram yields a response curve. Positive sentences, i.e., those including the keyword, mostly lead to a distinct and positive maximum value, which corresponds to the location of the keyword in the sentence, while negative sentences, i.e., not including it, lead to a random-like, negative or close to zero values. We obtain a fixed length representation for the given sentence by extracting a set of global features, such as maximal value, dynamic range, etc. A binary classifier for sentences is trained using these global feature vectors obtained from positive and negative sentences.

While negative examples are easily obtained, positive examples are much harder to acquire since, as mentioned above, mobile users are willing to record themselves saying a keyword just a limited number of times. In this setup, the amount of positive examples available for training is much smaller than the negative one, so the training process may result in a classifier that is biased towards negativity. We avoid this situation, and still exploit the diversity of the negative training set by using bootstrap aggregating, also referred to as bagging predictors [17]. According to this approach, a series of classifiers are trained, each using uniformly sampled subsets of the larger set and the smaller set; inference is made by applying all trained classifiers and taking the majority decision. We applied this concept for classification of both words and sentences.

To demonstrate the advantages of the proposed method we performed experiments on speech of both adults and children, in several challenging conditions considering: training set size and background noise. We followed a previously suggested experiment for keyword spotting on speech signals of adults and showed that our system outperformed the methods presented by [14, 15]. For clean speech signals of children, our method is significantly better than an HMM-keyword spotter, when trained using very few positive samples (5-10). When tested on noisy speech (car and babble), our method outperforms the benchmark HMM system in all the examined cases regardless of training set size, SNR value or noise type.

## 1.1 Thesis Structure

The rest of the thesis is organized as follows: in Ch. 2 we provide a short theoretical background dealing with speech modeling. Common methods used for voice conversion are presented in Ch. 3. In Ch. 4 we present our proposed methods for GV enhancement. A new method for voice conversion in low resource environments, called Grid-Based voice conversion is proposed in Ch. 5. In Ch. 6 we present our method for non-parallel training of a voice conversion system. Our novel keyword spotting approach is presented in Ch. 7. In Ch. 8 we conclude by summarizing the main contributions of this work and propose further research directions.

# Chapter 2

# Speech Modeling

## 2.1 Sinusoidal-Based Models

A speech signal can be modelled by an excitation signal passing through a time varying linear filter. The most familiar form of this representation models the speech signal as an auto-regressive (AR) signal [18], where the filter is assumed time-invariant at each analysis frame. The excitation signal is taken as a pulse train, in case of a voiced frame, or noise for an unvoiced frame. The Sinusoidal Model (SM) [19] is a sum of sinusoids representing both types of excitation signal. Assuming, again, time-invariance of the filter during a processing frame, the output also takes the form of a SM:

$$s(t) = \sum_{l=1}^{L(t)} A_l(t) e^{j\theta_l(t)}, \qquad \theta_l(t) = 2\pi f_l(t)t + \theta_l^0 \tag{2.1}$$

where $L(t)$ is the number of sinusoids at time $t$ and $A_l(t)$, $f_l(t)$, $\theta_l^0$ are the amplitude, frequency and phase, correspondingly, of the $l$-th sinusoid. The Harmonic Model (HM), used for voices frames, is a particular case of the SM. It assumes an harmonic relation between the sinusoids:

$$f_l(t) = l \cdot f_0(t). \tag{2.2}$$

In practice, in each time frame the speech signal is assumed to be stationary, so the model parameters - $\{A_l(m), f_l(m), f_0(m), \theta_l(m)\}$, where $m$ is the frame index, are estimated as constants, at each frame.

The most popular speech model for voice conversion is the Harmonic Plus Noise Model (HNM) [20]. The speech signal is taken as a sum of two signals - harmonic and stochastic, where the harmonic part, $s_h(t)$, is a sum of several harmonics of the pitch frequency:

$$\begin{aligned} s(t) &= s_h(t) + s_n(t) \\ s_h(t) &= \sum_{l=1}^{L(t)} A_l(t) e^{j2\pi l \cdot f_0(t)t}, \end{aligned} \tag{2.3}$$

and the stochastic part, $s_n(t)$, is modelled as an AR process:

$$s_n(t) = h(t,\tau) * e(t) \tag{2.4}$$

where $*$ denotes convolution, $e(t)$ is white Gaussian noise and $h(t,\tau)$ is assumed to be a time-invariant all pole filter during a single frame, but may vary from frame to frame. During analysis of a voiced frame, the harmonic parameters are first estimated. The stochastic part is then taken as the model-representation residual signal of the estimated harmonic part, $\hat{s}_h(t)$:

$$\hat{s}_n(t) = s(t) - \hat{s}_h(t) \tag{2.5}$$

Finally, the filter $h(t,\tau)$ is estimated from $\hat{s}_n(t)$, regarded as an AR signal, as described in [18]. Unvoiced frames are modelled as purely stochastic.

## 2.2 Spectral Envelope Modeling

The spectral envelope of a speech signal is an important part of the speaker's identity, so it plays an important role in speaker identification and conversion methods. These algorithms mostly operate on feature vectors, each representing the spectral envelope in a certain voiced frame. Selection of a suitable feature space to represent the spectral envelopes is very important, since it can dramatically affect the performance of a speech processing algorithm. Feature properties such as good interpolation characteristics, numerical stability and dimensionality should be considered.

The harmonic amplitudes defined in the HNM are usually not used as the actual feature vectors, since their dimensionality varies according to the pitch frequency, and is usually relatively high. Instead, the harmonic amplitudes, defined in eqn. (2.3), are used to evaluate feature vectors. Two of the most popular feature representations used for voice conversion are Mel Frequency Cepstrum Coefficients (MFCC), [21], and Line Spectral Frequencies (LSF's) [22]. Once the processing stage is completed, the amplitudes are reconstructed from the processed vectors, and are used for synthesizing the output speech signal. In this work the speech signals are analyzed and synthesized using the HNM, with MFCC's as feature vectors that represent the spectral envelope in every voiced frame.

During spectral envelope conversion, the spectral feature vectors related to a source speaker are converted, to match the spectral envelope characteristics of the target speaker, and new harmonic amplitudes are evaluated to reconstruct the converted harmonic signal. The stochastic part is usually not converted. There has been an attempt to convert also the spectral representation of the stochastic part, but no significant improvement was attained, and sometimes it even damaged the quality of the converted signal [23].

# Chapter 3

# Background On Voice Conversion

## 3.1 Problem Formulation

### 3.1.1 Prosody Modification

The identity of the speaker is related to the spectral envelope of the speech signal, and to its prosody parameters: pitch, energy and duration. Most voice conversion methods aim to transform the spectral envelope of the source speaker, to the spectral envelope of a target speaker, as described in Sec. 3.1.2. The duration of the converted speech signal can be adjusted to the mean rate of target speaker using parametric methods as suggested in [24], or non-parametric methods such as Time-Domain Pitch-Synchronous Overlap and Add (TD-PSOLA), proposed in [25].

The simplest pitch conversion method is based on a linear transformation evaluated by the global mean values of the pitch frequency: (see [2]):

$$\tilde{f_0}^y = \mu^{\left(f_0^y\right)} + \frac{\sigma^{\left(f_0^y\right)}}{\sigma^{\left({f_0}^x\right)}} \left({f_0}^x - \mu^{\left({f_0}^x\right)}\right) \tag{3.1}$$

where ${f_0}^x$ and ${f_0}^y$ are the source and target speakers pitch values, respectively, $\mu$ and $\sigma$ are the corresponding global mean and standard deviation. This method does not change the general shape of the pitch contour, but its offset and scale.

More sophisticated methods for pitch conversion have been proposed, among them: high order polynomial mapping [26], GMM mapping [27], piecewise linear conversion [28] and contour conversion based on codebook prediction [26, 27]. Still, most voice conversion methods use the linear conversion described in eqn. (3.1) for its simplicity and its fair results.

### 3.1.2 Spectral-Envelope Conversion

Let $\{\mathbf{x^q}\}_{q=1}^{Q^x} \in \mathbb{R}^P$ be a set of feature vectors, representing the spectral characteristics of a set of sentences said by a source speaker, and $\{\mathbf{y^q}\}_{q=1}^{Q^y} \in \mathbb{R}^P$ a similar set corresponding to a target speaker. In case where the two sets are assumed to be originated from a parallel training set, meaning that the two speakers said the same text, a time alignment is commonly performed to attain a one-to-one correspondence between the source and target feature vectors, so $Q^x = Q^y = Q$. Given a training set (parallel or not), and a new feature vector related to the source speaker, the goal of spectral envelope conversion is to evaluate the corresponding feature vector related to the target speaker. This evaluation is also referred to as a conversion function $\mathcal{F}$:

$$\tilde{\mathbf{y}} = \mathcal{F}\{\mathbf{x}\} \tag{3.2}$$

### 3.1.3 Objective Performance Measures

#### Log Spectral Distortion (LSD)

The mean spectral distance between a converted signal and the target signal is commonly evaluated in terms of Log Spectral Distortion (LSD). Given two time aligned sequences of feature vectors:

$$\begin{aligned} \mathbf{X}_{1:T} &= \{\mathbf{x^1}, \mathbf{x^2}, ..., \mathbf{x^T}\} && (3.3)\\ \mathbf{Y}_{1:T} &= \{\mathbf{y^1}, \mathbf{y^2}, ..., \mathbf{y^T}\}, && (3.4) \end{aligned}$$

their LSD in $dB$ is defined by:

$$\overline{LSD}\left(\{\mathbf{X}\}^{(1\ldots T)}, \{\mathbf{Y}\}^{(1\ldots T)}\right) = \frac{10}{T}\sum_{m=1}^{T}\sqrt{\frac{1}{2\pi}\int_{-\pi}^{\pi}\left(log_{10}\left|A_x^m(\theta)\right|^2 - log_{10}\left|A_y^m(\theta)\right|^2\right)^2 d\theta}, \tag{3.5}$$

where $A_x^m(\theta)$ and $A_y^m(\theta)$ are the spectral envelopes evaluated from the feature vectors $\mathbf{x}^m$ and $\mathbf{y}^m$, accordingly.

The feature vectors used in this work are MFCC's. Using this parametrization, the LSD between two spectral envelopes, $A_x(\theta)$ and $A_y(\theta)$, can be estimated using the Euclidean distance between their corresponding feature vectors, $\mathbf{x}$ and $\mathbf{y}$, [2] :

$$LSD(\mathbf{x}, \mathbf{y}) \approx \frac{10}{ln10}\sqrt{2\sum_{p=1}^{P}\|x(p) - y(p)\|^2}, \tag{3.6}$$

where $x(p)$ and $y(p)$ are the $p$-th elements of the source and target Cepstrum vectors correspondingly, and $P$ is the length of the cepstral feature vectors.

A Normalized Distortion (ND) is used to obtain a fair comparison between conversion sequences of different source-target pairs: the mean spectral distortion between the converted and target signals is normalized by the mean spectral distortion between the source and target signals [29]:

$$\mathrm{ND}\left(\tilde{\mathbf{Y}}_{1:T}, \mathbf{Y}_{1:T}\right) = \frac{\sum_{m=1}^{T}\mathrm{LSD}\left(\tilde{\mathbf{y}}^m, \mathbf{y}^m\right)}{\sum_{m=1}^{T}\mathrm{LSD}\left(\mathbf{x}^m, \mathbf{y}^m\right)}, \tag{3.7}$$

where $\tilde{\mathbf{Y}}_{1:T} \triangleq \left(\tilde{\mathbf{y}}^1, \tilde{\mathbf{y}}^2, \ldots, \ \tilde{\mathbf{y}}^T\right)^\top$ is the converted sequence.

### Normalized Global Variance (NGV)

The Global Variance (GV) of the $p$-th elements of a sequence, $\tilde{\mathbf{Y}}_{1:T}$, representing a converted speech utterance, is:

$$\sigma^2_{\tilde{\mathbf{Y}}_{1:T}}(p) = \frac{1}{T}\sum_{m=1}^{T}\left(\tilde{y}^m(p) - \frac{1}{T}\sum_{\tau=1}^{T}\tilde{y}^\tau(p)\right)^2, \tag{3.8}$$

In this work we use a Normalized Global Variance (NGV) to measure the variability of a sequence of converted vectors:

$$\mathrm{NGV}\left\{\tilde{\mathbf{Y}}_{1:T}\right\} \triangleq \frac{1}{P}\sum_{p=1}^{P}\frac{\sigma^2_{\tilde{\mathbf{Y}}_{1:T}}(p)}{\sigma^2_{\mathbf{Y}}(p)}, \tag{3.9}$$

where $\sigma^2_{\mathbf{Y}}(p)$ is the empirical GV of the $p$-th elements of the target speaker, obtained from the target training vectors:

$$\sigma^2_{\mathbf{Y}}(p) = \frac{1}{Q_y}\sum_{k=1}^{Q_y}\left(y^k(p) - \frac{1}{Q_y}\sum_{n=1}^{Q_y}y^n(p)\right)^2. \tag{3.10}$$

Note that the target GV defined in eqn. (3.10) is evaluated by averaging over the entire training corpus. This evaluation of GV is different from the one proposed in [6] for spectral conversion and GV enhancement, where the GV is separably evaluated for every utterance of the target speaker.

The desired values for these measures are ND → 0 and NGV → 1, indicating that the converted outcome is close to the target signal in terms of spectral similarity and global variance.

## 3.1.4 Subjective Performance Measures

The objective measures presented above indicate trends in comparing the quality of the examined conversion methods. Unfortunately, they do not always correspond to a subjective impression of a human listener. It is common that two converted signals have the same objective measures, but one of them would sound better. That is why it is customary to compare conversion methods by performing subjective listening tests, in addition to objective measures.

Subjective listening tests usually include 10-15 listeners, presented with synthesized outputs of several conversion methods. The listeners are asked to express their opinion regarding the quality and the individuality (similarity to the target speaker) of the given signals. To avoid bias and achieve a statistical validity, each test is repeated several times using different sentences in a randomly selected order.

### Preference Tests

Preference tests are binary evaluations where the listeners are asked to choose between two synthesized signals, each of a different conversion method. Two common preference tests are:

1. AB quality test - the listeners are asked to indicate which sentence (A or B) is of better quality.
2. ABX individuality test - the listeners are asked to indicate which sentence (A or B) is more similar to a reference sentence, marked as X (the target speaker).

Preference tests are most useful for comparing two conversion methods, as they clearly indicate which of the examined methods is preferred by the listeners. However, when several methods are to be compared, it could be tedious for the listeners to compare between each pair of the examined methods. A Multi Stimulus test with Hidden Reference and Anchor (MUSHRA) [30] described next, provides a more compact protocol for comparing several conversion methods.

#### MUSHRA

The purpose of the MUSHRA test [30] is to evaluate the quality of several processed signals compared to a given high quality, usually unprocessed, reference signal. The listeners are presented with several test signals including: outputs of the examined systems, a hidden anchor signal - usually a filtered version of the reference signal, and a hidden reference. The test signals are randomly ordered, and the listeners are not informed about the hidden reference and anchor signals being included in the test set. During evaluation, the listeners are asked to compare the reference signal to the test signals and rate them between 0 to 100, where at least one of the signals must be rated 100, because the hidden reference is identical to the declared reference.

A MUSHRA format can also be used for rating the individuality of the synthesized signals, as conducted by Godony et. al. [31]. The listeners are presented with several synthesized signals (including the hidden reference) and are asked to rate their similarity to the reference signal, in terms of the speaker's identity, while ignoring their perceived quality.

### 3.1.5 Correspondence Between Objective And Subjective Measures

The objective measures presented above aim to reflect some properties of the subjective measures:

- ND - linked to individuality tests: as the ND increases, the converted and target spectra are further, producing converted signals which are less similar to the target speaker.

- NGV - linked to quality preference tests - as the NGV decreases towards zero, the converted signal sounds more muffled and therefore of lower quality.

As stated above, in some cases the objective and subjective evaluations do not agree, especially when the examined conversion systems lead to similar objective performance.

## 3.2 Common Spectral Conversion Methods

### 3.2.1 Data Driven Conversion Approaches

#### Code Book Methods

One of the earliest voice conversion methods, presented in [32], proposed a codebook based conversion. The spectral envelopes the two speakers are represented by a parallel source-target codebook obtained by Vector Quantization (VQ). Each of the source codewords is matched with a weighted sum of target codewords, as its conversion outcome. The weights are determined according to the number of times each source codeword was matched to each of the target codewords, in the training set. During conversion, a test source vector is quantized by the source codebook and its conversion is the suitable target weighted sum. Later, another codebook-based method was proposed in [33], where the spectral features of each phoneme is represented by the centroid of its utterances that appeared in the training set. The conversion output is a weighted sum of the target centroids, where the weights are determined considering the resemblance of the corresponding source test vector to the source centroids. Speech signals produced by these codebook-based methods are reported to suffer from poor quality due to temporal discontinuities of the converted spectral envelope and deficient representation due to a limited target codebook.

#### Unit Selection Methods

Unit selection approaches use the training set related to the target speaker as a diverse codebook. In addition to spectral resemblance to the target, they also consider the temporal continuity of the selected vectors to achieve a smooth evolution of the converted spectra.

A simple unit-selection approach addressing this problem proposed selection by minimizing a cost function, considering both spectral resemblance and temporal continuity of the selected feature vectors [34]. Given a sequence of $T$ source vectors $\mathbf{x}_{1:T}$ and a target training set, $\{\mathbf{y}_k\}_1^Q$, used as a codebook, the conversion of the source sequence is obtained by selecting the target codewords minimizing the following cost function:

$$\tilde{\mathbf{y}}_t = \underset{k\in\{1,\ldots,Q\}}{\operatorname{argmin}} \{\alpha\|\mathbf{y}^k - \mathbf{x}_t\|^2 + (1-\alpha)\,\|\mathbf{y}^k - \tilde{\mathbf{y}}_{t-1}\|^2\}, \tag{3.11}$$

where $\tilde{\mathbf{y}}_{t-1}$ is the converted feature vector at the previous frame. This cost function aims to consider both spectral resemblance and quality of the converted signal: it selects the entry that is most similar to each input source vector, and also similar to the previously selected entry. The parameter $\alpha$ determines the relative weight of similarity versus temporal continuity. Eqn. (3.11) uses the spectral vectors related to the source speaker as reference signals. Obviously, this is not ideal since the two speakers may have considerably difference. An improved approach, proposed by Dutoit et al. uses a converted version of the source vectors as reference signals: instead of comparing the previously selected target vector to the source vectors, they compare it to the converted vectors [35].

### Exemplar-Based Sparse Representation [36]

Exemplar-based conversion is a non-parametric (parallel) method that uses sparse representation to describe a speech spectrogram as a linear combination of a basis spectra:

$$\mathbf{X} \approx \mathbf{A} \cdot \mathbf{H} \quad s.t. \;\; \mathbf{H} \geq 0, \tag{3.12}$$

where $\mathbf{X} \in \mathbb{R}^{F \times M}$ is a high resolution spectrogram extracted from $M$ speech frames, using $F$ dimensional Fast Fourier Transform (FFT), $\mathbf{A} \in \mathbb{R}^{F \times N}$ is a fixed dictionary extracted from the source training set, $N$ is the amount of exemplars taken for this dictionary and $\mathbf{H} \in \mathbb{R}^{N \times M}$ is called an activation matrix. The main assumption of this conversion method is that the source and target speakers share the same activation matrix, if their dictionaries are parallel and aligned, so the converted spectrogram is evaluated as:

$$\mathbf{Y} \approx \mathbf{B} \cdot \mathbf{H}, \tag{3.13}$$

where $\hat{\mathbf{Y}} \in \mathbb{R}^{F \times M}$ is the converted spectrogram, and $\mathbf{B} \in \mathbb{R}^{F \times N}$ is a fixed dictionary extracted from the target training set. Given a test sentence, a spectrogram $\mathbf{X}$ is extracted, the activation matrix $\mathbf{H}$ is evaluated by minimizing the following cost function:

$$\mathbf{H} = \underset{\mathbf{H} \geq \mathbf{0}}{\text{argmin}} d\left(\mathbf{X}, \mathbf{AH}\right) + \lambda \left\|\mathbf{H}\right\|_1, \tag{3.14}$$

where $d\left(\cdot, \cdot\right)$ is the generalized Kullback-Leibler (KL) divergence and $\left\|\cdot\right\|_1$ is added as a regularizer encouraging sparsity. The minimizer of eqn. (3.14) is obtained by an iterative process according to a multiplicative update rule:

$$\mathbf{H} \leftarrow \mathbf{H} \bigotimes \frac{\mathbf{A}^\top \frac{\mathbf{X}}{\mathbf{AH}}}{\mathbf{A}^\top + \lambda}, \tag{3.15}$$

where divisions are element-wise and $\bigotimes$ is an element-wise multiplication.

As opposed to codebook and unit selection methods, requiring large scale training sets to achieve a reasonable quality, this method achieve high quality signals using just 10 sentences. The main disadvantage of this method is its high computational complexity (45 times higher than the classical GMM-based method described in 3.2.2) and memory footprint (295 times higher than GMM-based conversion).

## 3.2.2 GMM Conversion

The commonly used statistical voice conversion is based on training a Gaussian Mixture Model (GMM) as a statistical model for the parallel training set [3]. Let $\mathbf{z}^q$ denote the joint source-target spectral feature vector:

$$\mathbf{z}^q = \left( (\mathbf{x}^q)^T, (\mathbf{y}^q)^T \right)^T, \quad q = 1, ..., Q \tag{3.16}$$

The joint vectors are divided into $M$ classes, and the vectors in every class are assumed to be jointly Gaussian:

$$p\left(\mathbf{z}^q\right) = \sum_{m=1}^{M} p\left(w_m\right) N\left(\mathbf{z}^q; \mu^m, \mathbf{\Sigma}^m\right), \quad q = 1, ..., Q \tag{3.17}$$

where $p\left(w_m\right)$, is the probability of the class $w_m$ and $N\left(\cdot;\mu^m,\mathbf{\Sigma}^m\right)$ is a normal distribution with the parameters:

$$\mu^m = \begin{pmatrix} \mu^{(x),m} \\ \mu^{(y),m} \end{pmatrix}, \qquad \mathbf{\Sigma}^m = \begin{pmatrix} \mathbf{\Sigma}^{(xx),m} & \mathbf{\Sigma}^{(xy),m} \\ \mathbf{\Sigma}^{(yx),m} & \mathbf{\Sigma}^{(yy),m} \end{pmatrix} \tag{3.18}$$

$\mu^{(x),m}$, and $\mu^{(y),m}$ are $P \times 1$ vectors representing the source and target mean vectors of the $m-th$ class and $\mathbf{\Sigma}^{(\cdot\cdot),m}$ is a $P \times P$ covariance matrix. These parameters are usually estimated using the Expectation Maximization (EM) [16] algorithm fed with the parallel training set.

The conversion function can be interpreted as an estimator for a target vector, given a source vector - $E\left[\mathbf{y}|\mathbf{x}\right]$. Since the joint feature vector, $\mathbf{z}$, is modeled as a GMM, the estimator is a linear combination of a simple linear predictor of each class:

$$\mathcal{F}^{(Joint-GMM)}\{\mathbf{x}\} = \sum_{m=1}^{M} p\left(w_m|\mathbf{x}\right)\left(\mu^{(y),m} + \mathbf{\Sigma}^{(yx),m}(\mathbf{\Sigma}^{(xx),m})^{-1}\left(\mathbf{x}-\mu^{(x),m}\right)\right) \tag{3.19}$$

where $p\left(w_m|\mathbf{x}\right)$ is a conditional probability evaluated using the GMM parameters and Bayes' theorem:

$$p\left(w_m|\mathbf{x}\right) = \frac{p\left(w_m\right)N\left(\mathbf{x};\mu^{(x),m},\mathbf{\Sigma}^{(xx),m}\right)}{\sum_{m=1}^{M} p\left(w_m\right)N\left(\mathbf{x};\mu^{(x),m},\mathbf{\Sigma}^{(xx),m}\right)} \tag{3.20}$$

In an earlier work presented be Stylianou et. al. [2], a GMM was trained using only the source training set. The conversion function took the same linear form:

$$\mathcal{F}^{(LS-GMM)}\{\mathbf{x}\} = \sum_{m=1}^{M} p\left(w_m|\mathbf{x}\right)\left(\nu^m + \mathbf{\Gamma}^m(\mathbf{\Sigma}^{(xx),m})^{-1}\left(\mathbf{x}-\mu^{(x),m}\right)\right), \tag{3.21}$$

where the missing conversion parameters, $\{\mathbf{\Gamma}^m,\nu^m\}_{m=1}^{M}$, were evaluated using Least Squares (LS), so that the mean Euclidian distance between the converted and target spectral features is minimized:

$$\min_{\{\mathbf{\Gamma}^m,\nu^m\}_{m=1}^{M}} \sum_{q=1}^{Q} \|\mathcal{F}^{(LS-GMM)}\{\mathbf{x}^q\} - \mathbf{y}^q\|^2. \tag{3.22}$$

More specifically, define the matrices $\mathbf{P}$ and $\mathbf{D}$,

$$\begin{aligned} \{\mathbf{P}\}_{m,q} &= p\left(w_m|\mathbf{x}^q\right) \\ \{\mathbf{D}\}_{m,q} &= p\left(w_m|\mathbf{x}^q\right)\left(\mathbf{x}^q-\mu^{(x),m}\right)^T\left(\left(\Sigma^{(xx),m}\right)^{-1}\right)^T \\ m &= 1,...,M; \quad q=1,...,Q \end{aligned} \tag{3.23}$$

where $\{\bullet\}_{m,q}$ denotes the $(m,q)$ element. Define also the conversion parameters $\mathbf{\Gamma}$ and $\mathbf{V}$:

$$\begin{aligned} \mathbf{\Gamma} &= \begin{pmatrix} \mathbf{\Gamma}^1 & \cdots & \mathbf{\Gamma}^M \end{pmatrix}^T \\ \mathbf{V} &= \begin{pmatrix} \nu^1 & \cdots & \nu^M \end{pmatrix}^T \end{aligned} \tag{3.24}$$

Where $\mathbf{\Gamma}$ and $\mathbf{V}$ are $PM \times P$ and $M \times P$ matrices correspondingly. Denote the target training matrix $\mathbf{Y}$:

$$\mathbf{Y} = \begin{pmatrix} \mathbf{y}^1 & \cdots & \mathbf{y}^Q \end{pmatrix}^T \tag{3.25}$$

So the conversion equations (3.21) can be formulated in a matricieal form:

$$\mathbf{Y} = \begin{pmatrix} \mathbf{P} & \vdots & \mathbf{D} \end{pmatrix} \cdot \begin{pmatrix} \mathbf{V} \\ \cdots \\ \mathbf{\Gamma} \end{pmatrix}, \tag{3.26}$$

and the parameters are estimated using LS:

$$\begin{pmatrix} \hat{\mathbf{V}} \\ \cdots \\ \hat{\mathbf{\Gamma}} \end{pmatrix} = \begin{pmatrix} \mathbf{P}^T\mathbf{P} & \vdots & \mathbf{P}^T\mathbf{D} \\ \cdots & & \cdots \\ \mathbf{D}^T\mathbf{P} & \vdots & \mathbf{D}^T\mathbf{D} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{P}^T\mathbf{Y} \\ \cdots \\ \mathbf{D}^T\mathbf{Y} \end{pmatrix} \tag{3.27}$$

Training a full covariance GMM involves estimation of $M\left(P^2+P\right)+M$ parameters, so it requires a large scale training set. The commonly used diagonal relaxation assumes that the spectral features are statistically independent, so the covariance matrices $\mathbf{\Sigma}^{(xx),m}$ are diagonal. In this case, there are $2PM+M$ parameters only to be evaluate, so the training process requires a much smaller training set. In addition, the matrices $\{\mathbf{\Gamma}^m\}_{m=1}^M$ are also diagonal, so their evaluation defined in (3.22) can be separated to $P$ independent minimization problems - one for every coordinate, $p=1,...,P$:

$$\min_{\{\gamma_p^m,\nu_p^m\}_{m=1}^M} \sum_{q=1}^{Q} \|\mathcal{F}\{x_p^q\} - y_p^q\|^2 \tag{3.28}$$

where $\{\gamma_p^m\}_{m=1}^M$ are the $(p,p)$ elements of $\{\mathbf{\Gamma}^m\}_{m=1}^M$, and $\{\nu_p^m\}_{m=1}^M$ are the $p$-th elements of $\{\nu^m\}_{m=1}^M$.

### 3.2.3 Non-Parallel Conversion

Many voice conversion systems require parallel training sets of the source and target speakers, among them the classical GMM method described in Sec. 3.2.2. Their training process is based on having some prior knowledge regarding the correspondence between the source and target spectral feature vectors.

In a non-parallel setup, no assumptions are made regarding the content of the training sentences. The source-target correspondence is not straightforward as in the parallel case, thus presenting a greater challenge. Some non-parallel methods bypass this problem by modeling the two speakers separately and perform alignment or adaptation of the model parameters [37, 38]. Some train a conversion using an additional parallel set and adapt its parameters to the desired target speaker [39, 40].

A different approach for non-parallel training called Iterative combination of a Nearest Neighbor search step and a Conversion step Alignment method (INCA), was recently proposed [1]. This approach provides a framework for applying parallel training techniques using non-parallel training sets. It is based on an iterative evaluation of an auxiliary conversion function and matching functions between the source and target vectors. Convergence of this process was demonstrated using empirical evaluations, but, as indicated by the authors of INCA, the alignment process is prone to phonetic mismatch. To smooth these errors they train their auxiliary conversion function using the classical GMM-based method, which is known to have over-smoothing characteristics.

### INCA

Let $X = \left\{\mathbf{x}^k\right\}_{k=1}^{N_x}, Y = \left\{\mathbf{y}^j\right\}_{j=1}^{N_y} \in \mathbb{R}^P$ be two (non-parallel) training sets of feature vectors related to source and target speakers. The training process is based on an iterative evaluation of a parallel auxiliary conversion function, $\mathcal{F}(\cdot)$, its inverse, and two matching functions between the source and target vectors:

$$\begin{cases} p(k) = j & \text{if } \mathbf{x}^k \text{ matches } \mathbf{y}^j \\ q(j) = k & \text{if } \mathbf{y}^j \text{ matches } \mathbf{x}^k. \end{cases} \tag{3.29}$$

where $k = 1, ..., N_x$ and $j = 1, ..., N_y$. Therefore, each vector $\mathbf{x}$ is matched through $p$ to a single vector at the target, and each vector $\mathbf{y}$ is matched through $q$ to a single vector at the source.

The iterative process begins by initializing at $t = 0$ an auxiliary conversion function to be the identity function: $\mathcal{F}_0(\mathbf{x}) = \mathbf{x}$. In each iteration, the two matching functions, $p_t(\cdot)$ and $q_t(\cdot)$, are evaluated using a nearest neighbor search between converted source vectors and the target vectors, and vice versa, based on the previous auxiliary function $\mathcal{F}_{t-1}$:

$$\begin{aligned} p_t(k) &= \underset{j}{\operatorname{argmin}} \left\|\mathcal{F}_{t-1}\left(\mathbf{x}^k\right) - \mathbf{y}^j\right\|^2 \\ q_t(j) &= \underset{k}{\operatorname{argmin}} \left\|\mathbf{x}^k - \mathcal{F}_{t-1}^{-1}\left(\mathbf{y}^j\right)\right\|^2, \end{aligned} \tag{3.30}$$

These matching functions define a parallelized training set, $\left\{\left(\mathbf{x}^k, \mathbf{y}^{p(k)}\right), \left(\mathbf{x}^{q(j)}, \mathbf{y}^j\right)\right\}$, which reduces the training process of the auxiliary function, $\mathcal{F}_t$, to the parallel case. The simple nearest neighbor search defined in eqn. (3.30) often leads to alignment errors, where vectors related to different phonemes are matched. To reduce the influence of miss-aligned vectors, the classical GMM-based conversion, known for its smoothing characteristics, is used to train the auxiliary function.

Convergence is measured via the mean squared-error (MSE) between the converted sets and the original sets:

$$\begin{aligned} D_t = \frac{1}{N_x + N_y} &\left(\sum_{k=1}^{N_x} \left\|\mathcal{F}_t\left(\mathbf{x}^k\right) - \mathbf{y}^{p_t(k)}\right\|^2\right. \\ &\left.+ \sum_{j=1}^{N_y} \left\|\mathbf{x}^{q_t(j)} - \mathcal{F}_t^{-1}\left(\mathbf{y}^j\right)\right\|^2\right). \end{aligned} \tag{3.31}$$

Erro et al. [1] show that this measure converges empirically. Once convergence is achieved, the conversion function in the last iteration is used for conversion. Alternatively, any other parallel conversion function may be trained, based on the parallelized set using the final matching functions. The overall iterative process is summarized in Table 3.1.

In Ch. 6 we formulate the training process as a minimization problem of a joint cost function, considering both conversion and context-based matching functions. We propose an iterative solution for this minimization problem and prove its convergence.

Table 3.1: INCA - Training stage [1].

**Input:** a non-parallel training set $\{X, Y\}$

**Initialization:** set the initial conversion function to identity $\mathcal{F}_0(x) = x$.

**Main Iteration:** for $t = 1, 2...$ perform the following steps:

1. Evaluate the matching functions, $p_t, q_t$, using eqn. (3.30).

2. Train an auxiliary conversion function using the parallelized training set

3. Evaluate $D_t$ using eqn. (3.31) and check convergence.

**Output:** conversion and matching functions $\mathcal{F}_t, p_t, q_t$.

# Chapter 4

# Global Variance Enhancement

Due to the averaging process used in statistical modeling, GMM-based methods produce overly smoothed spectral envelopes, leading to muffled synthesized outputs. Still, the classical GMM-based conversion method trained either by LS estimation [2] or by joint GMM training [3] are two of the most popular approaches for spectral voice conversion to date. Several modifications of the GMM-based conversion have been proposed since [5, 41, 42]. Yet, these GMM-based conversion methods still produce muffled output speech, apparently due to excessive smoothing of the temporal evolution of the spectral envelope.

Another GMM-based approach [6] aims to capture the temporal evolution of the spectral envelope. It uses Maximum Likelihood (ML) estimation to train a conversion based on aligned sequences of the source and target spectral feature vectors. This approach also enhances the Global Variance (GV) of the spectral features, thus increasing their dynamic range, and hence decreasing the muffling effect.

In this work we present two methods dealing with GV enhancement (see Ch. 4): 1) using the framework of the classical GMM training, while constraining the GV of the converted feature vectors to match its evaluated value for the target speaker [43]. 2) a GV enhancement module, designed independently of a specific conversion procedure [44]. Given a sequence of converted feature vectors, the module evaluates their enhanced version by maximizing their GV, under a spectral distortion constraint.

## 4.1 Voice Conversion using GMM with Enhanced Global Variance

In this section we present an approach for GV enhancement using the classical conversion proposed in [2]. We formalize the training process as a constrained least squares minimization problem: the mean distance between the converted and target features is minimized under the constraint that the GV of the converted features should match the GV of the target features. Objective tests show that compared to the classical method, the proposed approach increases the GV of the spectral features, but the spectral similarity to the target is somewhat reduced. Nevertheless, subjective evaluations indicate that the output of the constrained conversion is preferable in terms of both quality and similarity to the target.

### 4.1.1 Training Stage

As described in [2], a matrix form of (3.28) can be formulated as:

$$\min_{\mathbf{q}^p} \|\mathbf{A}^p \mathbf{q}^p - \mathbf{y}_p\|^2, \tag{4.1}$$

where $Q$ is the amount of training vectors, $M$ is the number of Gaussians in the trained GMM, $\mathbf{y}_p$ is a $Q \times 1$ vector including the $p$-th element of all the target training vectors, $\mathbf{A}^p$ is a $Q \times 2M$ matrix defined in (4.3), and $\mathbf{q}^p$ is a $2M \times 1$ vector including the conversion parameters defined in (4.4).

$$\mathbf{y}_p \triangleq \begin{pmatrix} y_p^1 & \cdots & y_p^Q \end{pmatrix}^T, \tag{4.2}$$

$$\begin{aligned}
\mathbf{A}^p &\triangleq \begin{pmatrix} \mathbf{P} & \vdots & \mathbf{D}^p \end{pmatrix} \\
\{\mathbf{P}\}_{m,q} &= p\left(w_m|\mathbf{x}^q\right) \\
\{\mathbf{D}^p\}_{q,m} &= p\left(w_m|\mathbf{x}^q\right) \frac{1}{\sigma^{q,m}} \left(\mathbf{x}_p^q - \mu_p^{(x),m}\right) \\
m &= 1, ..., M; \quad q = 1, ..., Q
\end{aligned} \tag{4.3}$$

$$\mathbf{q}^p \triangleq \begin{pmatrix} \nu_p^1 & \cdots & \nu_p^M & \vdots & \gamma_p^1 & \cdots & \gamma_p^M \end{pmatrix}^T. \tag{4.4}$$

The LS solution of (4.1) is given by:

$$\hat{\mathbf{q}}^p = \left(\mathbf{A}^{pT}\mathbf{A}^p\right)^{-1} \mathbf{A}^{pT}\mathbf{y}_p \tag{4.5}$$

The GV of the $p$-th element of the target feature vectors can be evaluated by:

$$Var\{\mathbf{y}_p\} \simeq \frac{1}{Q}\sum_{q=1}^{Q}\left(y_p^q - \frac{1}{Q}\sum_{r=1}^{Q} y_p^r\right)^2 \tag{4.6}$$

where $\mathbf{y}_p$ is the $Q \times 1$ vector defined in (4.2). A matrix form of the r.h.s. of (4.6) is:

$$\frac{1}{Q}\sum_{q=1}^{Q}\left(y_p^q - \frac{1}{Q}\sum_{r=1}^{Q} y_p^r\right)^2 = \|\mathbf{\Delta} \cdot \mathbf{y}_p\|^2 \triangleq c_p^2, \tag{4.7}$$

where $\mathbf{\Delta}$ is a $Q \times Q$ matrix defined by:

$$\mathbf{\Delta} \triangleq \frac{1}{\sqrt{Q}} \left( \mathbf{I}_{Q\times Q} - \frac{1}{Q} \begin{pmatrix} 1 & \cdots & 1 \\ \vdots & \cdots & \vdots \\ 1 & \cdots & 1 \end{pmatrix} \right) \tag{4.8}$$

Similarly, the GV of the $p$-th element of the converted vectors can be evaluated by:

$$Var\{\mathcal{F}\{\mathbf{x}_p^q\}\} \simeq \|\mathbf{\Delta} \cdot \mathbf{A}^p\mathbf{q}^p\|^2 = \|\mathbf{B}^p\mathbf{q}^p\|^2, \tag{4.9}$$

where $\mathbf{B}^p \triangleq \mathbf{\Delta} \cdot \mathbf{A}^p$.

In order to enhance the GV of the converted elements, while minimizing the mean Euclidian distance between the converted and target vectors, we propose a constrained formulation:

$$\begin{array}{l} \min\limits_{\mathbf{q}^p} \|\mathbf{A}^p \mathbf{q}^p - \mathbf{y}_p\|^2 \\ s.t. \;\; \|\mathbf{B}^p \mathbf{q}^p\|^2 = c_p^2 \end{array} \quad ; \quad p = 1, ..., P \tag{4.10}$$

where $c_p^2$ is the evaluated GV of the target, defined in (4.7).

The $P$ constrained minimization problems defined in eqn. (4.10) can be solved by using the Lagrange Multiplier method and joint diagonalization of the pairs $\{\mathbf{A}^p, \mathbf{B}^p\}_{p=1}^{P}$, as described in [45].

### 4.1.2 Experimental Results

#### Experiment Setup

We used two U.S. English male speakers from the CMU ARCTIC database [46]: 50 parallel sentences were used for training and 50 other parallel sentences for testing, all sampled at 16kHz and phonetically annotated. Analysis and synthesis were performed using the Harmonic Plus Noise Model (HNM) [20] by the toolkit available at [47]. The first 24 Mel Frequency Cepstrum Coefficients (MFCC's) were extracted using the harmonic amplitudes as described in [21]. The analysis frames were time aligned and the feature vectors were matched using a Dynamic Time Warping (DTW) algorithm based on the phonetic labeling as described in [48].

The dynamic range of the cepstral coefficients in natural speech usually decreases as their order increases, so enhancement of the high order coefficients in the converted signal is not as important as for low order coefficients. The training stage described above includes high computational complexity when solving the constrained minimization problems defined in eqn. (4.10), therefore, the GV was enhanced only for cepstral coefficients lower than a specific threshold $P_0 = 12$. For $p > P_0$ the conversion parameters were evaluated using the classical, unconstrained approach. Several GMM models were examined using $(4, 8, 16, 32, 64)$ mixtures and the final value was set to 32 as it lead to the lowest spectral distortion. The pitch was converted linearly as described in eqn. (3.1).

The performance of our proposed conversion method was examined and compared to the performance of classical conversion [2], using both objective and subjective measures. To reduce audible artifacts converted outcomes by both methods were processed. Before synthesis, the temporal evolution of each cepstral coefficient was filtered by $\mu + (1 - \mu) z^{-1}$, using $\mu = 0.5$. After synthesis, the waveforms were filtered using a low-pass filter having a 5kHz cut-off frequency.

### 4.1.3 Objective Evaluations

The similarity of the converted signals to the target signals was evaluated using mean LSD and NGV as described in Sec. 3.1.3.

As seen in Table 4.1, the proposed approach (noted as CGMM) increased the mean normalized GV from 10% to 90% of its natural value, at the expense of a degradation of $1.1dB$, in the LSD. The mean normalized GV achieved by the constrained approach did not reach 100%, though it was constrained to match the natural value of the target signal, since the test sentences were not included in the training set. We trained the proposed approach by multiplying the target NGV in the constraint term with a factor smaller than 1 (specifically, 0.3), to achieve an intermediate working point: the mean LSD was increased just by 0.2dB compared to the classical GMM, but the NGV reached only to 30% of its natural value. We used the proposed CGMM method with a factor equal to 1 for the subjective evaluation tests presented in the next section, since this value leads to the best quality in our informal listening tests.

Table 4.1: *Objective performance of the proposed approach (labeled as Constrained GMM) compared to the Classical GMM-based method.*

| Conversion Method | Mean LSD [dB] | NGV |
|---|---|---|
| Classical GMM | 6.2 | 0.1 |
| CGMM with a factor 0.3 | 6.4 | 0.3 |
| CGMM | 7.3 | 0.9 |

### 4.1.4 Subjective Evaluations

Listening tests were used to evaluate the performance of the proposed constrained approach, compared to the classical method, in terms of quality and individuality. We conducted two quality tests: an AB preference test and a MUSHRA quality test [30], and an ABX individuality preference test as described in Sec. 3.1.4. In every test, 10 different sentences were examined by 10 listeners which included 20-30 years old, non-experts, men and women.

In the quality AB preference test the examined signals were outputs of the proposed constrained method and outputs of the classical conversion method. The results, presented in Fig. 4.2, indicate that the enhanced output was almost always (about 95% of the time) preferred by the listeners.

In the MUSHRA quality tests the listeners were presented with 50 signals, overall. Ten on them were the original (unprocessed) target sentences, used as reference signals. Four versions were presented as test samples (10 sentences of each): (1) A converted outcome by the proposed method. (2) A converted outcome by the classical method. (3) A hidden anchor - the target signal, low-pass filtered with a 3.5kHz cut-off frequency. (4) A hidden reference - the original unprocessed target signal. All of the listeners rated the hidden target signal as 100, and the anchor received a mean score of 80. The grades of the converted outputs presented in Fig. 4.1(a), demonstrate the improved quality achieved by the proposed constrained approach, compared to the classical approach.

In the ABX individuality test, the listeners were presented with two converted outputs (by the proposed

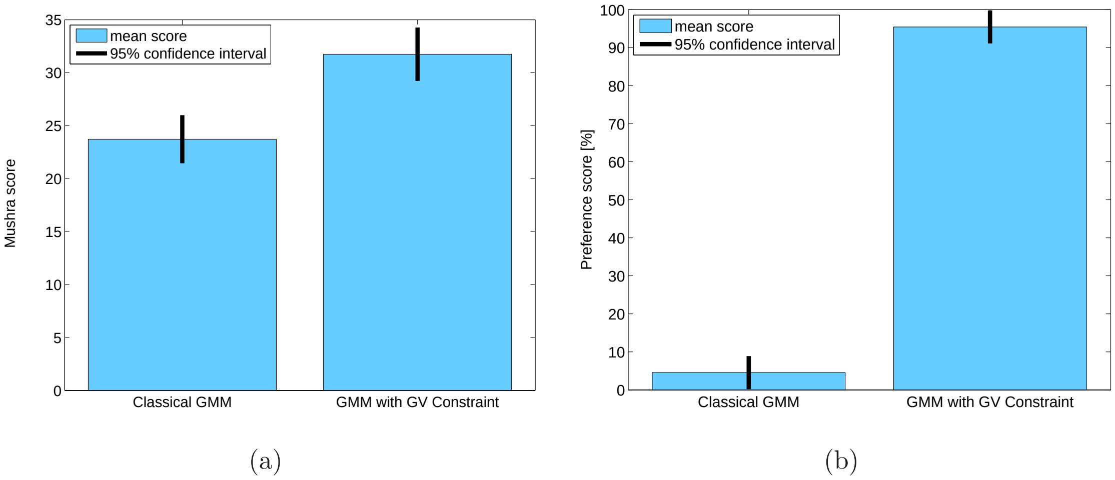


Figure 4.1: The classical GMM conversion method [2] compared with the proposed CGMM: (a) - MUSHRA quality test; (b) - preference quality test (AB).

and classical methods), randomly marked as A or B, and a processed version of the target signal, marked as X. The target signal was processed by the same tools used for processing the converted outputs. First, the target waveform was analyzed, and its cepstral coefficients were filtered by $\mu + (1-\mu)\,z^{-1}$, using $\mu = 0.5$. Then the waveform was re-synthesized and filtered using a low-pass filter having a 5kHz cut-off frequency. The results of the individuality preference test are presented in Fig. 4.2. They indicate that in 75% of the tests the enhanced outputs were perceived as more similar to the target signal than those obtained by the classical method, even though the constrained approach suffers from some degradation in terms of mean spectral distance.

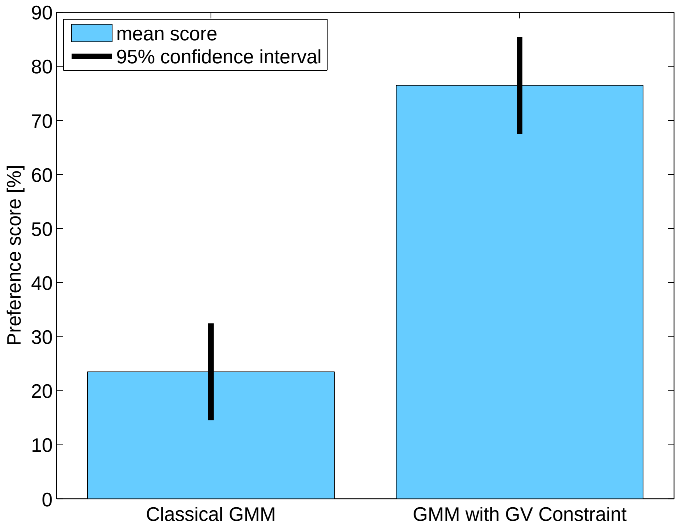


Figure 4.2: ABX preference individuality test - the classical GMM conversion against the proposed constrained conversion.

## 4.2 Modular Global Variance Enhancement

In this section we present a method for GV enhancement, designed independently of a specific conversion procedure. As opposed to other previously proposed methods, where the GV enhancement is intergraded into the conversion process [6, 43], the proposed method is applied as a post-processing block. Given a sequence of converted feature vectors, their enhanced version is obtained by maximizing their GV, under a spectral distortion constraint. The GV of the enhanced sequences is increased up to the level where the mean spectral distance between the converted sequence and its enhanced version reaches a preset threshold value. This threshold enables the user to control the individuality-quality tradeoff: as the allowed spectral distance increases, the GV can be further increased. Therefore, the GV enhancement improves the quality of the synthesized output (sounds less muffled), at the expense of some degradation in the individuality - the similarity of the converted signal to the target speaker. A naive approach to increase the GV would be to just add white noise to the MFCC parameters with a variance determined by a threshold. As expected, our informal listening tests showed that it results in noisy converted speech and is not a viable approach.

We evaluated our GV enhancement method by applying it as a post-processing block on converted outcomes of the classical GMM method [2]. The enhanced sentences were compared to the original converted sentences, and also to sentences converted (with integrated enhancement) by the Constrained GMM (CGMM) method [43]. Listening tests showed that the proposed GV enhancement method improved the quality of sentences converted by the classical GMM method [2]. In addition, most listeners preferred these results over converted sentences obtained by CGMM [43], both in terms of quality and individuality.

### 4.2.1 GV Enhancement Module

Let $\tilde{\mathbf{Y}}_{1:T}$ be a $T \times P$ matrix consisting of a sequence of $T$ converted feature vectors:

$$\tilde{\mathbf{Y}}_{1:T} \triangleq \begin{pmatrix} \tilde{\mathbf{y}}_1, & \tilde{\mathbf{y}}_2, & \dots & , \tilde{\mathbf{y}}_T \end{pmatrix}^\top, \tag{4.11}$$

where $\{\tilde{\mathbf{y}}_t\}_1^T \in \mathtt{R}^P$.

Let $\tilde{\mathbf{Z}}_{1:T}$ be a $T \times P$ matrix comprising the enhanced version of the converted sequence $\tilde{\mathbf{Y}}_{1:T}$. We set the enhanced sequence as the solution of the following problem:

$$\begin{aligned} \tilde{\mathbf{Z}}_{1:T} &= \underset{\mathbf{Z}_{1:T}}{\operatorname{argmax}} \; \{\mathrm{NGV}\{\mathbf{Z}_{1:T}\}\} \\ & \quad s.t \quad \overline{\mathrm{LSD}}\left(\mathbf{Z}_{1:T}, \tilde{\mathbf{Y}}_{1:T}\right) \leq \theta_{LSD}, \end{aligned} \tag{4.12}$$

where $\overline{\mathrm{LSD}}\left(\mathbf{Z}_{1:T}, \tilde{\mathbf{Y}}_{1:T}\right)$ is the mean log-spectral distance (defined in (4.17) bellow) between the enhanced and converted sequences, $\tilde{\mathbf{Y}}_{1:T}$ and $\mathbf{Z}_{1:T}$, correspondingly, and $\theta_{LSD}$ is a pre-set threshold for the mean LSD in dB. If this threshold is set to zero, the constraint is disabled and the converted sequence remains unchanged. For any positive value, the NGV of the enhanced sequence is higher than the NGV of the converted sequence $\tilde{\mathbf{Y}}_{1:T}$, while the $\overline{\text{LSD}}$ between these two sequences is not higher than $\theta_{LSD}$.

In order to obtain the enhanced sequence, we now further develop (4.12) in terms of explicit expressions for NGV and $\overline{\text{LSD}}$. Define $\mathbf{C}$ as a diagonal $P \times P$ matrix, comprising the GV of the target spectral features, evaluated by (3.10):

$$\mathbf{C} \triangleq diag\left(\sigma^2_{\mathbf{Y}}(1)\}, \sigma^2_{\mathbf{Y}}(2)\}, \dots, \sigma^2_{\mathbf{Y}}(P)\}\right). \tag{4.13}$$

Like in [43] we define a covariance operator, $\mathbf{\Delta}_T$:

$$\mathbf{\Delta}_T \triangleq \frac{1}{\sqrt{T}} \left( \mathbf{I}_{T\times T} - \frac{1}{T}\mathbf{J}_T \right) \in \mathtt{R}^{T\times T}, \tag{4.14}$$

where $\mathbf{J}_T$ is a $T \times T$ matrix of all ones. Using (3.8), (4.13) and (4.14), we write the NGV of the converted sequence $\tilde{\mathbf{Y}}_{1:T}$ as:

$$\text{NGV}\{\tilde{\mathbf{Y}}_{1:T}\} = \frac{1}{P}\|\Delta_T \cdot \tilde{\mathbf{Y}}_{1:T} \cdot \mathbf{C}^{-\frac{1}{2}}\|_2^2. \tag{4.15}$$

If Mel Frequency Cepstral Coefficients (MFCCs) are used as spectral features, the mean LSD, between each converted vector $\tilde{\mathbf{y}}_t$ and its enhanced version $\tilde{\mathbf{z}}_t$ can be evaluated using the Euclidian distance between them (see eqn. (3.6):

$$\text{LSD}\left(\tilde{\mathbf{z}}_t, \tilde{\mathbf{y}}_t\right) \approx \kappa\|\tilde{\mathbf{z}}_t - \tilde{\mathbf{y}}_t\|_2 \;\; [dB], \tag{4.16}$$

where $\tilde{y}_t(p)$ and $\tilde{z}_t(p)$ are the $p$-th element of the $t$-th time frame of the converted and enhanced sequences, correspondingly, and $\kappa \triangleq 10\sqrt{2}/ln10$.

Therefore, the mean LSD between the two sequences is approximated by:

$$\begin{aligned}\overline{\text{LSD}}\left(\tilde{\mathbf{Z}}_{1:T}, \tilde{\mathbf{Y}}_{1:T}\right) &\approx \frac{\kappa}{T}\sum_{t=1}^{T}\|\tilde{\mathbf{z}}_t - \tilde{\mathbf{y}}_t\|_2 \\ &= \frac{\kappa}{T}\|\tilde{\mathbf{Z}}_{1:T} - \tilde{\mathbf{Y}}_{1:T}\|_{2,1},\end{aligned} \tag{4.17}$$

where $\|\bullet\|_{2,1}$ is the mixed $\ell_{2,1}$ norm:

$$\|\tilde{\mathbf{Z}} - \tilde{\mathbf{Y}}\|_{2,1} = \sum_{t=1}^{T}\sqrt{\sum_{p=1}^{P}\left(\tilde{z}_t(p) - \tilde{y}_t(p)\right)^2}. \tag{4.18}$$

Using (4.15) and (4.17) we formulate (4.12) as:

$$\begin{aligned}\tilde{\mathbf{Z}}_{1:T} &= \underset{\mathbf{Z}_{1:T}}{\text{argmax}} \;\; \|\mathbf{\Delta}_T\mathbf{Z}_{1:T}\mathbf{C}^{-\frac{1}{2}}\|_2^2 \\ & \quad s.t. \quad \|\mathbf{Z}_{1:T} - \tilde{\mathbf{Y}}_{1:T}\|_{2,1} \leq \frac{T\theta_{LSD}}{\kappa}.\end{aligned} \tag{4.19}$$

We solve the problem by minimizing the Lagrangian:

$$\begin{aligned}\mathcal{L}\left(\mathbf{Z}_{1:T}\right) &= -\|\mathbf{\Delta}_T\mathbf{Z}_{1:T}\mathbf{C}^{-\frac{1}{2}}\|_2^2 + \\ &+ \lambda\left(\|\mathbf{Z}_{1:T} - \tilde{\mathbf{Y}}_{1:T}\|_{2,1} - \frac{T\theta_{LSD}}{\kappa}\right)\end{aligned} \tag{4.20}$$

We diagonalize the covariance operator, $\mathbf{\Delta}_T = \mathbf{USV}^\top$, and denote:

$$\begin{aligned}
\mathbf{\Psi} &\triangleq \mathbf{V}^\top \mathbf{Z}_{1:T} \\
\mathbf{\Phi} &\triangleq \mathbf{V}^\top \tilde{\mathbf{Y}}_{1:T} \\
\mathbf{\Omega} &\triangleq \mathbf{\Psi} - \mathbf{\Phi}
\end{aligned} \tag{4.21}$$

Substituting (4.21) in (4.20) we get:

$$\begin{aligned}
\mathcal{L}\left(\mathbf{\Omega}\right) &= -\|\mathbf{S}\left(\mathbf{\Omega}+\mathbf{\Phi}\right)\mathbf{C}^{-\frac{1}{2}}\|_2^2 + \\
&+ \lambda\left(\|\mathbf{\Omega}\|_{2,1} - \frac{T\theta_{LSD}}{\kappa}\right) = \\
&= -\sum_{t=1}^{T}\sum_{p=1}^{P}\frac{\mathbf{S}_{t,t}^2}{\mathbf{C}_{p,p}}\left(\omega_t\left(p\right)-\phi_t\left(p\right)\right)^2 + \\
&+ \lambda\Big(\sum_{t=1}^{T}\sqrt{\sum_{p=1}^{P}\omega_t^2\left(p\right)} - \frac{T\theta_{LSD}}{\kappa}\Big)
\end{aligned} \tag{4.22}$$

where $\omega_t\left(p\right)$ and $\phi_t\left(p\right)$ are the $(t,p)$ elements of $\mathbf{\Omega}$ and $\mathbf{\Phi}$, respectively. Taking the derivative of this Lagrangian with respect to $\omega_t\left(p\right)$ we get:

$$\begin{aligned}
\frac{\partial\mathcal{L}}{\partial\omega_t\left(p\right)} &= -2\frac{\mathbf{S}_{t,t}^2}{\mathbf{C}_{p,p}}\left(\omega_t\left(p\right)-\phi_t\left(p\right)\right) \\
&+ \lambda\frac{\omega_t\left(p\right)}{\sqrt{\sum_{\rho=1}^{P}\omega_t^2\left(\rho\right)}}
\end{aligned} \tag{4.23}$$

The optimal solution, obtained by setting $\frac{\partial\mathcal{L}}{\partial\omega_t(p)} = 0$, is:

$$\omega_t\left(p\right) = \frac{-\phi_t\left(p\right)}{1-\lambda\mathbf{C}_{p,p}/2\mathbf{S}_{t,t}^2\|\boldsymbol{\omega}_t\|_2}. \tag{4.24}$$

where $\boldsymbol{\omega}_t = \left(\omega_t\left(1\right),...,\omega_t\left(P\right)\right)^\top$. Since $\|\boldsymbol{\omega}_t\|_2$ depends on $\omega_t\left(p\right)$, we use the constraint and set: $\|\boldsymbol{\omega}_t\|_2 = \theta_{LSD}/\kappa$. One of the diagonal elements the matrix $\mathbf{S}$ is zero so to avoid ill conditioning we assume, without loss of generality, that it is the last one and evaluate $\lambda$ using only the first $T-1$ vectors from (4.24):

$$\left(\sum_{t=1}^{T-1}\sqrt{\sum_{p=1}^{P}\omega_t^2\left(p\right)}\right)\left(\lambda\right) = \frac{\left(T-1\right)\theta_{LSD}}{\kappa} \tag{4.25}$$

The Lagrange parameter, $\lambda^*$ can be evaluated by performing grid search and taking the minimal positive value for which (4.25) is approximately sustained. The enhanced sequence is finally obtained by setting $\mathbf{\Psi}\left(\lambda^*\right) = \mathbf{\Omega}\left(\lambda^*\right) + \mathbf{\Phi}$ and $\tilde{\mathbf{Z}}_{1:T}\left(\lambda^*\right) = \mathbf{V}\mathbf{\Psi}\left(\lambda^*\right)$.

During speech synthesis, each converted sequence is substituted by its-GV enhanced version. Consequently, the GV is increased, while the mean LSD between the enhanced and the originally converted sequence is constrained by $\theta_{LSD}[dB]$.

### 4.2.2 Experimental Results

#### Experiment Setup

We used the same setup as described in Ch. 4.1.2. Three conversion schemes were examined: the classical GMM-based conversion [2], the classical GMM-based conversion followed by the proposed GV enhancement scheme, and CGMM [43], also described in Sec. 4.1. The synthesized outputs were evaluated using both objective and subjective measures.

#### Objective Evaluations

We used two objective measures to evaluate the synthesized outputs: mean Log-Spectral Distortion ($\overline{\text{LSD}}$) between the converted and target signals and normalized GV (NGV). MFCCs were used as spectral features, so the mean LSD between the converted and target signals was evaluated using (4.17), and the NGV of the converted signals was evaluated using (4.15). The proposed enhancement method was examined using three threshold values: 1dB, 2dB and 4dB. We also examined the CGMM method using a relaxed GV constraint, by multiplying the target NGV in the constraint term with a factor smaller than 1.

Table 4.2: *Objective performance of the classical GMM-based method (LS-GMM) [2] compared to its enhanced version by the proposed approach and compared to CGMM [43].*

| Conversion Method | Mean LSD [dB] | Mean Norm. GV |
|---|---|---|
| LS-GMM | 6.2 | 0.1 |
| Enhanced, $\theta_{LSD} = 1dB$ | 6.4 | 0.2 |
| CGMM with a factor 0.3 | 6.4 | 0.3 |
| Enhanced, $\theta_{LSD} = 2dB$ | 6.7 | 0.3 |
| Enhanced, $\theta_{LSD} = 4dB$ | 7.3 | 0.4 |
| CGMM | 7.3 | 0.9 |

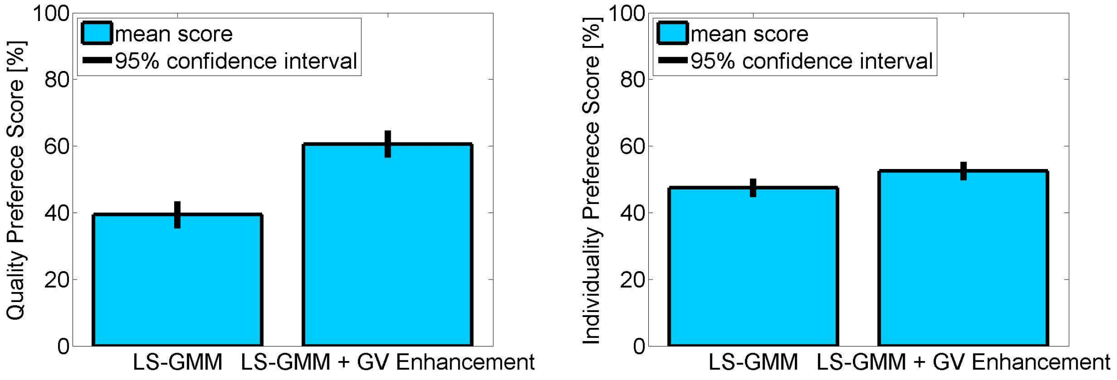
As seen in Table 4.2, the proposed approach increases the NGV of the converted sentences at the expense of their spectral similarity to the target. Allowing a higher distance between the converted and enhanced signals leads to a further increase of the NGV of the enhanced output. In terms of the objective measures we examined, our method was outperformed by CGMM [43]: for the same NGV of 0.3, CGMM (with a factor) achieved a lower LSD than the proposed approach did, and for the same mean LSD of $7.3dB$, CGMM achieved a higher NGV than the proposed approach did. However, listening tests, presented in the next subsection, showed that the proposed approach was preferable by the majority of listeners in terms of both individuality and quality, when compared to the other examined approaches, including CGMM.

### Subjective Evaluations

Listening tests were carried out to subjectively assess the performance of the examined methods. The examined signals were compared using AB quality tests and ABX individuality tests, as described in Sec. 3.1.4. In each test, 10 different randomly ordered sentences were examined by 12 listeners. The group of listeners comprised 20-30 years old non-experts men and women.

We utilized the controlled enhancement to select the best configuration, in terms of subjective quality. We set $\theta_{LSD} = 2$dB, as informal listening tests showed that the proposed enhancement approach produced the best quality with this threshold value. As mentioned above, several working points were also examined for CGMM using factors smaller than 1 multiplying the target NGV in the constraint term. Eventually, we used the CGMM method with a factor equal to 1 since this value leads to the best quality in our informal listening tests.

First, we report the impact of the proposed enhancement on the outputs of the classical conversion method [2]. The results, presented in Fig.4.3(a) show that increasing the GV indeed improved the perceived quality of the converted sentences. Interestingly, the similarity to the target signal was slightly improved, as seen in Fig.4.3(b), even though objectively, the enhanced signal is less similar to the target speaker in terms of mean LSD. This was probably caused by the difficulty of some of the listeners to ignore the signals' quality while rating their individuality.

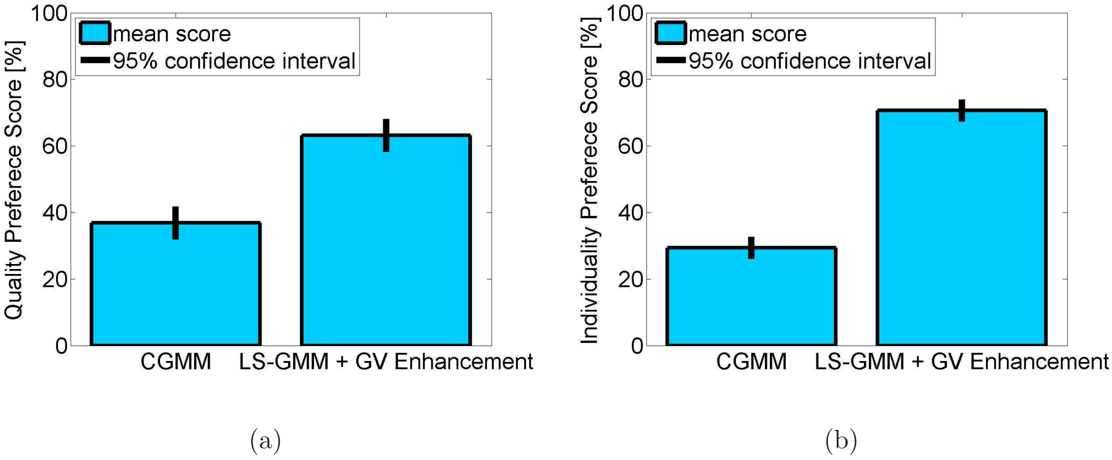


(a) (b)

Figure 4.3: The classical GMM conversion method [2] compared with the classical conversion followed by the the proposed enhancement: (a) - quality preference test; (b) - individuality preference test.

Second, the overall output of the classical conversion followed by the proposed enhancement was compared to the output of CGMM [43]. The proposed enhancement outperformed CGMM: it was preferred in 60% of the cases in terms of quality and in 70% of the cases in terms of similarity to the target, as seen in Figs.4.4(a) and 4.4(b), respectively.

To conclude, the listeners preferred the outputs of the proposed method over the other two methods both in terms of quality and similarity to the target.



(a)

(b)

Figure 4.4: CGMM [43] compared with the classical GMM conversion followed by the the proposed enhancement: (a) - quality preference test; (b) - individuality preference test.

## 4.3 Chapter Summary

The classical spectral envelope conversion approach is based on GMM modeling and linear conversion. This method and several others that were suggested since, are reported to suffer from a muffling effect, ascribed to excessive smoothing of the spectral envelopes.

In this chapter, we proposed two different methods for GV enhancement:

1. CGMM - based on the classical conversion method, where the enhancement process is integrated within the training process of the conversion function.
2. A modular enhancement method, applied as a post-processing block, independent of the conversion process.

The training stage of the CGMM approach formalized as a constrained LS problem. The spectral distance between the converted and target signals is minimized, under a constraint that the GV of the converted features should match the GV of the target sentences. Experimental results show that the CGMM approach significantly increased the GV of the converted spectral features. However, the mean spectral distortion obtained by the proposed approach is somewhat higher than the mean distance achieved by the classical approach. Still, subjective evaluations indicate that the signals obtained by the proposed approach are mostly preferred by the listeners in terms of both quality and similarity to the target speaker, when compared to the converted outputs of the classical method.

The modular enhancement block is designed independently of any specific conversion method. This method is based on GV maximization under a spectral similarity constraint. The extent of enhancement is controlled by tuning the allowed spectral distance between the enhanced and the originally converted signal. We presented a novel formulation for the mean spectral distance between two sequences of feature vectors, so that the threshold value for the spectral distance is specified in [dB]. Experimental results showed that the new enhancement method improved the perceived quality and individuality of the classical GMM conversion method.

# Chapter 5

# Grid-Based Voice Conversion

## 5.1 Background

Most training algorithms require parallel data sets, that is, prerecorded sentences of the source and target speakers saying the same text. In such a setup, evaluation of a conversion function is based on coupled feature vectors - source and target. Alternatively, some methods have been proposed, suggesting training algorithms which avoid the need for pre-alignment altogether. These non-parallel methods need to estimate the source-target correspondence, in addition to the conversion function itself such as TC-INCA presented in Ch. 6 and others, [1, 7]. Although these methods were designed for a non-parallel setup, they can be used in a parallel setup, when aligned data is unavailable.

Even when a parallel training set is available, matching an analysis frame of the source speaker to one of the analysis frames of the target speaker is not straightforward, since the two speakers generally do not pronounce the text at the exact same rate. A time alignment is usually carried out using Dynamic Time Warping (DTW), constrained by starting and ending of speech utterances [48]. These time stamps are commonly obtained by phonetic labeling, representing the beginning and ending of each phoneme. Since the source and target training sentences are not spoken in exactly the same rate, DTW often replicates or omits feature vectors, artificially producing a match. The importance of correct time alignment was recently demonstrated as having a large influence on the quality of the synthesized converted speech [49]. A different approach was suggested by [50], where a statistical model for an eigen-voice was trained using several parallel data-sets. The conversion function is trained using the eigen-voice model and speech sentences related to a target speaker (not necessarily parallel to the source data-sets).

GMM-based conversion methods, using either parallel or non-parallel data, typically require several dozens of sentences for training, and therefore when applied in a mobile environment impose a long recording session on the user. Even the low delay GMM-based approach suggested by Toda at al. was reported to be trained using 60-250 mixtures and 50 training sentences [51]. Therefore applying them in a mobile environment would compel the user to a long recording session.

In this chapter we propose a method for spectral conversion based on a Grid-Based (GB) approximation [52]. We express the spectral conversion process as a sequential Bayesian estimation problem of tracking the target spectrum using observed samples from the source spectrum. We propose models for evaluation of the evidence and likelihood probabilities needed for the GB formulation. Using these approximated probabilities the algorithm sequentially evaluates the converted spectrum as a weighted sum of the target training vectors. Recently, we presented a similar method using GB approximation which requires phonetic labeling during the test stage [53]. In this chapter we propose a modified version of this method, which does not require any labeling for testing. Additionally, as in TC-INCA (Ch. 6), we use context vectors instead of single vectors in order to improve the estimation of the likelihood probability.

Some approaches for training a conversion function that are not based on GMM have been proposed, among them training using a state-space representation [54], and using exemplar-based sparse-representation [36]. Since these methods are closely related to the proposed GB method, we address them and discuss the differences between them and the GB approach in more details after describing the proposed method in this work (see Sec. 5.4). Still, these method are also not suitable for mobile environment since they require several hundreds of parallel training sentences and/or very high computational load during conversion and a substantial memory footprint.

Furthermore, as opposed to previously proposed methods that use parallel and time aligned training sets, the GB conversion approach does not require a one-to-one correspondence between the source and target training vectors. The training process uses parallel sentences but is based on soft correspondence between the source and target vectors, obtained by phonetic labeling of the training sentences without frame alignment, thus eliminating the need for DTW.

Unlike other GMM-based methods that use statistical modeling of the spatial structure of the source and target spectra, the GB method is data-driven, so it is easily trained using merely 5-10 sentences. Its training stage involves simple computations based on the Euclidean distance between the training vectors.

Objective evaluations show that the GB conversion method proposed here leads to GV values that are closer to the GV values of the target speaker than the classical GMM conversion method and to lowest (or very close to it) spectral distance to the target spectra, at the same time. To further improve the quality we applied our GV enhancement post-processing block (see Ch. 4.2). we present an overall scheme, Enhanced-GB (En-GB), consisting of GB conversion, followed by GV enhancement. We used objective measures and also performed extensive subjective evaluations, to compare our proposed En-GB scheme to JGMM, [3], also followed by the same GV enhancement block (En-JGMM), and to CGMM presented in Ch. 4.1.Objectively, En-GB leads to better performance than En-JGMM and CGMM in terms of both spectral distance and GV, using 10 sentences. Listening tests show that in terms of similarity to the target, En-GB outperforms the other examined methods. In terms of quality, CGMM was rated as best, where En-GB was rated as comparable to En-GMM.

## 5.2 Grid-Based Formulation

A brief formulation of sequential estimation using Bayesian tracking is presented in Sec. 5.2.1. In many practical cases, applying this formulation yields a high computational load, which is sometimes unfeasible. The GB method provides a discrete approximation for Bayesian tracking with much less computational complexity, as described in Sec. 5.2.2.

### 5.2.1 Bayesian Tracking

Denote by $\mathbf{y}_t$ a hidden state vector, following a first order Markov dynamics:

$$\mathbf{y}_t = f_t\left(\mathbf{y}_{t-1}, \mathbf{u}_t\right), \tag{5.1}$$

where $f_t$ is a function (not necessarily linear) of $\mathbf{y}_{t-1}$ and of an i.i.d. noise sequence $\mathbf{u}_t$. The observed signal, $\mathbf{x}_t$, depends on the hidden state and on an i.i.d. measurement noise,$\mathbf{v}_t$:

$$\mathbf{x}_t = h_t\left(\mathbf{y}_t, \mathbf{v}_t\right), \tag{5.2}$$

where $h_t\left(\cdot\right)$ may also be non-linear.

The Bayesian optimal estimate for the state vector $\mathbf{y}_t$ in terms of minimizing the mean squared error, given $t$ vectors sequentially sampled from the observed process - $\mathbf{x}_{1:t} \triangleq \{\mathbf{x}_1, ..., \mathbf{x}_t\}$, is obtained by[1]:

$$\hat{\mathbf{y}}_t = E\left[\mathbf{y}_t|\mathbf{x}_{1:t}\right] = \int p\left(\mathbf{y}_t|\mathbf{x}_{1:t}\right)\mathbf{y}_t d\mathbf{y}_t. \tag{5.3}$$

The posterior probability $p\left(\mathbf{y}_t|\mathbf{x}_{1:t}\right)$ can be obtained recursively in two stages:

1. Prediction - obtain the prior probability:

$$p\left(\mathbf{y}_t|\mathbf{x}_{1:t-1}\right) = \int p\left(\mathbf{y}_t|\mathbf{y}_{t-1}\right) p\left(\mathbf{y}_{t-1}|\mathbf{x}_{1:t-1}\right) d\mathbf{y}_{t-1}. \tag{5.4}$$

2. Update - use the current observation $\mathbf{x}_t$ to update the posterior probability:

$$p\left(\mathbf{y}_t|\mathbf{x}_{1:t}\right) = \frac{p\left(\mathbf{x}_t|\mathbf{y}_t\right) p\left(\mathbf{y}_t|\mathbf{x}_{1:t-1}\right)}{p\left(\mathbf{x}_t|\mathbf{x}_{1:t-1}\right)}, \tag{5.5}$$

where,

$$p\left(\mathbf{x}_t|\mathbf{x}_{1:t-1}\right) = \int p\left(\mathbf{x}_t|\mathbf{y}_t\right) p\left(\mathbf{y}_t|\mathbf{x}_{1:t-1}\right) d\mathbf{y}_t. \tag{5.6}$$

This recursion is initialized by setting the prior probability to be equal to the initial probability of the state vector: $p\left(\mathbf{y}_0|\mathbf{x}_0\right) = p\left(\mathbf{y}_0\right)$, where $p\left(\mathbf{y}_0\right)$ is assumed to be known (in practice, mostly taken as a uniform distribution). The likelihood function $p\left(\mathbf{x}_t|\mathbf{y}_t\right)$ that appears in (5.5) is determined according to the measurement model (eqn. (5.2)) and the statistics of the measurement noise $\mathbf{v}_t$.

When the noise signals $\mathbf{u}_t$ and $\mathbf{v}_t$ are Gaussian, and the functions $f_t\left(\cdot\right)$ and $h_t\left(\cdot\right)$ are linear and time invariant (meaning that $f_t\left(\cdot\right) \equiv f\left(\cdot\right)$ and $h_t\left(\cdot\right) \equiv h\left(\cdot\right)$), this recursion can be computed analytically,

[1] In general, any arbitrary integrable function of the state vector $\mathbf{y}_t$ can be evaluated [52].

leading to Kalman filtering [55]. Yet, in most practical cases where these conditions are not sustained, this derivation is hard and often performed using approximation methods such as GB approximation or particle filtering [52]. These methods sequentially evaluate the posterior probability as a discrete weighted sum using a given set of samples in case of GB, or a randomly drawn set in case of Particle Filtering.

In this work, we express the spectral conversion process as a sequential estimation problem tracking the target spectrum, using observed samples from the source spectrum. We propose models for the evidence and likelihood probabilities needed for the GB formulation. Using these approximated probabilities the algorithm sequentially evaluates the converted spectrum as a weighted sum of the target training vectors. It is well known that the performance of particle filtering crucially depends on successful statistical modeling of the state-space temporal evolution. The performance of GB, on the other hand, depends on dense modeling of the state-space by a set of predetermined grid-points. Nevertheless, in the following sections we show that 5-10 training sentences alone, which still result in several thousands of spectral feature vectors, are sufficient for training a GB conversion. Our subjective evaluations show that the GB conversion is found to be better or comparable, at least, to the classical GMM conversion method, when trained by this small set.

### 5.2.2 Grid-Based Approximation

The main principle of GB approximation is to provide a Bayesian sequential estimation framework while avoiding the integral computations in (5.4) and (5.6) by using a discrete evaluation of the posterior probability.

Let $\left\{\mathbf{y}_t^k\right\}_{k=1}^{N_y}$ be a set of predetermined grid-points taken from the state-space $\left\{\mathbf{y}_t\right\}$. We divide the state space into cells, so that each cell has a grid point $\mathbf{y}_t^k$ as its center. Thus, the posterior probability can be approximated by[2]:

$$p\left(\mathbf{y}_t|\mathbf{x}_{1:t}\right) \approx \sum_{k=1}^{N_y} w_{t|t}^k \delta\left(\mathbf{y}_t - \mathbf{y}_t^k\right). \tag{5.7}$$

where the posterior weights $\left\{w_{t|t}^k\right\}_{k=1}^{N_y}$ denote the conditional probabilities:

$$w_{t|t}^k = p\left(\mathbf{y}_t = \mathbf{y}_t^k|\mathbf{x}_{1:t}\right). \tag{5.8}$$

Using this discrete approximation, the prior probability is also approximated as a discrete sum:

$$p\left(\mathbf{y}_t|\mathbf{x}_{1:t-1}\right) \approx \sum_{k=1}^{N_y} w_{t|t-1}^k \delta\left(\mathbf{y}_t - \mathbf{y}_t^k\right). \tag{5.9}$$

The prior weights can be estimated sequentially [52]:

$$w_{t|t-1}^k \approx \sum_{l=1}^{N_y} w_{t-1|t-1}^l p\left(\mathbf{y}_t^k|\mathbf{y}_{t-1}^l\right), \tag{5.10}$$

[2] If the state space is indeed discrete and finite, and the grid-points consist of all its states, this evaluation becomes exact.

where $p\left(\mathbf{y}_t^k|\mathbf{y}_{t-1}^l\right)$, called the *evidence probability*, is derived from the state space dynamics (eqn. (5.1)). The posterior weights $\{w_{t|t}^k\}_{k=1}^{N_y}$ are evaluated by:

$$w_{t|t}^k \approx \frac{w_{t|t-1}^k p\left(\mathbf{x}_t|\mathbf{y}_t^k\right)}{\sum_{l=1}^{N_y} w_{t|t-1}^l p\left(\mathbf{x}_t|\mathbf{y}_t^l\right)}, \tag{5.11}$$

where, as stated above, the likelihood probability $p\left(\mathbf{x}_t|\mathbf{y}_t^k\right)$ is derived from the measurement model (eqn. (5.2)).

Finally, the hidden state vector $\mathbf{y}_t$ is approximated using the posterior weights:

$$\hat{\mathbf{y}}_t = E\left[\mathbf{y}_t|\mathbf{x}_{1:t}\right] \approx \sum_{k=1}^{N_y} w_{t|t}^k \mathbf{y}_t^k. \tag{5.12}$$

Note that equations (5.10), (5.11) and (5.12) are discrete evaluations of equations (5.4)-(5.3), correspondingly. It is known [52] that the estimated terms in (5.7) and in (5.12) are biased for any finite $N_y$. Still, as more grid points are taken the bias gets smaller and the approximation improves, since the state space is more densely represented.

The sequential estimation process is initialized using the initial probability of the state vector $p\left(\mathbf{y}_0^k\right)$, which as stated above, is assumed to be known:

$$w_{0|0}^k = p\left(\mathbf{y}_0^k\right). \tag{5.13}$$

Table 5.1 summarizes the main stages of sequential Bayesian estimation using GB approximation.

Table 5.1: Bayesian Estimation Using Grid-Based Approximation.

**Input:** a sequence of states sampled from the observed process - $\mathbf{x}_{1:T}$

**Initialization:** set the initial weights, $\{w_{0|0}^k\}_{k=1}^{N_y}$, using eqn. (5.13)

**Main Iteration:** for $t = 1, ...T$, perform the following steps:

1. Evaluate the prior weights, $\{w_{t|t-1}^k\}_{k=1}^{N_y}$, using eqn. (5.10).
2. Evaluate the posterior weights, $\{w_{t|t}^k\}_{k=1}^{N_y}$, using eqn. (5.11).
3. Evaluate the hidden state, $\hat{\mathbf{y}}_{\mathbf{t}}$, using eqn. (5.12).

**Output:** a sequence of the estimated hidden states - $\hat{\mathbf{y}}_{1:T}$

## 5.3 Voice Conversion Using Grid-Based Approximation

We now use the GB approximation method described above as a framework for spectral voice conversion. We express the conversion as a sequential estimation problem, where the observed process is the source spectrum, and the tracked state-space is the target spectrum. We propose models for both likelihood and evidence densities, required for the sequential estimation process, as described in equations (5.10)-(5.12).

The GB conversion method proposed here uses a parallel training set, but does not require time alignment between the source and target training vectors since it is trained using soft correspondence between them, rather than matched pairs. The training and conversion stages of the proposed GB conversion method are presented below in Secs. 5.3.1 and 5.3.2, respectively.

### 5.3.1 Training Stage

The training process described here includes pre-computation of the evidence and discrete likelihood probabilities. These probabilities are evaluated using all available training data. Note the difference from our previously presented GB method, where these probabilities were evaluated separately for each phoneme [53]. The source and target training sentences are assumed to be parallel and phonetically labeled. The spectral features of the two speakers are extracted from the voiced frames, but, as stated above, no time alignment is performed. Instead, a matching process of the source and target utterances is performed as follows. Each sequence of frames related to a certain phoneme at the source, is matched to its corresponding sequence at the target, according to the phonetic labeling. When matching frames extracted from recordings of the the word "father", for example, the sequence of frames related to the phoneme "f" at the source is matched to the sequence of frames related to the phoneme "f", taken from the target's recording of this word. The same is done for "a", "th" etc. Note that although matched sequences mostly have different lengths, our training process does not require using an alignment procedure such as DTW, unlike GMM-based methods do. Based on the matched sequences, we model the *discrete likelihood probability* used in eqn. (5.11), as:

$$p\left(\mathbf{x}_t = \mathbf{x}^m | \mathbf{y}_t = \mathbf{y}^k\right) \propto \begin{cases} 1 & \mathbf{x}^m, \mathbf{y}^k \text{ belong to the same phonetic sequence} \\ 0 & \text{otherwise,} \end{cases} \tag{5.14}$$

where $\{\mathbf{x}^m\}_{m=1}^{N_x}$ and $\{\mathbf{y}^k\}_{k=1}^{N_y}$ are source and target training vectors, respectively. We normalize the obtained discrete likelihood probability so that:

$$\sum_{m=1}^{N_x} p\left(\mathbf{x}_t = \mathbf{x}^m | \mathbf{y}_t = \mathbf{y}^k\right) = 1, \qquad \forall k = 1, ..., N_y. \tag{5.15}$$

The discrete likelihood probability defines a relaxed correspondence between source and target training vectors, as opposed to a one-to-one match defined in other parallel methods, for which $p\left(\mathbf{x}_t = \mathbf{x}^m | \mathbf{y}_t = \mathbf{y}^k\right) = \delta_{m,k}$.

The evidence probability, as mentioned before, expresses the transition probability from state $\mathbf{y}^l$ to state $\mathbf{y}^k$. In natural speech, spectral feature vectors related to consecutive time frames are typically similar, but not identical. Motivated by this behavior, we model the transition probability as having the same value for all the states inside a ball, centered at $\mathbf{y}^k$ with a radius $R_y$. The probability of transitions to farther states, however, is taken as a simple Gaussian distribution, centered at $\mathbf{y}^k$. Altogether, we model the *discrete evidence probability*, used in eqn. (5.10), as:

$$p\left(\mathbf{y}_t = \mathbf{y}^k | \mathbf{y}_{t-1} = \mathbf{y}^l\right) \quad = \quad \frac{e^{-\frac{M_{k,l}^2}{2}}}{\sum_{k=1}^{N_y} e^{-\frac{M_{k,l}^2}{2}}}, \tag{5.16}$$

where the exponential term in eqn. (5.16) is the maximum between the LSD of the two states $\mathbf{y}^l$ and $\mathbf{y}^k$ (as defined in eqn. 3.6, Ch. 3.1.3), normalized by a parameter $R_y$, and 1:

$$M_{k,l} = \max\left(\frac{\text{LSD}\left(\mathbf{y}^k, \mathbf{y}^l\right)}{R_y}, 1\right), \tag{5.17}$$

where $y^p\left(p\right)$ and $y^l\left(p\right)$ are the $p$-th elements of $\mathbf{y}^k$ and $\mathbf{y}^l$, respectively. An alternative approach would be to take the exponential term, defined in eqn. (5.17), as a normalized distance. For example, $M_{k,l} = \text{LSD}\left(\mathbf{y}^k, \mathbf{y}^l\right)/R_y$, where $R_y$ is a parameter selected by the user. However, in case of a sparse training set the most substantial probability would be for staying in the same state. Since the training set is fixed, the likelihood and evidence densities are in fact time invariant.

### 5.3.2 Conversion Stage

The likelihood probability modeled above in eqn. (5.14) is defined only for a discrete set consisting of the source training vector. In this section we extend (5.14) to model any input vector $\mathbf{x}_t \in \mathtt{R}^P$, as required by the GB formulation.

In our previous work dealing with GB-conversion, [53], we modeled the continuous likelihood probability $p\left(\mathbf{x}_t | \mathbf{y}_t = \mathbf{y}^k\right)$ as a sum of the discrete likelihood probabilities $p\left(\mathbf{x}^m | \mathbf{y}_t = \mathbf{y}^k\right), \quad m = 1, ..., N_x$, (defined in (5.14) and (5.15)), each weighted by a Gaussian kernel, centered at $\mathbf{x}^m$:

$$p\left(\mathbf{x}_t | \mathbf{y}_t = \mathbf{y}^k\right) \quad = \quad \frac{\sum_{m=1}^{N_x} p\left(\mathbf{x}^m | \mathbf{y}_t = \mathbf{y}^k\right) e^{-\text{LSD}^2(\mathbf{x}_t, \mathbf{x}^m)/2R_x^2}}{\sum_{k=1}^{N_y} \sum_{m=1}^{N_x} p\left(\mathbf{x}^m | \mathbf{y}_t = \mathbf{y}^k\right) e^{-\text{LSD}^2(\mathbf{x}_t, \mathbf{x}^m)/2R_x^2}}, \tag{5.18}$$

where $R_x$ is a parameter determined by the user. The Gaussian term $e^{-\text{LSD}^2(\mathbf{x}_t, \mathbf{x}^m)/2R_x^2}$ can be viewed as an interpolation factor from the discrete space represented by the source training vectors to the continuous space of the test source vectors.

Denote $\mathbf{X}_t = \left(\mathbf{x}_{t-\tau/2}, ..., \mathbf{x}_t, ..., \mathbf{x}_{t+\tau/2}\right)$ as context test vector - a sequence of test source vectors. Also denote $\{\mathbf{X}_t^m\}_{m=1}^{N_x}$ as training context vectors similarly obtained from the source training set. In a recent work, [56], (also presented Ch. 6), we have shown that Euclidian distance between context vectors leads to improved spectral matching compared with Euclidian distance between single vectors . Although that was shown for matching spectral segments of two different speakers, it is certainly beneficial for matching spectral segments taken from the same speaker. Therefore, we substitute the LSD term in the Gaussian kernel in eqn. (5.18) with the mean LSD between context vectors, i.e.:

$$\begin{aligned} p\left(\mathbf{x}_t|\mathbf{y}_t=\mathbf{y}^k\right) &= \frac{\sum_{m=1}^{N_x} p\left(\mathbf{x}^m|\mathbf{y}_t=\mathbf{y}^k\right) e^{-\overline{\mathrm{LSD}}^2(\mathbf{X}_t,\mathbf{X}_t^m)/2R_x^2}}{\sum_{k=1}^{N_y}\sum_{m=1}^{N_x} p\left(\mathbf{x}^m|\mathbf{y}_t=\mathbf{y}^k\right) e^{-\overline{\mathrm{LSD}}^2(\mathbf{X}_t,\mathbf{X}_t^m)/2R_x^2}} \\ \overline{\mathrm{LSD}}^2\left(\mathbf{X}_t,\mathbf{X}_t^m\right) &= \frac{1}{\tau}\sum_{\nu=-\tau/2}^{\tau/2} \mathrm{LSD}\left(\mathbf{x}_{t+\nu},\mathbf{x}_{t+\nu}^m\right) \end{aligned} \tag{5.19}$$

Define $w_{t|t}^k$ as the posterior weights corresponding to the training vectors $\{\mathbf{y}^k\}_{k=1}^{N_y}$:

$$w_{t|t}^k \triangleq p\left(\mathbf{y}_t|\mathbf{x}_{1:t}\right). \tag{5.20}$$

During conversion, the posterior weights are sequentially evaluated, using the corresponding evidence and likelihood probabilities defined in (5.16) and (5.19), according to equations (5.10) and (5.11). The posterior weights are used to obtain the converted outcome as a discrete Bayesian approximation (as defined in (5.12)):

$$\mathcal{F}\{\mathbf{x}_t\} = E\left[\mathbf{y}_t|\mathbf{x}_{1:t}\right] \approx \sum_{k=1}^{N_y} w_{t|t}^k \mathbf{y}_t^k. \tag{5.21}$$

Due to the sequential update of the posterior weights, the converted spectral outputs evolve smoothly in time, within each phonetic segment, also during transitions between phonemes. Figure 5.1 demonstrates the obtained time evolution of the first and third MFCCs using GB conversion, compared to the classical GMM-based conversion - JGMM [3]. The classical GMM-based conversion are applied frame by frame which may lead to discontinuities. The proposed GB, however, is based on a sequential update leading to a smoother time evolution of the cepstral elements, as seen in Fig. 5.1.

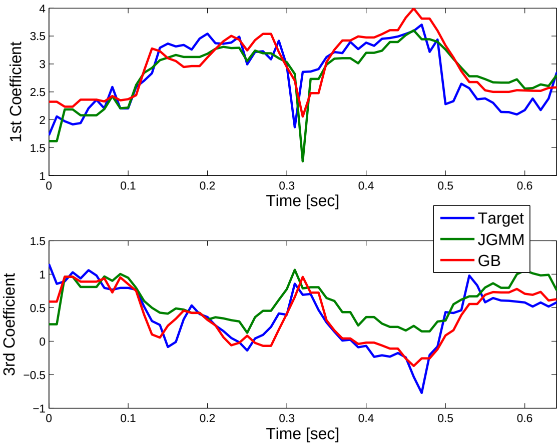


Figure 5.1: Temporal evolution of the 1st and 3rd cepstral coefficients of: the target signal - blue; JGMM - green; GB - red.

To conclude, the main stages of converting a sequence of source vectors are summarized in Table 5.2.

Table 5.2: Voice Conversion Using GB Approximation.

```
Input: a sequence of feature vectors related to the source speaker $\mathbf{x}_{1:T}$
Initialization: set the initial weights, $\{w^k_{0|0}\}^{N_y}_{k=1}$.
Main Iteration: for $t = 1, ...T$, perform the following steps:
1. Evaluate the prior weights, $\{w^k_{t|t-1}\}^{N_y}_{k=1}$, using equations (5.10) and (5.16).
2. Evaluate the posterior weights, $\{w^k_{t|t}\}^{N_y}_{k=1}$, using equations (5.11) and (5.14).
3. Evaluate $\tilde{\mathbf{y}}_t = \mathcal{F}\{\mathbf{x}_t\}$, using (5.21).
Output: a sequence of converted vectors - $\tilde{\mathbf{y}}_{1:T}$
```

## 5.4 Related Work

The GB approach uses a state-space representation of the source and target spectra to obtain a converted spectra as a weighted sum of the target training vectors. In this section we address two related methods: 1) a method based on state-space representation [54]; 2) an exemplar based approach [36], where the converted spectra is evaluated as a weighted sum of the target training vectors. We discuss here the similarities and differences between these methods and our proposed approach.

In [54], a state-space approach for representing speech spectra as an observed process generated from an underling sequence of a hidden Markov process has been proposed. The source and target speech are both modeled using this state-space representation. The state-space parameters are divided into two parts: a common part related to the uttered speech (assuming a parallel training set) and a the differentia part related to the difference between the speakers. These parts are evaluated during training time using an iterative algorithm known as Expectation Maximization (EM) [16]. During test, the common parameters related to the test utterance are evaluated using EM and then used, along with the trained differentia part to obtain the converted spectra. Both training and conversion stages include iterative training (EM). Conversion results reported by the authors were obtained using several hundreds of parallel training sentences. Although our method and Xu et al.'s method, [54], both use state-space for representing the temporal evolution of the speech sprecta, in our method the source and the target spectra are linked through a state-space dynamics, where in Xu et al.'s approach the parallel source and target spectra are each modeled as the observed signals of a shared underlined unobserved Markov process.

An exemplar-based sparse-representation approach for voice conversion has been proposed in [36]. Each speech signal is modeled as a linear combination of basis vectors (the training vectors), where the weighting matrix is called an activation matrix. The main assumption used in this method is that the speaker's identity is modeled by the basis vectors, where the information regarding the uttered text lies entirely in the activation matrix. Therefore, given a test source signal, its activation matrix is evaluated and then multiplied by the target training set, used as the target basis vectors, to obtain the converted spectra. Therefore, this method does not require any training, but its testing stage includes high computational load and a substantial memory footprint. As the exemplar-based method, our proposed GB method also uses a linear combination of the target training vectors. Besides the obvious differences in the models used by the two methods, there are two major differences:1) We use sequential evaluation of the weights to ensure smooth temporal evolution while in the exemplar based the activation matrix is evaluated as a batch. 2) We use scalar weights while the exemplar-based method uses weighting vectors (the activation matrix).

## 5.5 Experimental Results

### 5.5.1 Experiments Setup

In our experiments we used speech sentences of four U.S. English speakers taken from the CMU ARCTIC database [46]: two males (bdl, rms) and two females (clb, slt). Two different sizes of training sets 5 and 10 parallel sentences were used to demonstrate the performance of the examined methods as a function of training set size. The testing set consisted of 50 additional parallel sentences. All sentences were sampled at 16kHz and were phonetically labeled.

Analysis and synthesis were both carried out using an available vocoder [57]. This vocoder uses a two-band harmonic/noise parametrization, separated by a maximal voicing frequency for representing each spectral envelope [58]. 25 Mel Frequency Cepstrum Coefficients (MFCCs) were extracted from the harmonic parameters [21]: the zero-th coefficients, related to the energy, were not converted. The other 24 coefficients were used as spectral feature vectors during training and conversion.

The spectral features of unvoiced frames were not converted but simply copied to the converted sentence, since they do not capture much of the speaker's individuality [59] and their conversion often leads to quality degradation [60]. The maximal voicing frequency was also not converted but re-estimated from the converted parameters by the vocoder. The sequences of the training data set used for GB conversion were matched (without alignment), as described in Sec. 5.3.1. The training set used for the other examined methods, and the testing set, were each time aligned using a DTW algorithm based on phonetic labeling [48]. The pitch was converted linearly as described in eqn. (3.1).

### 5.5.2 Objective Evaluations

The examined GMM-based methods (JGMM and CGMM) were trained using diagonal covariance matrices and $1-4$ Gaussian mixtures, due to the low amount of training data.

We begin with a short examination of the influence of each of the three parameters of the proposed GB method ($R_x$, $R_y$ and $\tau$) on its performance. Figure 5.2 presents the ND vs. NGV values obtained for the proposed GB method using $R_x \in [0.3, 2]$, $R_y \in [1, 4]$ and $\tau = 1$, trained by 10 sentences, for a male-to-male conversion. As the parameter $R_x$ gets higher, more grid-points are considered in the weighted sum, so that ND decreases, but the NGV also decreases. Since the evidence probability is solely determined by the training set (see eqn. (5.16)), we also examined the performance of the GB method using a data-driven value for $R_y$, specifically, the median of the MCD between all training vectors pairs related to the target speaker. These values vary between 2-3dB when using different source-target pairs and data-set sizes. As depicted in Fig. 5.2, the median leads to the best ND-NGV values so all results presented from now on were obtained using this value for $R_y$. Figure 5.3 presents the ND vs. NGV values obtained for the proposed GB method using $R_x \in [0.3, 2]$, $\tau = (0, 1, 2)$, trained by 10 sentences, for a male-to-male conversion. Using $\tau = 1$ leads to higher NGV values than using $\tau = 0$, with a slight increase in the ND. However, increasing $\tau$ further leads to the same NGV values with a minor decrease in the ND. Table 5.3 summarizes the ND and NGV values achieved by JGMM [3] and the proposed GB conversion method, for all four gender conversions: male-to-male (M2M), male-to-female (M2F), female-to-male (F2M) and female-to-female (F2F), using 5 and 10 training sentences. The number of mixtures for JGMM, and parameters for the GB ($R_x$ and $\tau$) were selected for each method and training set so that a minimal ND was attained, while keeping the NGV as high as possible. As mentioned above, $R_y$ was taken as the median. The proposed GB leads to higher NGV values in all the cases. For 5 training sentences JGMM leads to lower ND values (except for F2M), however, using 10 training sentences, the proposed GB achieves lower or very similar ND values. Still, both methods lead to very low NGV values and consequently, muffled sounding synthesized signals.

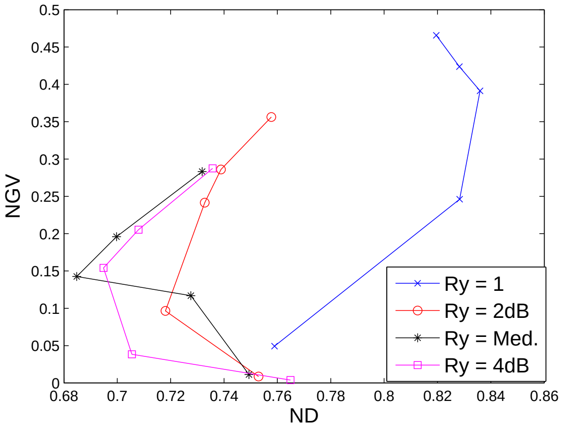


Figure 5.2: ND vs. NGV for GB conversion for a male-to-male conversion using 10 training sentences and $R_x \in [0.3, 2]\, dB$, $\tau = 1$ and: $R_y = 1dB$ - blue x; $R_y = 2dB$ - red circle; $R_y = median$ - black astrict; $R_y = 3dB$ - magenta square.

To further improve the quality of the synthesized speech, we applied the post-processing method for GV enhancement [44]. This method maximizes the GV of an input sequence, under a spectral distortion constraint. The GV of each enhanced sequence is increased up to the level where the MCD between the converted sequence and its enhanced version reaches a preset threshold value, denoted as $\theta_{MCD}$. We

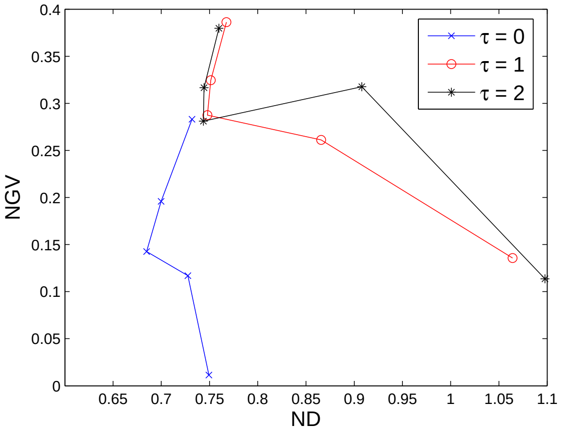


Figure 5.3: ND vs. NGV for GB conversion for a male-to-male conversion using 10 training sentences and $R_x \in [0.3, 2]$, $R_y$ = median: $\tau = 0$ - blue x; $\tau = 1$ - red circle; $\tau = 2$ - black astrict.

Table 5.3: Objective performance: ND and NGV values using 5 and 10 training sentences, for all four gender conversions.

| | | 5 Train. Sent | | 10 Train. Sent | |
|---|---|---|---|---|---|
| | | ND | NGV | ND | NGV |
| M2M | JGMM | **0.72** | 0.15 | 0.71 | 0.13 |
| | GB | 0.73 | **0.25** | **0.69** | **0.14** |
| M2F | JGMM | **0.7** | 0.15 | 0.7 | 0.12 |
| | GB | 0.71 | **0.21** | **0.69** | **0.19** |
| F2M | JGMM | 0.74 | 0.14 | 0.71 | 0.13 |
| | GB | **0.71** | **0.34** | 0.71 | **0.42** |
| F2F | JGMM | **0.8** | 0.22 | **0.8** | 0.18 |
| | GB | 0.88 | **0.34** | 0.81 | **0.31** |

recently showed [44] that this method leads to significant improvement in the perceived quality of signals converted by the classical GMM method [2]. In this work we applied this GV enhancement method to JGMM [3], and to our proposed GB conversion outcomes. We also examined the performance of CGMM, which considers GV enhancement at training.

Table 5.4 summarize the ND and NGV values achieved by the examined conversion methods, for all four gender conversions using 5 and 10 training sentences. Again, the GB conversion, followed by GV enhancement with $\theta_{MCD}$ = 2dB (En-GB) leads to the highest NGV values. Using 5 training sentences, JGMM leads to the lowest ND values, while En-GB comes in second (except for F2F). Using 10 training sentence, En-GB, produces the lowest ND and at the same time the highest NGV, for M2M and M2F conversion. For F2M and F2F conversion, En-GB leads to the highest NGV with very similar ND values to JGMM, which are the lowest.

Table 5.4: Objective performance: ND and NGV values using 5 and 10 training sentences, for all four gender conversions with GV enhancement ($\theta = 2dB$).

| | | 5 Train. Sent | | 10 Train. Sent | |
|---|---|---|---|---|---|
| | | ND | NGV | ND | NGV |
| M2M | JGMM | **0.76** | 0.6 | 0.74 | 0.55 |
| | CGMM | 0.83 | 0.46 | 0.82 | 0.45 |
| | GB | 0.79 | **0.8** | **0.73** | **0.6** |
| M2F | JGMM | **0.74** | 0.57 | 0.74 | 0.54 |
| | CGMM | 0.83 | 0.45 | 0.84 | 0.46 |
| | GB | 0.76 | **0.73** | **0.73** | **0.68** |
| F2M | JGMM | 0.77 | 0.63 | **0.75** | 0.69 |
| | CGMM | 0.86 | 0.62 | 0.85 | 0.61 |
| | GB | **0.76** | **0.95** | 0.77 | **1.1** |
| F2F | JGMM | **0.86** | 0.79 | **0.85** | 0.65 |
| | CGMM | 0.91 | 0.63 | 0.89 | 0.6 |
| | GB | 0.95 | **1** | 0.87 | **0.98** |

To conclude the objective examination, in terms of NGV, the proposed EN-GB conversion scheme outperforms all the examined methods. In terms of ND, JGMM leads to lower ND values using 5 training sentences. Using 10 training, En-GB leads to the lowest (or very similar to the lowest) ND values.

In the next section we present subjective evaluation results comparing the proposed En-GB conversion scheme to the classical GMM-based conversion method (with enhancement) and to CGMM, in terms of perceived quality and similarity to the target speaker.

### 5.5.3 Subjective Evaluations

Listening tests were carried out to subjectively assess the performance of the examined methods (all trained by 10 sentences). In every test, 10 different sentences were examined by 11 listeners. The group of listeners included 20-30 years old, non-experts, men and women. The same four speakers (two males and two females) that were used for the objective evaluations, were used for the subjective evaluations.

The number of mixtures for the GMM-based methods and parameters for the GB conversion ($R_x$ and $\tau$) were set so minimal spectral distortion would be attained while keeping the NGV as high as possible. We used informal listening tests to select the threshold value for GV enhancement from $\theta_{MCD} = 0.5, 1, 2, 4$dB. The best perceived quality was obtained with $\theta_{MCD} = 2$dB, for both JGMM and GB. All four gender conversions were performed using the same parameters values as described above.

We conducted subjective quality evaluations in a format similar to Multi Stimulus test with Hidden Reference and Anchor (MUSHRA) [30]. The listeners were presented with four test signals: (a) a hidden reference - the target speaker; (b) Enhanced JGMM; (c) CGMM; (d) Enhanced GB (En-GB). The test signals were randomly ordered, and the listeners were not informed about the hidden reference signals being included in the test set. During evaluation, the listeners were asked to compare the test signals to the reference signal (the target speaker) and rate their quality between 0 to 100, where at least one of the test signals (the hidden reference) must be rated 100. As expected, all the listeners rated the hidden reference as 100. The mean scores of the examined methods for M2M, M2F, F2M and F2F conversions, and also their scores averaged over all four conversions are presented in Figures 5.4 and 5.5, respectively. All subjective results are presented with their 95% confidence intervals. We evaluated the individuality performance using, again, a similar format to MUSHRA, as conducted by Godony et. al. [31]. The listeners were presented with the same test signals (including the hidden reference) and were asked to rate their similarity to the reference signal, in terms of the speaker's identity, while ignoring their perceived quality. The mean individuality scores of the examined methods for M2M, M2F, F2M and F2F conversions, and also their scores, averaged over all four conversions, are presented in Figures 5.6 and 5.7, respectively.

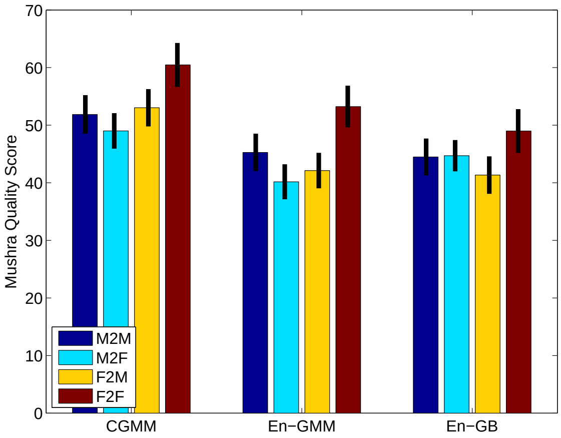


Figure 5.4: Subjective quality test, comparing: Enhanced JGMM (En-GMM), CGMM [43] and Enhanced GB (En-GB).

Except for F2F, the proposed EN-GB was rated as most similar to the target speaker (Fig. 5.6). In terms of perceived quality, CGMM was rated as having the best quality, while EN-JGMM and EN-GB were rated as comparable (Fig. 5.4). All in all, considering all four gender conversion, the proposed

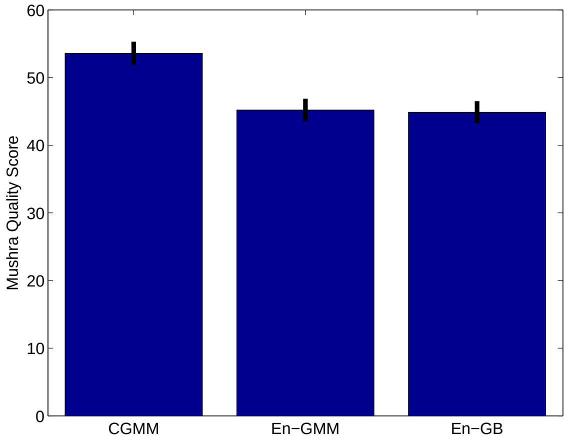


Figure 5.5: Subjective quality test averaged over all four gender conversions comparing: Enhanced JGMM (En-GMM), CGMM [43] and Enhanced GB (En-GB).

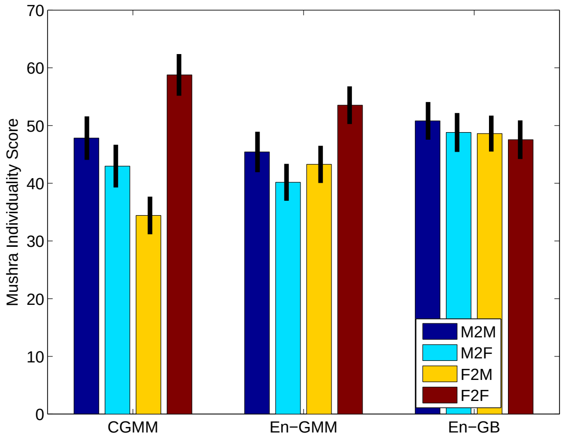


Figure 5.6: Subjective individuality test, comparing: Enhanced JGMM (En-GMM), CGMM [43] and Enhanced GB (En-GB).

EN-GB was marked as most similar to the to the target speaker, while CGMM was marked as having best quality.

## 5.6 Chapter Summary

We propose here a GB voice conversion method suitable for low resource environments, which can be successfully trained using very few sentences (5-10) and does not require phonetic labeling of the test signals.

The GB conversion method is based on sequential Bayesian tracking, using a Grid-Based (GB) formulation. The target spectral evolution is modeled as a hidden Markov process, tracked by using the source spectrum, modeled as the observed process. The training stage is very simple and based on

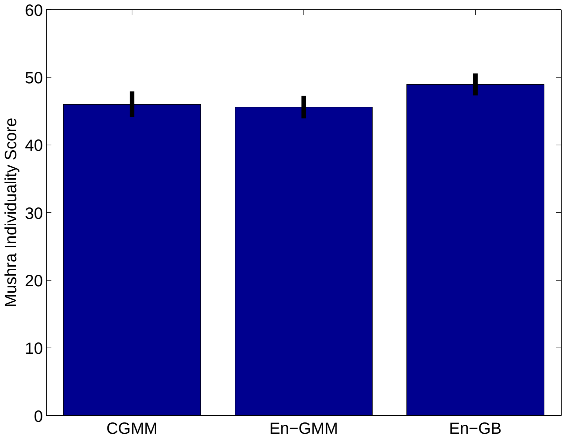


Figure 5.7: Subjective individuality test averaged over all four gender conversions comparing: Enhanced JGMM (En-GMM), CGMM [43] and Enhanced GB (En-GB).

Euclidean distances between the training vectors and it is successfully performed using very small training sets. Additionally, although GB is trained using a parallel set, time alignment is not needed. During training, the evidence and likelihood probabilities needed for the GB formulation are approximated as discrete densities. During conversion, the converted spectrum is obtained as a weighted sum of the training target vectors, used as grid-points. The weights are sequentially evaluated so that a smooth temporal evolution of the converted spectra is produced.

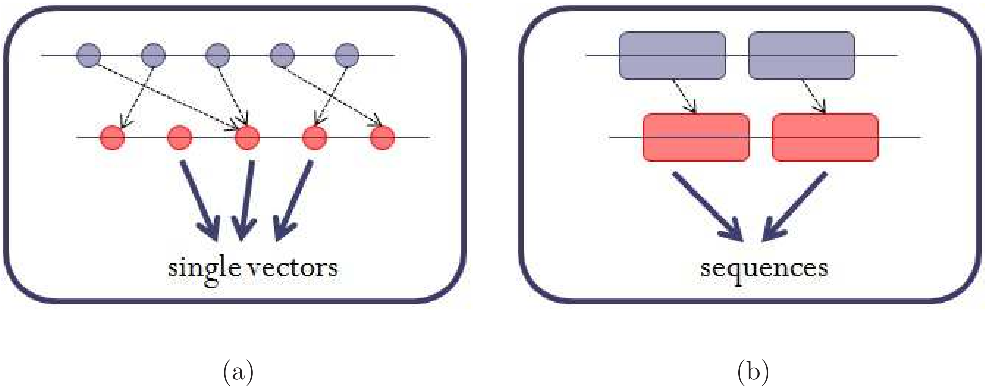
We used a small set of just 10 sentences for training both the classical GMM-based conversion function and our GB method. According to our experiments, the GB conversion method achieves lower spectral distances between the converted and target spectra and GV values which are closer to the target speaker's values, than the classical GMM-based conversion. To further improve the quality of the synthesized speech, we increased the variability of the converted vectors by applying GV enhancement as a post-processing block. We compared the proposed Enhanced GB (En-GB) scheme to CGMM and to classical GMM-based conversions, with GV enhancement, using listening tests. This comparison showed that En-GB is best in terms of similarity to the target speaker and comparable to the enhanced GMM conversion, in terms of quality.

# Chapter 6

# Non-Parallel Conversion

In this chapter we formulate the non-parallel training process as a minimization problem of a joint cost, considering temporal-context alignment and conversion function. We propose a generalization of INCA (described in Sec. 3.2.3), denoted here Temporal-Context INCA (TC-INCA), based on matching sequences of vectors (rather than single vectors), according to their original temporal context. We show that TC-INCA (and hence also INCA) are, in fact, alternating minimization steps of the joint cost, and prove their convergence.

Fig. 6.1 illustrates the main difference between TC-INCA, which is based on matching temporal context vectors, and INCA's, which is based on matching single vectors.



(a) (b)

Figure 6.1: Alignment process: (a) - matching feature vectors used in INCA; (b) - temporal context vectors (sequences of feature vectors) used in TC-INCA.

We present objective and subjective evaluations comparing the proposed TC-INCA to INCA. Our method significantly increases the amount of correctly matched pairs and leads to improved synthesized quality and similarity to the target.

## 6.1 TC-INCA

### 6.1.1 Joint Cost

In this section we formulate the training stage of a non-parallel conversion as a minimization problem of a joint cost, considering both conversion and context-based matching functions. We define a set of context vectors $\{\mathbf{X}_k\}_{k=1}^{\tilde{N}_x} \in \mathbb{R}^{d(T+1)}$, $\{\mathbf{Y}_k\}_{k=1}^{\tilde{N}_y} \in \mathbb{R}^{d(T+1)}$ obtained by concatenating $T/2$ ($T$ is even) successive vectors before and after each training vector:

$$\begin{aligned}\mathbf{X}_k &\triangleq \left(\mathbf{x}_{k-T/2}^\top, ..., \mathbf{x}_k^\top, ...\mathbf{x}_{k+T/2}^\top\right)^\top \\ \mathbf{Y}_j &\triangleq \left(\mathbf{y}_{j-T/2}^\top, ..., \mathbf{y}_j^\top, ...\mathbf{y}_{j+T/2}^\top\right)^\top,\end{aligned} \tag{6.1}$$

where $\tilde{N}_x = N_x - T, \tilde{N}_y = N_y - T$ are the number of the source and target context vectors, respectively. We assume that the non-parallel source and target sets are extracted from several continuous utterances (words, sentences). To simplify the notation we also assume that the indices $k$ and $j$ reflect their temporal ordering, meaning that $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$, for example, are extracted from consecutive time frames.

Given a spectral conversion function, its inverse, $\mathcal{F}^{-1}(\cdot)$, and two matching functions $p(\cdot)$ and $q(\cdot)$ - pairing each source context vector to a target context vector and vice versa, we write a joint cost function, similar to eqn. (3.31):

$$\mathcal{L} = \sum_{k=1}^{\tilde{N}_x} \left\|\mathcal{F}(\mathbf{X}_k) - \mathbf{Y}_{p(k)}\right\|^2 + \sum_{j=1}^{\tilde{N}_y} \left\|\mathbf{X}_{q(j)} - \mathcal{F}^{-1}(\mathbf{Y}_j)\right\|^2, \tag{6.2}$$

where the converted context vectors $\mathcal{F}(\mathbf{X}_k)$ are obtained by applying the conversion function on each feature vector:

$$\mathcal{F}(\mathbf{X}_k) \triangleq \left(\mathcal{F}\left(\mathbf{x}_{k-T/2}\right)^\top, ..., \mathcal{F}(\mathbf{x}_k)^\top, ...\mathcal{F}\left(\mathbf{x}_{k+T/2}\right)^\top\right)^\top, \tag{6.3}$$

and similarly for $\mathcal{F}^{-1}(\mathbf{Y}_j)$.

The cost presented in eqn. (6.2) is the empirical squared-error between the source and target sequences and their estimated versions (using the conversion function), according to the two alignment functions, $p$ and $q$. Therefore, we regard the training stage as an optimization problem, aiming to minimize this cost:

$$\{\mathcal{F}^*, p^*, q^*\} = \underset{\{\mathcal{F},p,q\}}{\operatorname{argmin}} \mathcal{L}(p, q, \mathcal{F}). \tag{6.4}$$

In the parallel case, alignment is obtained by using DTW and phonetic labeling (if available). Assuming, w.l.o.g., that the source and target training vectors are ordered so that $\mathbf{x}_k$ matches $\mathbf{y}_k$, $\forall k = 1, ..., N$, the matching functions become identity functions: $p(k) = q(k) = k$. Substituting eqn. (6.3) in eqn. (6.2) and neglecting the ends, our cost becomes:

$$\mathcal{L}_{para} = T\left(\sum_{k=1}^{N} \left\|\mathcal{F}(\mathbf{x}_k) - \mathbf{y}_k\right\|^2 + \sum_{j=1}^{N} \left\|\mathbf{x}_j - \mathcal{F}^{-1}(\mathbf{y}_j)\right\|^2\right), \tag{6.5}$$

which is a symmetric generalization of the empirical loss minimized in the training process of the classical GMM-based conversion (see Sec. 3.2.2, eqn. (3.22)), up to a constant $T$.

### 6.1.2 Iterative Minimization

In this section we present an iterative approach, for reducing the joint cost defined in eqn. (6.4), similar to the iterative process of INCA [1]. Applying standard minimization techniques such as gradient descent is rather problematic considering the non trivial dependency of the joint cost with respect to the matching functions. Alternating minimization is a well known iterative technique for minimizing cost functions depending on more than one variables [61]. Applying this method for minimizing the joint cost, reduces eqn. (6.4) to two minimization problems solved iteratively for $t = 1, 2, ...$:

$$\begin{aligned} \{p_t, q_t\} &= \underset{\{p,q\}}{\operatorname{argmin}} \mathcal{L}\left(p, q, \mathcal{F}_{t-1}\right) && (6.6) \\ \mathcal{F}_t &= \underset{\mathcal{F}}{\operatorname{argmin}} \mathcal{L}\left(p_t, q_t, \mathcal{F}\right), && (6.7) \end{aligned}$$

**Lemma 6.1.** *The series $\mathcal{L}\left(p_t, q_t, \mathcal{F}_t\right)$ converges to a (local) minimum.*

*Proof.* According to eqns. (6.6) and (6.7), the solutions $\{\mathcal{F}_t, p_t, q_t\}$ sustain:

$$\begin{aligned} \mathcal{L}_t &\triangleq \mathcal{L}\left(p_t, q_t, \mathcal{F}_t\right) \leq \mathcal{L}\left(p_t, q_t, \mathcal{F}_{t-1}\right) \\ &\leq \mathcal{L}\left(p_{t-1}, q_{t-1}, \mathcal{F}_{t-1}\right) \triangleq \mathcal{L}_{t-1}. && (6.8) \end{aligned}$$

The series $\{\mathcal{L}_t\}$ is non-increasing and obviously bounded by zero, therefore converges to a (local) minimum. □

Convergence to a global minimum, or even existence of a single minimum is not guarantied since the original minimization problem stated in eqn. (6.2) is not convex.

Given a conversion function, the joint cost is separable in $p$ and $q$, leading to a two-step solution of eqn. (6.6):

$$\begin{aligned} p_t &= \underset{p}{\operatorname{argmin}} \sum_{k=1}^{\tilde{N}_x} \left\|\mathcal{F}_{t-1}\left(\mathbf{X}_k\right) - \mathbf{Y}_{p(k)}\right\|^2 \\ q_t &= \underset{q}{\operatorname{argmin}} \sum_{j=1}^{\tilde{N}_y} \left\|\mathbf{X}_{q(j)} - \mathcal{F}_{t-1}^{-1}\left(\mathbf{Y}_j\right)\right\|^2 && (6.9) \end{aligned}$$

We apply a nearest-neighbor search, similar to the one applied for INCA, but instead of using single spectral feature vectors, we use the context vectors defined in eqn. (6.1):

$$\begin{aligned} p_t\left(k\right) &= \underset{j}{\operatorname{argmin}} \left\|\mathcal{F}_{t-1}\left(\mathbf{X}_k\right) - \mathbf{Y}_j\right\|^2 \\ q_t\left(j\right) &= \underset{k}{\operatorname{argmin}} \left\|\mathbf{X}_k - \mathcal{F}_{t-1}^{-1}\left(\mathbf{Y}_j\right)\right\|^2. && (6.10) \end{aligned}$$

According to our preliminary experiments, an optimal exhaustive search for the exact solutions of (6.9) yields a negligible improvement compared to a nearest-neighbor search.

Substituting eqn. (6.3) into eqn. (6.7) and neglecting the ends, the minimized term takes a similar form to the parallel symmetrical cost presented in eqn. (6.5):

$$\begin{aligned}\mathcal{F}_t \;=\; \underset{\mathcal{F}}{\operatorname{argmin}} & \left\{ \sum_{k=1}^{\tilde{N}_x} \left\| \mathcal{F}\left(\mathbf{x}_k\right) - \mathbf{y}_{p_t(k)} \right\|^2 + \right. \\ & \left. + \sum_{j=1}^{\tilde{N}_y} \left\| \mathbf{x}_{q_t(j)} - \mathcal{F}^{-1}\left(\mathbf{y}_j\right) \right\|^2 \right\}. \end{aligned} \tag{6.11}$$

Consequently, any parallel conversion method minimizing this squared error can be used as an auxiliary function, using the parallelized training set - $\left\{\left(\mathbf{x}_k, \mathbf{y}_{p_t(k)}\right), \left(\mathbf{x}_{q_t(j)}, \mathbf{y}_j\right)\right\}$. The classical GMM-based conversion, for example, can fit this description since its parameters are evaluated using Least Squares minimization of the MSE between the converted and target vectors as described in Sec. 3.2.2.

The TC-INCA algorithm, is summarized in Table 6.1. We note that if no context frames are considered, meaning $T = 0$, TC-INCA essentially becomes identical to INCA.

Table 6.1: Joint Cost Optimization Using TC-INCA.

| |
|---|
| **Input:** a non-parallel training set of context vectors $\{X, Y\}$<br>**Initialization:** set the initial conversion function to identity: $\mathcal{F}_0(\mathbf{X}) = \mathbf{X}$<br>**Main Iteration:** for $t = 1, 2...$ perform the following steps:<br>1. Evaluate the matching functions, $p_t, q_t$, using eqn. (6.10).<br>2. Train an auxiliary conversion function using eqn. (6.11).<br>3. Evaluate the cost function $\mathcal{L}(p_t, q_t, \mathcal{F}_t)$ using eqn. (6.2) and check convergence.<br>**Output:** conversion and matching functions $p_t, q_t, \mathcal{F}_t$. |

## 6.2 Experimental Results

### 6.2.1 Experiment Setup

Three U.S. English speakers (two females and one male) taken from the CMU ARCTIC database [46] were used for our objective and subjective evaluations in two directions - female to female (F2F) and female to male (F2M). Analysis, synthesis and extraction of 24 MFCCs were performed using an available toolkit [57], based on the Harmonic Plus Noise Model (HNM) [62, 63].

We used both parallel and non-parallel sets for training, consisting of $(5, 10, 50, 100)$ sentences, and an additional set of 50 parallel sentences for testing, all sampled at 16kHz. The pitch was converted using a simple linear function using the mean and the standard deviation values of the source and target speakers.

### 6.2.2 Objective Evaluations

We used two objective criteria to evaluate the performance of the trained matching and conversion functions: phonetic accuracy, measured by the percentage of training vectors having the same phonetic label as their matches (as suggested by [60]), and Normalized Distance (ND), as defined in eqn. (3.7)

We used the classical GMM method for training the auxiliary and final conversion functions using full covariance matrices and $(1, 2, 3, 4)$ mixtures for both methods. TC-INCA was trained using several context lengths, $T = (2, 4, 8, 10, 14, 18, 24, 26)$. The number of mixtures (for both methods) and context length (for TC-INCA) were tuned for F2F and M2F and for every training set size, so that maximal (training) accuracy would be attained. Generally, the best accuracy was obtained using longer context $T \in [14, 24]$ for the parallel sets, than for the non-parallel sets $T \in [2, 10]$. Also, as more training sentences were used, more mixtures were preferred.

Figs. 6.2 and 6.3 present the accuracy values attained by TC-INCA compared to INCA, averaged over both examined directions (F2F and M2F) using parallel and non parallel sets, respectively. TC-INCA leads to significantly higher phonetic accuracy, using either parallel or non-parallel training sets. The ND values achieved by both methods are very similar ($\pm 1\%$), in the range of 0.7-0.75 for the parallel sets and 0.75-0.8 for the non-parallel sets. Nevertheless, the improvement in accuracy has a great influence on the perceived quality and similarity to the target, as presented in the next section.

### 6.2.3 Subjective Evaluations

We carried out two preference tests comparing TC-INCA to INCA: AB quality tests and ABX individuality tests, as described in Sec. 3.1.4 Following Helander et al. [64], we allowed the listeners to answer "equal", if they felt they could not decide between the two options. In each test (quality and individu-

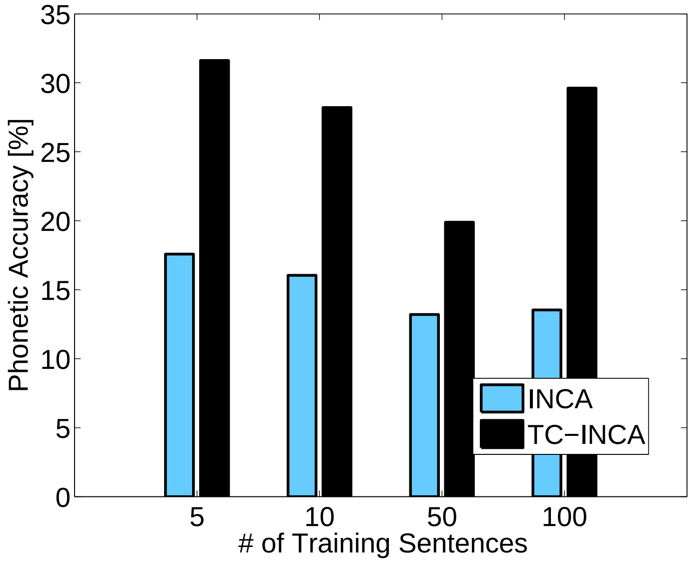


Figure 6.2: Maximal accuracy [%] (39 phonemes) vs. training set size obtained by TC-INCA and INCA using parallel training sets.

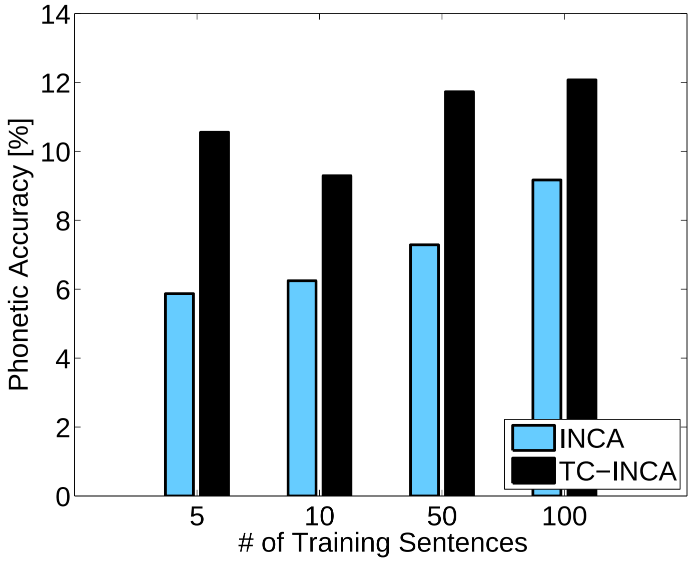


Figure 6.3: Maximal accuracy [%] (39 phonemes) vs. training set size obtained by TC-INCA and INCA using Non-parallel training sets.

ality), 10 different randomly ordered (pairs or triplets, correspondingly) were examined by 10 listeners, all 20-30 years old non-experts. For these evaluations we used non-parallel training sets consisting of 5 sentences. Table 6.2 presents the overall results, averaged over both F2F and F2M conversions. The advantage of TC-INCA is well demonstrated; most listeners marked it as having a higher quality and as more similar to the target speaker, than INCA.

Table 6.2: Subjective Preference Evaluations.

| | INCA [%] | TC-INCA [%] | Equal [%] |
|---|---|---|---|
| Quality | $20 \pm 2$ | $\mathbf{73 \pm 2}$ | $7 \pm 1$ |
| Individuality | $33 \pm 2$ | $\mathbf{54 \pm 2}$ | $13 \pm 1$ |

## 6.3 Discussion

In this chapter we presented a non-parallel training process as a minimization problem of a joint cost, considering both temporal-context alignment and conversion functions. We proposed TC-INCA (a generalization of INCA) for iteratively performing this minimization. We showed that TC-INCA reduces the joint cost (and therefore INCA too) and prove its convergence. Objectively, TC-INCA leads to a considerable increase of alignment accuracy and to similar spectral distance values, compared to INCA. Subjective evaluations demonstrate the great influence of accuracy improvement: TC-INCA was rated higher, both in terms of quality and similarity to the target speaker.

# Chapter 7

# Keyword Spotting

## 7.1 Background

Keyword Spotting (KWS) is a task of detecting whether a keyword was said in a given speech signal. It is used, for example, in mobile applications, smart homes and security purposes. If the query is given in the form of text, KWS can be viewed as a sub-task of automatic speech recognition (ASR). Some ASR systems aim to recognize whole word terms, so they use Large Vocabulary Continuous Speech Recognition (LVCSR) to generate word level transcription of the given speech signal [65]. These systems require an enormous amount of annotated data and detailed language models, which are not always available for under-documented languages or speech of children [66], for example. Many KWS systems addressing this task are based on phonetic recognizers used for ASR, thus eliminating the need for a detailed word-based language model. Such systems use Hidden Markov Models (HMMs) to statistically model sub-word units such as phonetic n-grams or multigrams [67–70]. Still, these systems require a great amount of phonetically labeled recordings.

When the query is given as a speech signal, Query-by-Example (QBE) approaches are applied. These methods usually do not use language models so they require much smaller training sets and considerably less annotated data, if any. Some QBE approaches are based on lattice representation of sub-word units, similarly to text-based systems. These supervised methods train the lattices using phonetically labelled recordings [8, 9]. Unsupervised QBE methods do not require any kind of labelled resource; they use a template representation of the searched keyword and compare it against a similar representation of a given speech utterance. Several methods based on a posterior representation of speech data have been proposed using: a phonetic division where the posterior values are obtained using the lattice output of a phonetic recognizer [8], the output of a Multi Layer perceptron (MLP) [10], statistical modeling of the speech signal using Gaussian Mixture Model (GMM) [11], or alternatively, using HMM [12]. The natural rate of speech varies with speakers and context so the posterior representation of the template and test signals do not match in length. Therefore most of these methods use Dynamic Time Warping (DTW).

An efficient implementation for DTW have been proposed [71], still using DTW impose a challenging computational load.

The main criticism against KWS methods presented above is that they use statistical models or phonetic segmentation for classification, so they are not directly optimized for minimizing keyword detection rate. In recent years, several keyword spotting methods have been proposed based on a discriminative classification. Discriminative methods use machine learning techniques for training optimal (in terms of detection rate) binary classifiers between speech signals including keywords and not including it. Keshet et al. proposed new feature representation for speech utterances based on the estimated duration of phonemes and transition times [13]. A linear classifier is trained using positive sentences (including the keyword) and negative sentences (not including it). This method is trained using phonetic segmented data at a medium size such as TIMIT, which consists of about 4 hours of recorded speech.

In some cases, when dealing with under-documented speech, such as children's voice or under documented languages, even a medium size data-set is unavailable. In some other limited-data applications, such as in a mobile environment, only few positive examples may be available for training and computational load is also limited. Two methods dealing with small training sets have been proposed. These methods use features extracted from the time-frequency representation of speech signals: sprectro-temporal patches [14] or patterns of high-energy tracks [15]. Both methods use isolated utterances of the keyword (as opposed to using positive sentences - including the keyword - as used by Keshet et al. [13]) and negative utterances including other words to train a binary classifier. Given a test sentence, a sequence of feature vectors is extracted using a sliding window. A binary classifier fed with this sequence produces a response curve, and a final decision regarding the existence of the keyword is taken by applying a threshold to the response vector.

In this chapter we present a novel discriminative method for unsupervised keyword spotting in a limited-data environment. Our method is based on two classifiers: an isolated word classifier, and a sentence classifier. We propose here a new representation for isolated words and for sentences, presented in Ch. 7.2.1 and Ch. 7.2.2, respectively.

In this work we specifically deal with limited-data setups such as mobile applications, where users are not willing to record themselves more then a few times, resulting in a very small positive data-set available for training. In such a setup where the positive training set is much smaller than the negative one, the training process may result in a classifier which is biased towards the negative class. To avoid this situation, while still exploiting the diversity of the negative training set, we use bootstrap aggregating, also referred to as bagging predictors [17], for training the isolated word classifier, as well as for training the global classifier for sentences, as described in 7.2.3. In Ch. 7.3 we present experimental results demonstrating the advantages of our approach compared to a HMM-based KWS benchmark system.

## 7.2 Proposed Approach

In this section we propose a new discriminative method for keyword spotting. This method is based on a histogram representation for classification of isolated words as described in Sec. 7.2.1, followed by global features representation for classification of sentences, as described in Sec. 7.2.2. In Sec. 7.2.3 we describe how bagging predictors are utilized for training robust and unbiased word and sentence classifiers. An overall description of our proposed inference procedure, based on the above, is presented in Sec. 7.2.4.

### 7.2.1 Histogram Representation For Isolated Words

Let $\mathcal{M}$ be a Gaussian Mixture Model (GMM), trained using spectral features extracted from all available training data:

$$\mathcal{M} = \{\lambda^m, \mu^m, \mathbf{\Sigma}^m; m = 1, ..., M\}, \tag{7.1}$$

where $\lambda^m \in R, \mu^m \in R^P$ and $\mathbf{\Sigma}^m \in R^{P\times P}$ are the weight, mean vector and covariance matrix of the m-th component (out of $M$ components in the mixture), respectively, and $P$ is the dimension of the spectral feature vectors. GMM is an unsupervised model, not requiring any labelling or other metadata, so even in cases of limited data resources such as under-documented languages, a sufficiently large amount of training data can be easily collected.

Given a sequence of $T_w$ spectral feature vectors extracted from a specific utterance of a single word - $(\mathbf{x}_1, ..., \mathbf{x}_{T_w}) \in R^{P\times T_w}$, we obtain its posteriograms, $\mathbf{z}_{1:T_w} = (\mathbf{z}_1, ..., \mathbf{z}_{T_w}) \in R^{M\times T_w}$, with respect to the GMM:

$$z_t\,(m) \quad = \quad \frac{\lambda^m \exp\{-1/2\,(\mathbf{x}_t - \mu^m)^\top \Sigma^{m-1}\,(\mathbf{x}_t - \mu^m)\}}{\sum_{n=1}^{M} \lambda^n \exp\{-1/2\,(\mathbf{x}_t - \mu^n)^\top \Sigma^{n-1}\,(\mathbf{x}_t - \mu^n)\}} \qquad \begin{array}{rcl} t &=& 1, ..., T_w \\ m &=& 1, ..., M \end{array} \tag{7.2}$$

where $z_t\,(m)$ is the m-th element of $\mathbf{z}_t$. For each vector $\mathbf{z}_t, t = 1, ..., T_w$, we set the maximal element to 1 and the rest to zero to obtain an indicator vector $\mathbf{u}_t \in R^M$ such that:

$$u_t\,(m) = \begin{cases} 1 & m = \underset{n=1,..,M}{\text{argmax}}\; z_t\,(n) \\ 0 & otherwise \end{cases} \tag{7.3}$$

This means that $\mathbf{u}_t$ is an $M \times 1$ indicator of the specific Gaussian component in $\mathcal{M}$ that has the highest probability, for a given $\mathbf{x}_t$. We obtain the word histogram representation, $\mathbf{v} \in R^M$, by averaging the indicator vectors, $\mathbf{u}_{1:T_w}$, over $t$:

$$\mathbf{v} = \frac{1}{T_w}\sum_{t=1}^{T_w} \mathbf{u}_t, \tag{7.4}$$

Therefore each element of $\mathbf{v}$ counts the fraction of times a certain Gaussian component led to the highest probability. Note that regardless of the value of $T_w$, the proposed histogram representation always results in an $M$-dimensional vector (depending on the size of the mixture), thus enabling training of discriminative classification methods with fixed input size such as Support Vector Machine (SVM).

Given a positive set of histograms extracted from utterances of the keyword and a negative set extracted from utterances of non keywords, we train a binary classifier for isolated words. In the following section we use this classifier to obtain a response curve of a given sentence, which is further used for extracting a global feature vector representing the entire sentence.

### 7.2.2 Global Feature Representation Of Sentences

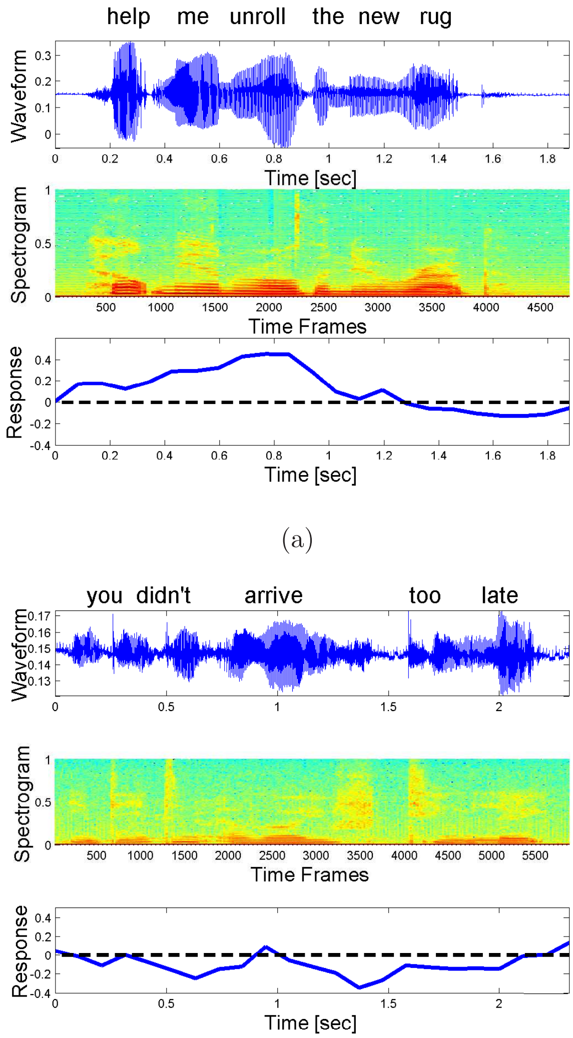
Given a sequence of spectral features related to a certain sentence, $(\mathbf{x}_1, ..., \mathbf{x}_{T_s})$, (positive or negative) we apply a sliding window of length $\alpha\bar{T}_w$, with a $\beta\bar{T}_w$ hop, where $\bar{T}_w$ is the mean length of the keyword, evaluated using the keyword utterances used for training the word classifier, and $\alpha > 1$ and $\beta < 1$. This way, in case of a positive sentence, at least most the of spectral feature vectors related to the keyword would fit into one of the window hops. For each window, we extract a histogram with respect to the GMM, $\mathcal{M}$. The sequence of histograms,$\mathbf{v}_{1:\tau_s}$, represents the sentence, where its length $\tau_s$ depends on the length of the spectral feature sequence, $\bar{T}_s$, extracted from the sentence, the mean length of the keyword, $\bar{T_w}$, and the sliding hop size. We apply the word classifier trained as described above in Sec. 7.2.1, to the sequence of histograms. Discriminative binary classifiers mostly produce a score value on which a threshold operation is applied to produce the classified label. In case of a linear classifier, for example, this score would be the distance of of the test vector from the classifying hyperplane. Inspired by previous work, [14, 15] we use these score values to form a response curve, $\mathbf{S}_{1:\tau_s} = (S_1, ..., S_{\tau_s})$, where $S_t$ is the score produced by the word classifier given the $t$-th histogram. Therefore a positive sentence is expected to yield a response curve having a distinct maximal value corresponding to the location of the keyword in the spoken sentence, while a negative sentence is expected to lead to random-like response. Figures 7.1(a) and 7.1(b) present the waveform, the spectrogram and the response curve, extracted as described above for the sentences “help me unroll the new rug” and “you didn’t arrive too late”, correspondingly, where the searched keyword is “unroll”. Note that the response curve related to the positive sentence, Fig. 7.1(a), has a distinct-positive valued maximum point, as opposed to the response curve related to the negative sentence, Fig. 7.1(b), which is quite random and mostly below zero.

A simple approach for classifying a response curve is to apply a threshold, as performed elsewhere [14, 15]. In this work we generalize this operation by training a binary classifier based on global features extracted from the response curve. Define $\sigma$ as the standard deviation of the response curve:

$$\sigma = \sqrt{\frac{1}{\tau_s}\sum_{t=1}^{\tau_s}\left(S_t - \frac{1}{\tau_s}\sum_{t'=1}^{\tau_s} S_{t'}\right)^2} \tag{7.5}$$

The global feature vector is $\phi = \left(M_x, m_n, a, DN, \delta, \delta^2\right)$, where:

- Normalized maximal value - $M_x = \max\{\mathbf{S}_{1:\tau_s}\}/\sigma$
- Normalized minimal value - $m_n = \min\{\mathbf{S}_{1:\tau_s}\}/\sigma$
- Normalized mean value - $a = \sum_{t=1}^{\tau_s}\{\mathbf{S}_t\}/\sigma$



(a)

(b)

Figure 7.1: Detection of the keyword "unroll" from the sentence: (a) "help me unroll the new rug"; (b) "you didn't arrive too late". Top - waveform; middle - spectrogram; bottom - response curve (solid blue) and zero response (dashed black).

- Normalized dynamic range - $DN = M_x - m_n$

- First Derivative - $\delta = \sum_{t=2}^{\tau_s} d_t/\sigma$, where $d_t = S_t - S_{t-1}$

- Second Derivative - $\delta^2 = \sum_{t=3}^{\tau_s} \left(d_t - d_{t-1}\right)/\sigma$.

Given response curves related to positive and negative training sentences, we obtain their global feature vectors and train a sentence classifier.

### 7.2.3 Bagging Predictors

In practice, labeled samples are harder to acquire than unlabeled ones. Therefore, we address the case where the amount of positive examples $N^+$ is very small, compared to the amount of negative examples $N^-$. It is preferable to use all available labelled data when training a discriminative classifier, to increase robustness. However, an extremely unbalanced training set will lead to a biased classifier, classifying almost everything as negative. To avoid this bias, and still, utilize the variety of the negative set we use bagging predictors [17]. When training an isolated word classifier we randomly select negative examples from the negative set, at the same amount as the number of available positive examples, $N^+$. We repeat this sampling to obtain $L_1$ negative subsets. Each negative subset along with the positive set is used to train a binary classifier, so that eventually we have $L_1$ isolated word classifiers. We use the same strategy for training the sentence classifiers by randomly selecting $L_2$ negative sets, each containing negative sentences at the same amount as the size of the positive set. At the end of the training process, we have $L_1$ isolated word classifiers and $L_2$ sentence classifiers.

### 7.2.4 Inference

Given a sequence of spectral feature vectors, $(\mathbf{x}_1, ..., \mathbf{x}_{T_s})$, related to a test sentence, inference is made as depicted in Fig. 7.2: we obtain the sequence of histograms representing the sentence, $\mathbf{v}_{1:\tau_s}$, with respect to the GMM, $\mathcal{M}$, using a sliding window according to eqns. (7.2)-(7.4). $L_1$ isolated word classifiers are applied producing $L_1$ response curves $\mathbf{S}^l_{1:\tau_s}$, $l = 1, ..., L_1$. The global feature vectors, $\phi^l$, $l = 1, ..., L_1$, are extracted from each response curve as described above. $L_2$ sentence classifiers are applied to the global feature vectors, producing $L_1 \cdot L_2$ predictions. A final decision is made by taking a majority decision.

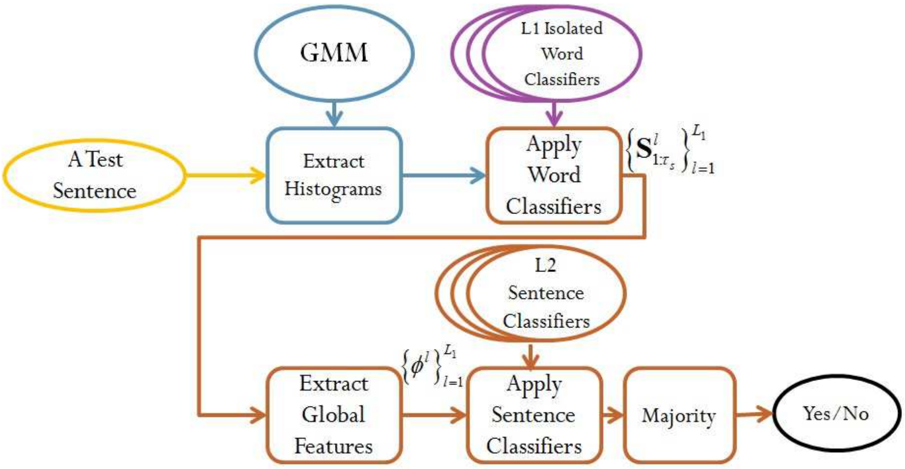

GMM
L1 Isolated Word Classifiers
A Test Sentence
Extract Histograms
Apply Word Classifiers
$\left\{\mathbf{S}^l_{1:\tau_s}\right\}^{L_1}_{l=1}$
L2 Sentence Classifiers
$\left\{\phi^l\right\}^{L_1}_{l=1}$
Extract Global Features
Apply Sentence Classifiers
Majority
Yes/No

Figure 7.2: Inference using the proposed approach for keyword spotting.

## 7.3 Experimental Results

The proposed keyword spotting system comprises two classifiers: an isolated word classifier - applied to sequences of histograms, and a sentence classifier - applied to global features. To demonstrate the advantages of our proposed approach we begin with an examination of the proposed isolated word classifier, in comparison with a classifier based on GMM-HMM. We proceed with a keyword spotting task, comparing the proposed system to an unsupervised HMM-based keyword spotter, using recordings of adults. We conclude by presenting the performance of the examined systems tested on clean and noisy speech signals of children. We used recording of children taken from CSLU [72], and adults' recordings taken from TIMIT, [73]. Noise signals were added to TIMIT and to CSLU datasets using a "Filtering and Adding Noise Tool" (Fant) [74].

Mel Frequency Cepstral Coefficients (MFCC's) along with their first and second derivatives were extracted from all waveforms every 10msecs using Kaldi - an open source software [75]. We used LIBSVM - an available toolkit [76] - for training our proposed classifiers for isolated word and sentence classification. We tested several kernels: the standard linear and RBF kernels, and a Chi-Square kerenl defined as:

$$K\left(\mathbf{x}_i, \mathbf{x}_j\right) = \exp\left(-\gamma \sum_n \frac{\left(x_i\left(n\right) - x_j\left(n\right)\right)^2}{x_i\left(n\right) + x_j\left(n\right)}\right), \tag{7.6}$$

where $\gamma$ is a parameter. In all our experiments, the Chi-Square kernel of eqn. (7.6), led to the best results for isolated word classification and a linear kernel was found best for sentence classification, so all the results presented here were obtained accordingly. All HMM-based systems presented here were trained using an available toolkit[1], where a wide range of values was explored for the amount of emitting states mixture components (6–25 and 1–8, respectively). Final values were set separately for each of the examined setups by using cross-validation.

### 7.3.1 Isolated Word Classification

In this section we present the performance of the proposed isolated word classifier, used in our overall keyword spotting system. In a keyword spotting task, this classifier is trained for a binary classification task between histograms related to the keyword or to non-keyword speech. In this section we demonstrate its performance in a more challenging task - a multi-class classification task - from among a given dictionary. Since no transcription was used for training (except for the label of each word, signifying if it is a keyword or not), we used an unsupervised HMM classifier as a benchmark, where all utterances of a specific word are used to train an HMM. Inference is made according to the HMM leading to the highest likelihood score (using Viterbi decoding).

For training and evaluation, we used recordings of children saying isolated words, taken from the CSLU database. We examined three different vocabularies, each consisting 10 words, defining three

[1] http://www.cs.ubc.ca/ murphyk/Software/HMM/hmm.html

different classification tasks[2]. All parameters (number of states and components for the HMM and SVM-constant in our approach) were set using a 10-fold cross validation, where in each fold, 8/10 of the data set was used for training, 1/10 for setting the parameters and 1/10 for testing. The final values for number of emitting states and mixture components were set between 6–12 and 1–2, respectively.

The spectral features of speech signals of children varies with age: young kids (6-10 years old) have higher variability in terms of the shape and location of formants. Towards their teens, the speech characteristics become more stable and more similar to adults' [77]. To examine the robustness of our system and the HMM classifier to this variability, we divided the data into three age groups: "low" - kindergarten–fifth grade, "high" - sixth grade–tenth grade, and "all" - kindergarten–tenth grade. We trained three classifiers using these age groups and tested each one on its corresponding group and on the other two.

The results presented in Table 7.1 are the accuracy rates, mean and STD values, achieved by each method, averaged over the three tasks, including all combinations of training and testing sets among age groups. In general, higher accuracy is achieved when training and testing are performed using the same age group, where the "low" age group was harder for both methods due to the high variability in speech signals of young children. Nevertheless, the proposed method leads to higher accuracy rates than HMM in all cases: 4-6% higher for training and testing on the same age group and 3-11% higher for cross-ages training and testing. Also note that the STD of the HMM classifier is between 1.3 and 3.4 for the "low" and "high" age groups while the STD of the proposed system is lower than 1 in both cases. For the "all" age group both methods lead to similar and low STD values. This indicates that the proposed classifier is more robust to training set size and variability than the HMM classifier, as it leads to more consistent accuracy rates.

### 7.3.2 Keyword Spotting - Speech Of Adults (TIMIT)

The TIMIT dataset was used to examine the performance of our proposed system for speech of adults, in a keyword spotting task. We followed the protocol presented by Ezzat and Poggio [14], and later by Barnwal et al. [15], and selected four frequent words in TIMIT as keywords: "greasy", "dark", "wash", and "oily". We used the standard division of TIMIT for training and testing (73% and 27%, respectively). To demonstrate the influence of the amount of positive examples, we trained the examined systems using several sets, consisting of 5, 10, 50, 100 and 200 positive examples, where 20% of the training set was taken as a development set, i.e., used for setting the parameters of the examined methods. In all the experiments, a single negative set was used, consisting of 100 sentences that do not include any of the

[2] The three vocabularies used for the isolated word classification experiments are: 1) background, bathe, behind, beyond, bigfoot, biology, birthmark, boomerang, breath, bronco. 2) earthquake,easier,eight, employees, endure, engrave, ethnic, explosion, faithful, fancy. 3) gumshoe, handshake, hardship, hawthorne, herbalist, homemaking, hoof, hopeful, hourly, humor.

Table 7.1: *Isolated word classification of speech of children (CSLU database): classification accuracy rates - mean and STD values, averaged over three different 10-word vocabularies.*

| | | Test Data | | |
|---|---|---|---|---|
| Train. Data | | "Low" [%] | "High" [%] | "All" [%] |
| "Low" | HMM | 89.0± 1.3 | 84.7 ± 2.9 | 86.4± 1.3 |
| | proposed | **94.6 ± 0.5** | **91.1± 0.2** | **93.2± 0.3** |
| "High" | HMM | 75.1± 2.7 | 91.0 ± 3.4 | 79.2± 4.2 |
| | proposed | **86.1± 0.8** | **97.2± 0.6** | **90.5± 0.7** |
| "All" | HMM | 91.4± 0.8 | 94.7± 0.4 | 92.7± 0.5 |
| | proposed | **95.3± 1.0** | **97.4± 0.3** | **96.1± 0.5** |

keywords.

### Benchmark System

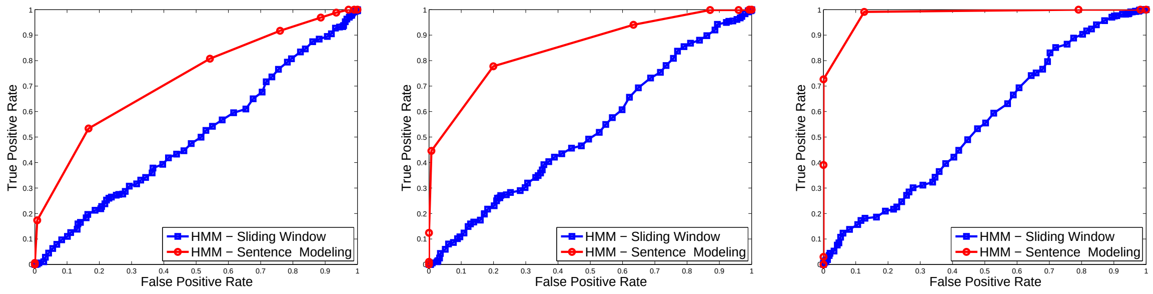
We examined two configurations for HMM keyword spotters to be used as our benchmark:

- Ezzat et al. [14] - an isolated word HMM-based classifier, trained using isolated utterances of the keyword. Inference is performed by applying a threshold onto a response curve comprised of log-likelihood values, obtained using a sliding window ($1.2\bar{T}_w$ long window with a $0.2\bar{T}_w$ hop).

- Keshet et al. [13] - training two HMMs: 1) a garbage+keyword model trained using the positive sentences 2) a garbage model trained using the negative sentences. Inference is performed by comparing the log-likelihood of one HMM to the log-likelihood of the other HMM.

Our system does not use phonetic transcription as done by Keshet et al. [13]. Hence, to allow a fair comparison between this methods and ours, we used unsupervised training for both HMM-based keyword spotters. As mentioned above, a wide range of emitting states and mixture components was examined, 6–25 and 1–8 respectively, for each HMM configuration, training-set size and fold. In general, as more positive examples are available for training, the tuning process resulted in selecting models trained with more emitting states and more mixture components.

Fig. 7.3 presents the Receiver Operating Characteristic (ROC) curves attained by the two HMM configurations averaged over detection of the words "greasy", "dark", "wash", and "oily" using 5, 10 and 50 positive examples. According to our experiments, the garbage+keyword approach leads to significantly higher detection rates than the sliding window approach. Therefore we used the garbage+keyword HMM as a benchmark. For simplicity, from now on we will refer to the garbage+keyword HMM approach as HMM.



(a) (b) (c)

Figure 7.3: ROC curve of benchmark systems averaged over detection of the words “greasy”, “dark”, “wash”, and “oily”, trained: HMM with sliding window - blue square; HMM with garbage+keyword models - red circle, using: (a) - 5 training sentences, (b) - 10 training sentences and (c) - 50 training sentences.

## Proposed System

Fig. 7.4 presents the Area Under the ROC Curve (AUC), averaged over detection of the words “greasy”, “dark”, “wash”, and “oily”, using 5–50 positive examples, obtained by the proposed system. In this experiment, several values of bagging predictors were used for word and sentence classification: $L_1 = L_2 = L \in [1, 5, 11, 51, 75]$. As expected, bagging predictors are mostly useful when the training set is highly uneven, in our case, when just 5 or 10 positive examples are available. For 50 positive examples, comparable results are achieved, regardless of the value of $L$.

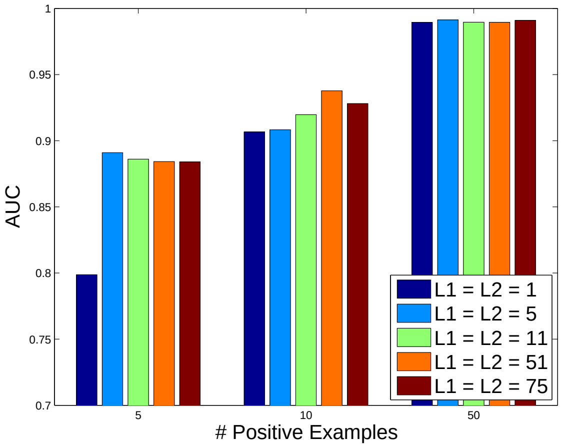


Figure 7.4: AUC averaged over detection of four keywords (“greasy”, “dark”, “wash”, and “oily”), obtained by the proposed approach using 5–50 positive examples, and $L_1 = L_2 = L \in [1, 5, 11, 51, 75]$.

Fig. 7.5 presents the averaged AUC values obtained by the proposed system, using 10 positive examples, for all combinations of $L_1, L_2 \in [1, 5, 11, 51, 75]$. In general, the AUC increases with $L_1$ and $L_2$. As mentioned above, the final decision regarding a sentence is made through a majority decision of $L_1 \cdot L_2$ predictions, so having more predictions raises the detection rate. Still, from this experiment it is clear that the amount of bagging predictors used for word classification, $L_1$, has a greater influence on the final outcome, in terms of AUC, than the amount of bagging predictors used for sentence classification, $L_2$. For a large $L_1$, in this case (greater than 51) it is sufficient, and also preferable to take $L_2 = 5$. We used the development set to tune the amount of bagging predictors and set them to be $L_1 = 51$ and $L_2 = 5$, in all further experiments in this section.

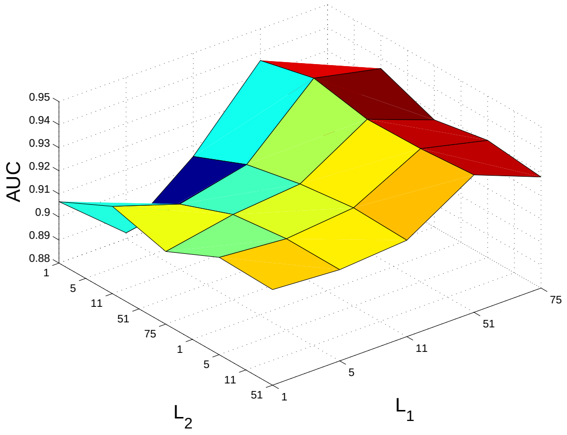


Figure 7.5: AUC averaged over detection of four keywords (”greasy”, “dark”, “wash”, and “oily”) obtained by the proposed approach using 10 positive examples and $L_1, L_2 \in [1, 5, 11, 51, 75]$.

Table 7.2 presents the overall AUC results obtained by the HMM benchmark system and the proposed approach for detection of the four keywords “greasy”, “dark”, “wash”, and “oily” and also the mean value for all four words, using 5–50 sentences. The mean and STD values presented in this table were obtained by averaging over 10 repetitions of each experiment, using randomly selected sub-sets for training, taken from the overall training set of TIMIT. Our approach outperforms, on average, the benchmark for 5–50 positive training examples. For 50 examples, HMM is slightly better for detection of the word “water”, but both methods reach almost perfect detection - AUC $\approx 1$. The results for 100–200 positive examples, are not presented in this table, as both methods are comparable, leading to almost perfect detection. Barnwal et al. [15] also followed the same experiment protocol suggesting a discriminative approach based on spectro-temporal features. While their system outperforms the patches-based discriminative system proposed by Ezzat et al. [14], our proposed approach exceeds both.

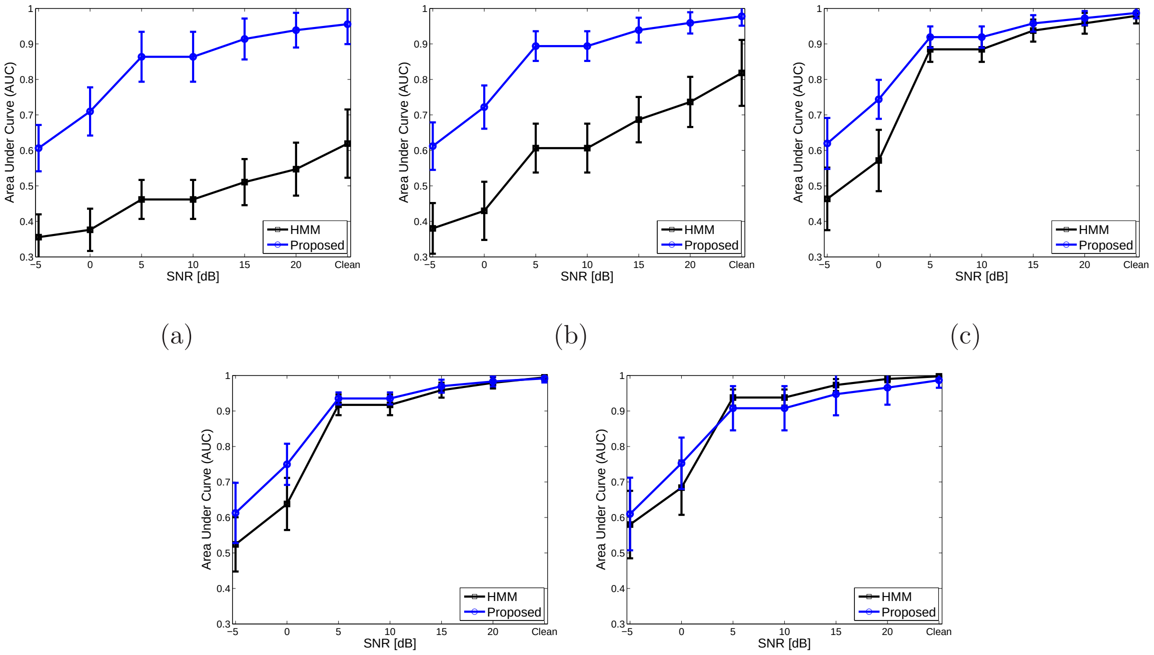
Table 7.2: Mean and STD of AUC results obtained by the HMM-based KWS and proposed system applied to speech of adults (TIMIT), averaged over 10 randomly selected training sets.

| | | **# Positive Examples** | | |
|---|---|---|---|---|
| **Keyword** | **Method** | **5** | **10** | **50** |
| **“greasy”** | **HMM** | 0.6±0.1 | 0.8 ±0.1 | 0.99 ± 0.01 |
| | **Proposed** | **0.94 ± 0.08** | **0.99 ± 0.004** | **0.995 ± 0.003** |
| **“dark”** | **HMM** | 0.6±0.1 | 0.8±0.1 | 0.96 ± 0.02 |
| | **Proposed** | **0.99 ±0.005** | **0.999 ± 0.001** | **0.997 ± 0.001** |
| **“wash”** | **HMM** | 0.6 ±0.1 | 0.8 ±0.1 | 0.99 ± 0.01 |
| | **Proposed** | **0.99 ± 0.001** | **0.99 ±0.002** | **0.995 ± 0.001** |
| **“water”** | **HMM** | 0.6 ±0.1 | 0.85 ±0.05 | **0.98 ± 0.01** |
| | **Proposed** | **0.9 ± 0.04** | **0.94 ±0.02** | 0.96±0.01 |
| **Mean** | **HMM** | 0.6±0.1 | 0.8 ±0.1 | 0.98 ± 0.02 |
| | **Proposed** | **0.96 ±0.04** | **0.98±0.02** | **0.99 ± 0.01** |

### 7.3.3 Keyword Spotting - Noisy Speech Signals Of Adults

We also compared the performance of the proposed approach and the benchmark system under noisy conditions. In this experiment, we used the same models trained by the clean set (Sec. 7.3.2), and applied them to noisy versions of the test set. Figs. 7.6 and 7.7 present the AUC (averaged over detection of the four keywords) obtained by the HMM benchmark and the proposed system when tested for two types of noise added to the speech signals. For 5–50 positive examples (Figs. 7.6(a)-7.6(c) and Figs. 7.7(a)-7.7(c)), mean STD values were obtained by averaging over detection of four keywords and over 10 repetitions. For 100–200 positive examples (Figs. 7.6(d), 7.6(e) and Figs. 7.7(d), 7.7(e)), mean STD values were obtained by averaging over detection of four keywords, but without repetitions. The two noise types were “babble” and “car”, at several SNR values, ranging from −5dB to 20dB. For comparison, the performance for testing on clean speech (presented above in the last row of Table 7.2) are are also presented in these figures. In general, the proposed system has a distinct advantage in extreme setups where very few positive examples are available for training and/or low SNR values at testing. Specifically, for 5–10 positive examples, the proposed system outperforms the HMM system at all SNR values and for both noise types. When training with 50 positive examples, the two systems are comparable for SNR $\geq$ 15dB. However, for SNR $<$ 15dB, the proposed system leads to higher AUC values than the benchmark system. The same behavior is observed for 100-200 positive examples, where the two systems are comparable for

SNR $\geq$ 5dB, but the proposed system is better for SNR $<$ 5dB.



(a)

(b)

(c)

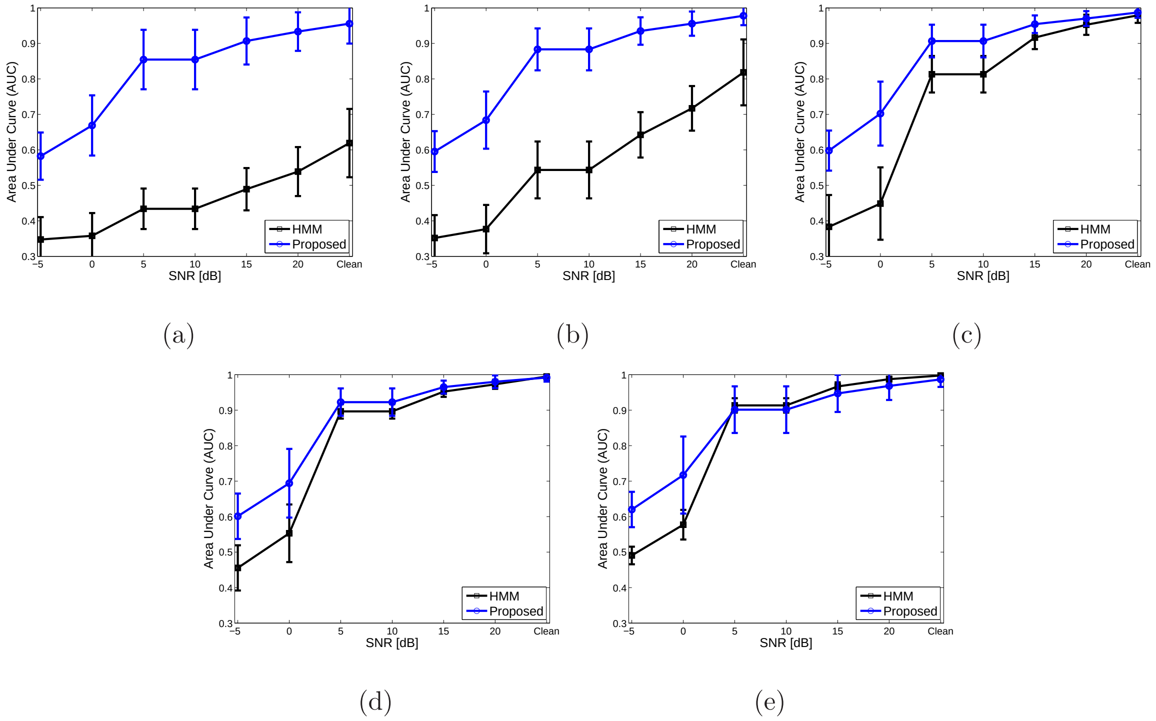
(d)

(e)

Figure 7.6: AUC averaged over detection of four keywords ("greasy", "dark", "wash", and "oily") taken from speech of adults (TIMIT), tested on clean and noisy speech (**babble noise**) by: HMM - black square; proposed approach - blue circle, using: (a) - 5 training sentences, (b) - 10 training sentences, (c) - 50 training sentences, (d) - 100 training sentences, and (e) - 200 training sentences. For 5–50 positive examples ((a)-(c)), mean and STD values were obtained by averaging 10 repetitions of randomly selected training sets.

### 7.3.4 Keyword Spotting - Speech Of Children (CSLU)

We examined our proposed approach also for speech of children taken from the CSLU dataset. We used three keywords which are most frequent in the dataset: "one", "two" and "unroll". Table. 7.3 presents the AUC (mean and STD) obtained by the HMM benchmark and the proposed systems for all age groups (kindergarten–tenth grade). For 5–50 positive examples, the AUC were averaged over detection of the three keywords and over 10 repetitions of each experiment, using randomly selected training sets. For 100–200 positive examples, the AUC were also averaged over detection of the three words, but without repetitions. The proposed method has a distinct advantage for 5-10 positive training examples, whereas for 50 and 200 positive examples, the benchmark system is better. For 100 positive examples, both methods are comparable.



(a) (b) (c)

(d) (e)

Figure 7.7: AUC averaged over detection of four keywords ("greasy", "dark", "wash", and "oily") taken from speech of adults (TIMIT), tested on clean and noisy speech (**car noise**) by: HMM - black square; proposed approach - blue circle, using: (a) - 5 training sentences, (b) - 10 training sentences (c) - 50 training sentences, (d) - 100 training sentences, and (e) - 200 training sentences. For 5–50 positive examples ((a)-(c)), mean and STD values were obtained by averaging 10 repetitions of randomly selected training sets.

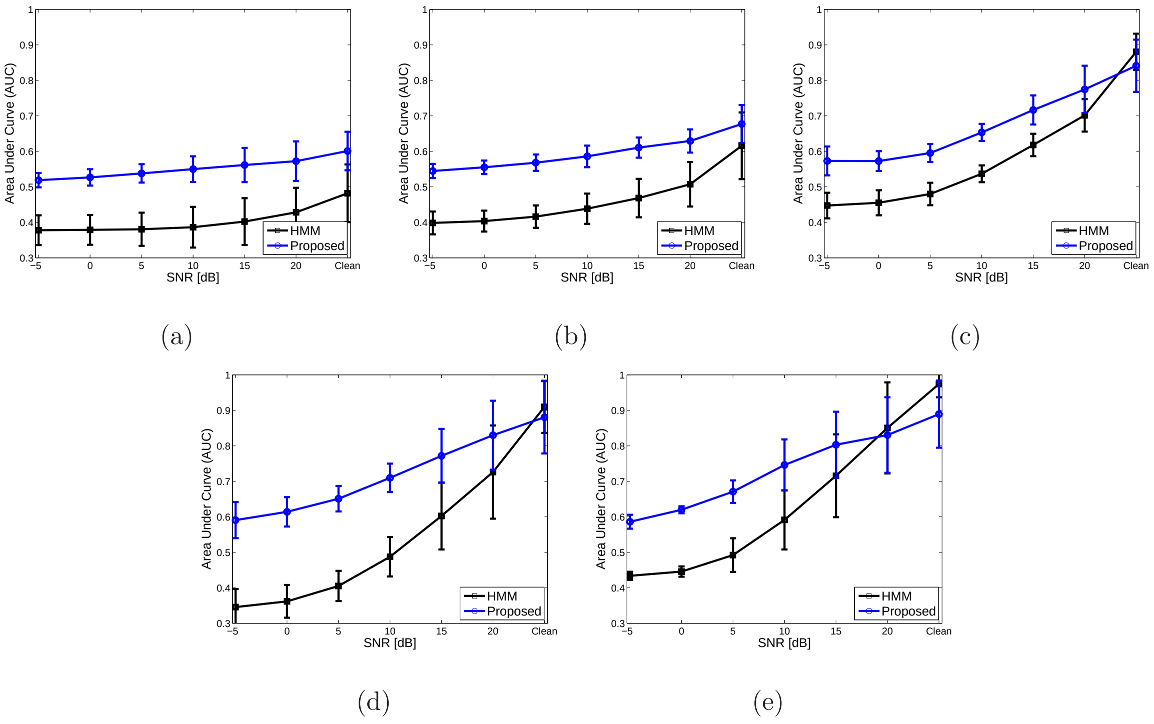
### 7.3.5 Keyword Spotting - Noisy Speech Signals Of Children

We tested the performance of the proposed approach and the benchmark system on noisy speech signals of children. In this experiment we used unified training and test sets consisting of all age groups together. We applied the same 10-fold cross validation as described above. At each fold, 8/10 of the set were used for training, 1/10 of the set for parameters setting (as before for clean speech), and 1/10 of the set, used for testing, was noisy. Figs. 7.8 and 7.9 present the AUC (averaged over detection of the three keywords) obtained by the HMM benchmark and the proposed system for testing on noisy speech signals. Similarly to Sec. 7.3.3, we examined the performance for testing two noise types - "babble" and "car", at several SNR values, ranging from $-5$dB to 20dB. For comparison, the performance for testing on clean speech (presented in Table. 7.3) is also presented in these figures. When testing on clean signals, our system leads to higher AUC values when 10 or less positive examples are available for training, whereas for 50 or more, the HMM system is comparable or better than the proposed system. When testing on noisy signals, our proposed system is more robust: it outperforms the HMM-benchmark system at all SNR

Table 7.3: Mean and STD of AUC results obtained by the HMM-based KWS and proposed system applied to speech of children (CSLU). Averaged over detection of three words: “one”, “two” and “unroll”. For 5–50 positive examples, also averaged over 10 repetitions of the experiment, using randomly selected training sets.

| | | **# Positive Examples** | | | | |
|---|---|---|---|---|---|---|
| | **Method** | **5** | **10** | **50** | **100** | **200** |
| **Mean AUC** | **HMM** | 0.5±0.1 | 0.6±0.1 | **0.9±0.1** | 0.9±0.1 | **0.97±0.03** |
| | **Proposed** | **0.6±0.05** | **0.7±0.05** | 0.8±0.1 | 0.9±0.1 | 0.9±0.1 |

values and for both noise types for 5–10 positive examples. For 50 positive examples or more, the two systems are comparable for high SNR values, still, for SNR ≤10dB, the proposed system is better.



(a)

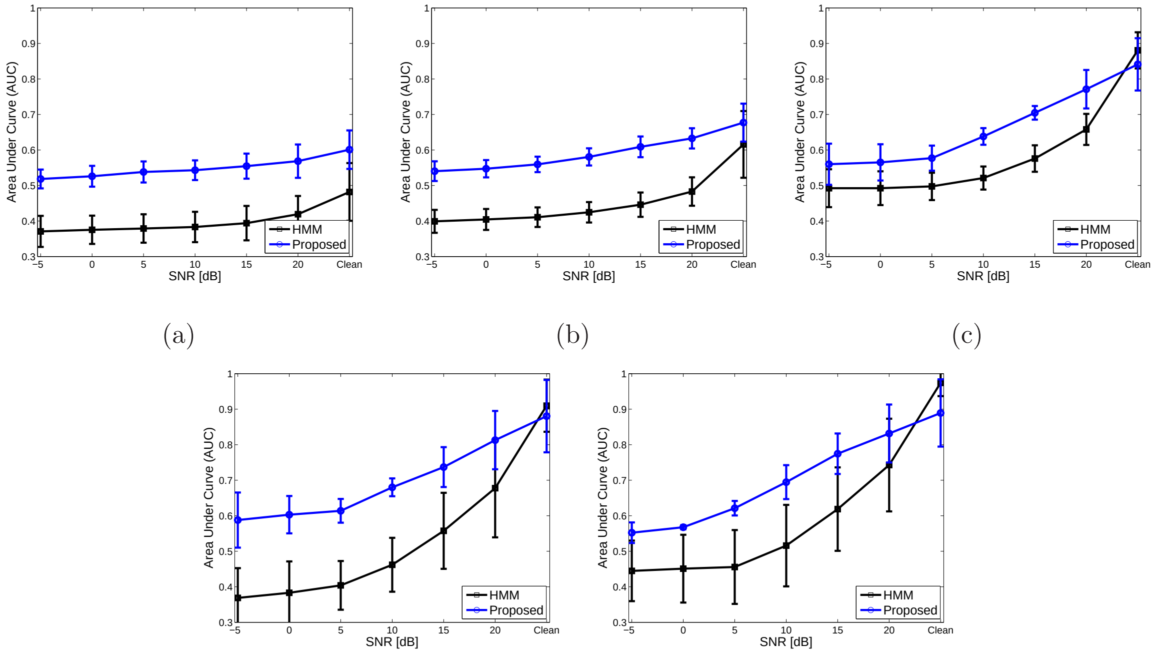
(b)

(c)

(d)

(e)

Figure 7.8: AUC averaged over detection of three keywords (“one”, “two”, and “unroll”) taken from the “All” age group (kindergarten to tenth grade) of CSLU, tested on clean and noisy speech (**babble noise**) by: HMM - black square; proposed approach - blue circle, using: (a) - 5 training sentences, (b) - 10 training sentences (c) - 50 training sentences, (d) - 100 training sentences, and (e) - 200 training sentences. For 5–50 positive examples ((a)-(c)), mean and STD values were obtained by averaging 10 repetitions of randomly selected training sets.



(a) (b) (c)

(d) (e)

Figure 7.9: AUC averaged over detection of three keywords ("one", "two", and "unroll") taken from "All" age group (kindergarten to tenth grade) of CSLU, tested on clean and noisy speech (**car noise**) by: HMM - black square; proposed approach - blue circle, using: (a) - 5 training sentences, (b) - 10 training sentences (c) - 50 training sentences, (d) - 100 training sentences, and (e) - 200 training sentences. For 5–50 positive examples ((a)-(c)), mean and STD values were obtained by averaging 10 repetitions of randomly selected training sets.

## 7.4 Chapter Summary

In this chapter we presented a novel approach for keyword spotting, specifically adequate for limited-data applications. It is based on fixed-length representations for words and for sentences, which enable training of discriminative classification methods such as Support Vector Machine (SVM). We avoided bias in training by using bootstrap aggregating, also referred to as bagging predictors, where a series of classifiers are trained using randomly sampled subsets of the larger training set. We demonstrated the advantages of our proposed approach, compared to an HMM-bases KWS benchmark system through a series of experiments using speech singles of both adults and children in several challenging setups, considering training-set size, and background noises - car and babble.

# Chapter 8

# Conclusion

## 8.1 Summary of Main Contributions

In this work we dealt with two main tasks of speech processing in low resource environments: a voice conversion task, and a keyword spotting task. Common methods for both tasks use relatively large data sets for training, which are not always available in case of limited resources such as mobile applications, under-documented languages or speech of children.

We addressed the following elements of the voice conversion task:

- **Speech quality -** we presented two methods for enhancing the global variance of converted signals and, as a result, improving their perceived quality and individuality: 1) training of a GMM-based conversion with a GV constraint; 2) a modular block for GV enhancement, applied as a post processing bock.

- **Low resource applications -** we proposed a new method for voice conversion, called Grid-Based conversion, which is suitable for low resource applications. This approach is based on sequential Bayesian tracking, by which the conversion process is expressed as a sequential estimation problem of tracking the target spectrum, based on the observed source spectrum. Combined with our post processing block for GV enhancement, the overall system is easily and successfully trained using very small data sets. In our subjective evaluations, comparing the overall enhanced GB system to the enhanced GMM conversion method and to CGMM, the enhanced GB system was marked as best, in terms of individuality, and as comparable to the enhanced GMM method, in terms of quality.

- **Non-parallel training -** for this setup, where the training set does not consist of the source and target speakers saying the same text, we presented a generalization of an existing method known as INCA. For determining the source-target matching, we proposed using temporal context vectors, rather than single feature vectors that are used in INCA. Furthermore, we formulated the training process as a minimization problem of a joint cost of the source-target matching and the spectral distance between the converted and target vectors. We showed that this optimization problem can be solved using an alternating minimization procedure which converges to a local minima, and by this way proved its convergence in the particular case of INCA. Our experimental results show that compared to INCA, our approach improves the matching process used for training the conversion, and as a result, improves the quality and individuality of the synthesized output signals.

For the keyword spotting task we proposed a new discriminative method which is suitable for limited-data setups such as: mobile applications, under-documented languages and speech of children. We presented a new fixed-length representation for isolated words, based on histograms obtained with respect to a pre-trained GMM. We also proposed a fixed-length representation for sentences, based on global feature vectors, extracted from the response curve obtained by the word classifier. A highly biased training set is a reasonable scenario in limited-data applications, so in order to avoid biased classifiers we used bagging predictors for training both word and sentence classifiers. According to our experiments on speech of adults, the proposed system outperforms the standard HMM-based benchmark system in challenging setups of small training sets and/or low SNR values. In case of highly variable speech of children, when tested on clean speech, our system outperformed the benchmark when 50 or fewer examples are available for training. Moreover, in presence of noise, our system leads to higher detection rates for all the examined cases, regardless of training-set size, noise type or SNR.

## 8.2 Further Research

Most voice conversion methods, including those presented in this work, deal with converting the spectral envelope. However, the identity of a speaker is also closely related to his prosody features. The global linear pitch conversion function, described in this work and commonly applied in other systems, is effective, but still, a more sophisticated modeling and conversion of the pitch contour of the two speakers would probably improve the individuality of the converted outcome. In this work, as in most other systems, the speaking rate was not converted al all. The spectral conversion process is performed by simply replacing the spectral envelopes extracted from the source signal with the converted outcome. As a result, the synthesized output has the same speaking rate as the source speaker. Further improvement can be obtained by modifying the duration of each converted utterance to match, on average, its corresponding value for the target speaker.

Spectral distortion and GV are commonly used as objective measures since they provide a simple and fully automated way for evaluating conversion systems. These objective measures may express significant trends and phenomena, but, as shown in this work, they do not always agree with subjective evaluation results. Further research is needed to design alternative measures for objective evaluation of conversion systems, with better correspondence to subjective results. In the mean time, subjective listening tests are imperative to properly evaluate and compare conversion methods.

The proposed GB conversion method, as presented here, is based on soft correspondence between the source and target vectors, obtained by using a parallel training set. The TC-INCA method presented here is a general approach for matching source and target spectra, enabling training of parallel voice conversion methods in a non-parallel setup. It is based on the classical GMM-based conversion and therefore requires several dozens of recorded sentences for training. Further research is needed for merging these two approaches - TC-INCA for matching, and GB for spectral conversion, to design a non-parallel voice conversion system, in a limited-data setup.

The histogram representation of keywords presented in this work is obtained with respect to a GMM. Therefore, the temporal correspondence of the spectral feature vectors is ignored. An alternative model, considering the temporal context of spectral feature vectors could provide better modeling of the keyword, and as a result, improve the detection rate. In this work we proposed a set of global features for representing sentences. These features were selected since they characterise the differences between positive and negative response curves. Still, exploring other features may lead to improved representation and classification of positive and negative response curves and therefore to improved detection rate.

# Bibliography

[1] D. Erro, A. Moreno, and A. Bonafonte, “INCA algorithm for training voice conversion systems from nonparallel corpora,” *IEEE Trans. on Audio, Speech and Lang. Proc.*, vol. 18, no. 5, pp. 944–953, 2010.

[2] Y. Stylianou, O. Cappé, and E. Moulines, “Continuous probabilistic transform for voice conversion,” *IEEE Trans. on Audio, Speech and Lang. Proc.*, vol. 6, no. 2, pp. 131–142, 1998.

[3] A. Kain and M. Macon, “Spectral voice conversion for text-to-speech synthesis,” in *Proc. ICASSP*, 1998, pp. 285–288.

[4] T. Toda, A. W. Black, and K. Tokuda, “Spectral conversion based on maximum likelihood estimation considering global variance of converted parameter,” in *Proc. ICASSP*, 2005, pp. 9–12.

[5] T. Toda, H. Saruwatari, and K. Shikano, “Voice conversion algorithm based on gaussian mixture model with dynamic frequency warping of straight spectrum,” in *Proc. ICASSP*, 2001, pp. 841–844.

[6] T. B. A. Toda and K. Tokuda, “Voice conversion based on maximum-likelihood estimation of spectral parameter trajectory,” *IEEE Trans. on Audio, Speech and Lang. Proc.*, vol. 15, no. 8, pp. 2222–2235, 2007.

[7] Y. Nankaku, K. Nakamura, T. Toda, and K. Tokuda, “Spectral conversion based on statistical models including time-sequence matching,” in *Proc. ISCA*, 2007, pp. 333–338.

[8] W. Shen, C. M. White, and T. J. Hazen, “A comparison of query-by-example methods for spoken term detection,” DTIC Document, Tech. Rep., 2009.

[9] C. Parada, A. Sethy, and B. Ramabhadran, “Query-by-example spoken term detection for oov terms,” in *ASRU*. IEEE, 2009, pp. 404–409.

[10] P. Fousek and H. Hermansky, “Towards ASR based on hierarchical posterior-based keyword recognition,” in *Proc. ICASSP*, vol. 1. IEEE, 2006, pp. I–I.

[11] Y. Zhang and J. R. Glass, “Unsupervised spoken keyword spotting via segmental dtw on gaussian posteriorgrams,” in *ASRU*. IEEE, 2009, pp. 398–403.

[12] H. Wang, T. Lee, and C.-C. Leung, “Unsupervised spoken term detection with acoustic segment model,” in *Speech Database and Assessments (Oriental COCOSDA)*. IEEE, 2011, pp. 106–111.

[13] J. Keshet, D. Grangier, and S. Bengio, "Discriminative keyword spotting," *Speech Communication*, vol. 51, no. 4, pp. 317–329, 2009.

[14] T. Ezzat and T. Poggio, "Discriminative word-spotting using ordered spectro-temporal patch features." in *Proc. Interspeech*, 2008, pp. 35–40.

[15] S. Barnwal, K. Sahni, R. Singh, and B. Raj, "Spectrographic seam patterns for discriminative word spotting," in *Proc. ICASSP*. IEEE, 2012, pp. 4725–4728.

[16] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *J. R. Stat. Soc. B*, vol. 39, pp. 1–38, 1977.

[17] L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, no. 2, pp. 123–140, 1996.

[18] R. W. Schafer and L. R. Rabiner, "Digital representations of speech signals," *Proceedings of the IEEE*, vol. 63, no. 4, pp. 662–667, 1975.

[19] R. McAulay and T. Quatieri, "Speech analysis/synthesis based on a sinusoidal representation," *IEEE Trans. on Acoustics, Speech and Signal Proc.*, vol. 34, no. 4, pp. 744–754, 1986.

[20] Y. Stylianou, "Applying the harmonic plus noise model in concatenative speech synthesis," *IEEE Trans. on Audio, Speech and Lang. Proc.*, vol. 9, no. 1, pp. 21–28, 2001.

[21] O. Cappe and E. Moulines, "Regularization techniques for discrete cepstrum estimation," *IEEE Signal Processing Letters*, vol. 3, no. 4, pp. 100–102, 1996.

[22] A. M. Kondoz, *Digital speech: coding for low bit rate communication systems*. John Wiley & Sons, 2005.

[23] H. Duxans, E. Erro, J. Perez, A. Bonafonte, and A. Moreno, "Voice conversion of non-aligned data using unit selection," in *TC-Star Speech to Speech Translation Workshop*, 2006, pp. 237–242.

[24] E. Moulines and J. Laroche, "Techniques for pitch-scale and time-scale transformation of speech. part i, non parametric methods," *Speech Communication*, vol. 16, pp. 175–205, 1995.

[25] E. Moulines and F. Charpentier, "Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones," *Speech Communication*, vol. 9, pp. 453–467, 1990.

[26] D. T. Chappell and J. H. Hansen, "Speaker-specific pitch contour modeling and modification," in *Proc. ICASSP*, vol. 2. IEEE, 1998, pp. 885–888.

[27] Z. Inanoglu, "Transforming pitch in a voice conversion framework," *St. Edmonds College, University of Cambridge, Tech. Rep*, 2003.

[28] Z. Wu, T. Kinnunen, E. Chng, and H. Li, "Text-independent f0 transformation with non-parallel data for voice conversion." in *Proc. Interspeech*, 2010, pp. 1732–1735.

[29] H. Ye and Y. S., "Quality-enhanced voice morphing using maximum likelihood transformations," *IEEE Trans. on Audio, Speech and Lang. Proc.*, vol. 14, no. 4, pp. 1301–1312, 2006.

[30] "Multi stimulus test with hidden reference and anchors (MUSHRA)," International Telecommunications Union, Tech. Rep. ITU-R BS.1534-1, Jan. 2003.

[31] E. Godoy, O. Rosec, and T. Chonavel, "Voice conversion using dynamic frequency warping with amplitude scaling, for parallel or nonparallel corpora," *IEEE Trans. on Audio, Speech and Lang. Proc.*, vol. 20, no. 4, pp. 1313–1323, 2012.

[32] M. R. Abe, S. Nakamura, K. Shikano, and H. Kuwabara, "Voice conversion through vector quantization," in *Proc. ICASSP*, 1988, pp. 655–658.

[33] L. M. Arslan, "Speaker transformation algorithm using segmental codebooks (stasc)," *Speech Communication*, vol. 28, no. 3, pp. 211–226, 1999.

[34] D. Sündermann, H. Höge, A. Bonafonte, H. Ney, A. Black, and S. Narayanan, "Text-independent voice conversion based on unit selection," in *Proc. ICASSP*, vol. 1. IEEE, 2006, pp. I–I.

[35] T. Dutoit, A. Holzapfel, M. Jottrand, A. Moinet, J. Perez, and Y. Stylianou, "Towards a voice conversion system based on frame selection," in *Proc. ICASSP*, vol. 4. IEEE, 2007, pp. IV–513.

[36] Z. Wu, T. Virtanen, E. S. Chng, and H. Li, "Exemplar-based sparse representation with residual compensation for voice conversion," *IEEE Trans. on Audio, Speech and Lang. Proc.*, vol. 22, no. 10, pp. 1506–1521, 2014.

[37] M. Zhang, J. Tao, J. Tian, and X. Wang, "Text-independent voice conversion based on state mapped codebook," in *Proc. ICASSP*, 2008, pp. 4605–4608.

[38] P. Song, W. Zheng, and L. Zhao, "Non-parallel training for voice conversion based on adaptation method," in *Proc. ICASSP*, 2013.

[39] A. Mouchtaris, J. Van der Spiegel, and P. Mueller, "Non-parallel training for voice conversion by maximum likelihood constrained adaptation," in *Proc. ICASSP*, vol. 1, 2004, pp. I–1.

[40] T. Toda, Y. Ohtani, and K. Shikano, "Eigenvoice conversion based on Gaussian mixture model," in *Proc. ICSLP*, pp. 2446–2449.

[41] A. Kain and M. W. Macon, "Design and evaluation of a voice conversion algorithm based on spectral envelope mapping and residual prediction," in *Proc. ICASSP*, 2001, pp. 813–816.

[42] E. Helander, T. Virtanen, J. Nurminen, and M. Gabbouj, "Voice conversion using partial least squares regression," *IEEE Trans. on Audio, Speech and Lang. Proc.*, vol. 18, no. 5, p. 91217921, 2010.

[43] H. Benisty and D. Malah, "Voice conversion using GMM with enhanced global variance," in *Proc. Interspeech*, 2011, pp. 669–672.

[44] H. Benisty, D. Malah, and K. Crammer, "Modular global variance enhancement for voice conversion systems," in *Proc. EUSIPCO*, 2012, pp. 370–374.

[45] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 3rd ed. The Johns Hopkins University Press, 1996.

[46] J. Kominek and A. W. Black, "CMU ARCTIC databases for speech synthesis," 2003.

[47] Software available at http://gps-tsc.upc.es/veu/soft/soft/vc_toolkit/.

[48] D. Erro, A. Moreno, and A. Bonafonte, "Voice conversion based on weighted frequency warping," *IEEE Trans. on Audio, Speech and Lang. Proc.*, vol. 18, no. 5, pp. 922–931, 2010.

[49] E. Helander, J. Schwarz, S. H. Nurminen, J, and M. Gabbouj, "On the impact of alignment on voice conversion performance," in *Proc. Interspeech*, 2008, pp. 1453–1456.

[50] T. Toda, Y. Ohtani, and K. Shikano, "Eigenvoice conversion based on Gaussian mixture model," in *Proc. ICSLP*, 2006, pp. 2446–2449.

[51] T. Toda, T. Muramatsu, and H. Banno, "Implementation of computationally efficient real-time voice conversion." in *Proc. Interspeech*, 2012.

[52] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Trans. on Signal Proc.*, vol. 50, no. 2, pp. 174–188, 2002.

[53] H. Benisty, D. Malah, and K. Crammer, "Sequential voice conversion using grid-based approximation," in *Proc. IEEEI.* IEEE, 2014, pp. 1–5.

[54] N. Xu, Z. Yang, L. Zhang, W. Zhu, and J. Bao, "Voice conversion based on state-space model for modelling spectral trajectory," *Electronics letters*, vol. 45, no. 14, pp. 763–764, 2009.

[55] B. Anderson and J. Moore, "Optimal filtering. 1979," 1979.

[56] H. Benisty, D. Malah, and K. Crammer, "Non-parallel voice conversion using joint optimization of alignment by temporal context and spectral distortion," in *Proc. ICASSP*, 2014, pp. 7909–7913.

[57] Software available at: http://aholab.ehu.es/users/derro/software.html.

[58] D. Erro, I. Sainz, and I. Hernaez, "Improved HNM-based vocoder for statistical synthesizers," in *Proc. Interspeech*, 2011, pp. 1809–1812.

[59] H. Kuwabara and Y. Sagisaka, "Acoustic characteristics of speaker individuality: Control and conversion," *IEEE Trans. on Signal Proc.*, vol. 16, no. 2, pp. 165–173, 1995.

[60] D. Erro, A. Moreno, and A. Bonafonte, "Inca algorythm for training voice conversion systems from nonparallel corpora," *IEEE Trans. on Audio, Speech and Lang. Proc.*, vol. 18, no. 5, pp. 944–953, 2010.

[61] I. Csiszar and G. Tusnady, "Information geometry and alternating minimization procedures," *Statistics and Decisions*, vol. 1, pp. 205–237, 1984.

[62] Y. Stylianou, J. Laroche, and E. Moulines, "High-quality speech modifcation based on a harmonic + noise model," in *Proc. EUROSPEECH*, 1995.

[63] Y. Stylianou, "Applying the harmonic plus noise model in concatenative speech synthesis," *IEEE Trans. on Audio, Speech and Lang. Proc.*, vol. 9, no. 1, pp. 21–29, 2001.

[64] E. Helander, J. Nurminen, and M. Gabbouj, "Lsf mapping for voice conversion with very small training sets," in *Proc. ICASSP*, 2008, pp. 4669–4672.

[65] J. S. Garofolo, C. G. Auzanne, and E. M. Voorhees, "The trec spoken document retrieval track: A success story." *NIST SPECIAL PUBLICATION SP*, vol. 500, no. 246, pp. 107–130, 2000.

[66] L. Boves, R. Carlson, E. W. Hinrichs, D. House, S. Krauwer, L. Lemnitzer, M. Vainio, and P. Wittenburg, "Resources for speech research: present and future infrastructure needs." in *Proc. Interspeech*, 2009, pp. 1803–1806.

[67] D. A. James and S. J. Young, "A fast lattice-based approach to vocabulary independent wordspotting," in *Proc. ICASSP*, vol. 1. IEEE, 1994, pp. I–377.

[68] K. Thambiratnam and S. Sridharan, "Dynamic match phone-lattice searches for very fast and accurate unrestricted vocabulary keyword spotting." in *proc. ICASSP*, 2005, pp. 465–468.

[69] D. Vergyri, I. Shafran, A. Stolcke, V. R. R. Gadde, M. Akbacak, B. Roark, and W. Wang, "The sri/ogi 2006 spoken term detection system." in *Proc. Interspeech*, 2007, pp. 2393–2396.

[70] J. Mamou, B. Ramabhadran, and O. Siohan, "Vocabulary independent spoken term detection," in *SIGIR*. ACM, 2007, pp. 615–622.

[71] Y. Zhang and J. R. Glass, "An inner-product lower-bound estimate for dynamic time warping," in *Proc. ICASSP*. IEEE, 2011, pp. 5660–5663.

[72] K. Shobaki, J.-P. Hosom, and R. Cole, "The ogi kids' speech corpus and recognizers," in *Proc. ICSLP, Beijing, China*, 2000.

[73] J. S. Garofolo, L. D. Consortium *et al.*, *TIMIT: acoustic-phonetic continuous speech corpus.* Linguistic Data Consortium, 1993.

[74] H.-G. Hirsch, "Fant-filtering and noise adding tool," *ACM Transactions on Intelligent Systems and Technology*, 2005, software available at http://dnt.kr.hs-niederrhein.de/index964b.html?option=com_content&view=article&id=22&Itemid=15&lang=de.

[75] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The kaldi speech recognition toolkit," in *Proc. ASRU*, Dec. 2011.

[76] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, software available at http://www.csie.ntu.edu.tw/~cjlin/libsvm.

[77] M. Gerosa, D. Giuliani, S. Narayanan, and A. Potamianos, "A review of ASR technologies for children's speech," in *Proc. WOCCI*. ACM, 2009, p. 7.

על מידת ההתאמה של כל היסטוגרמה למילת המפתח. מכאן שמשפט המכיל את מילת המפתח יביא לעקום תגובה בעל ערך מקסימאלי חיובי, בעוד שמשפט שאינו מכיל את מילת המפתח יביא לעקום רועש סביב האפס. אורכו של כל עקום תגובה תלוי במשכו של המשפט המקורי ובאופן החלקת החלון בחישוב ההיסטוגרמות ולכן אינו קבוע. לכן לשם אימון המסווג המבדל למשפטים, כל עקום מיוצג על ידי סדרה קבועה של פרמטרים גלובליים כגון ערכי מינימום ומקסימום, ממוצע וכד', מנורמלים ע"י סטיית התקן הכוללת של ערכי עקום התגובה.

באופן מעשי, ניתן להשיג הקלטות רבות שאינן מכילות את מילת המפתח – המכונות "שליליות", מאשר הקלטות של משפטים המכילים אותה – "חיוביות". לכן, ברוב המקרים הנדונים, סט האימון לא יהיה מאוזן ויהיו בו הרבה פחות דוגמאות חיוביות מאשר שליליות. כדי להימנע מאימון מסווג מוטה, במקום לאמן מסווג יחיד, מאמנים סדרה של מסווגים, שכל אחד מהם מאומן על ידי הסט החיובי ותת־קבוצה הנדגמת אקראית מתוך הסט השלילי. באופן זה ניתן לאזן בין מספר הדוגמאות החיוביות והשליליות בזמן האימון של כל מסווג ובו בעת לנצל את גודלו של הסט השלילי. בזמן הסיווג, מפעילים את סדרת המסווגים ומחליטים לפי הצבעת הרוב. תהליך זה מבוצע במערכת המוצעת הן עבור סיווג המילים והן עבור סיווג המשפטים.

בעבודה מוצגת סדרה של ניסויים, הבוחנים את ביצועי המערכת המוצעת לגילוי מילות מפתח, והמושווים לביצועי מערכת סטנדרטית המבוססת על מודל מרקובי חבוי, כפונקציה של גודלו של סט האימון, ובנוכחות רעשי רקע שונים, כגון, מנוע של רכב או דיבורים ברקע. עבור דיבור של מבוגרים ניכר כי המערכת המוצעת מביאה לביצועים טובים יותר מאשר זו הסטנדרטית, בעיקר במקרים קשים בהם מספר הדוגמאות החיוביות קטן במיוחד, או בנוכחות של רעשי רקע בעוצמה גבוהה. במקרים היותר קלים בהם היה מספר גדול של דוגמאות לאימון ורעש רקע חלש, שתי המערכות הביאו לביצועים טובים מאוד, כאשר המערכת המוצעת היתה מעט פחות טובה. דיבור של ילדים צעירים הינו פחות עקבי מאשר של מבוגרים ולכן מתאפיין באותות בעלי שונות גבוהה. במקרה זה המערכת המוצעת הביאה לביצועים טובים יותר בכל המקרים, ללא תלות במספר הדוגמאות או בעוצמת הרעש.

כמו כן, מוצעת כאן שיטה חדשה להמרת ספקטרום של דובר, הניתנת לאימון בהצלחה בעזרת מסד נתונים קטן. בשיטה זו עוקבים אחר הספקטרום של דובר המטרה בעזרת התהליך הנצפה שהוא הספקטרום של דובר המקור. סט האימון של דובר המטרה משמש סריג המייצג את המרחב הספקטראלי של אות הדיבור של דובר זה. בתהליך ההמרה, הספקטרום של דובר המטרה משוערך בצורה עוקבת, כסכום משוקלל של הנקודות בסריג. המשקלים שחושבו עבור מסגרות הזמן הקודמות נלקחים בחשבון בחישוב המשקלים עבור מסגרת הזמן העכשווית, לקבלת התפתחות זמנית חלקה של הספקטרום. במבחני השמע, משפטים שיוצרו על ידי המערכת המוצעת דורגו כיותר איכותיים מאלה שיוצרו על ידי גישת ההמרה הקלאסית. כמו כן, נבחנו הביצועים של ההמרה המוצעת, יחד עם היחידה הנפרדת להגברת השונות הגלובלית של הרכיבים הספקטראליים, בהשוואה לגישה הקלאסית. ההשוואה לגישה הקלאסית בוצעה גם היא עם הגברת השונות הגלובלית של הרכיבים הספקטרליים בכל אחת משתי השיטות – תוך כדי האימון של פונקצית ההמרה או כיחידה נפרדת, לאחר ההמרה. במקרה זה, המרת הסריג המוגברת הובילה לביצועים הטובים ביותר מבחינת דמיון לדובר המטרה. מבחינת איכות השמע, הגישה הקלאסית עם הגברת השונות הגלובלית בזמן האימון סומנה כטובה ביותר, בעוד שהמרת הסריג המוגברת סומנה כשקולה לגישה הקלאסית יחד עם היחידה הנפרדת.

תהליך האימון של רוב המערכות להמרת דובר מתבסס על מסד נתונים מקבילי בו דובר המקור ודובר המטרה אומרים את אותם המשפטים. בעבודה זו אנו מטפלים גם במצב הלא מקבילי, בו לא מניחים דבר לגבי הטקסט שהוקלט. במקרה זה, יש צורך בהתאמה בין וקטורי האימון של דובר המקור לבין וקטורי האימון של דובר המטרה על מנת לשערך את פונקצית ההמרה. בעבודה זו מוצעת הכללה לגישה קיימת לשערוך עוקב של פונקציות ההמרה וההתאמה. בגישה הקיימת, שערוך ההתאמה מתבצע על ידי חיפוש השכן הקרוב ביותר, המוביל להתאמה בין וקטורים השייכים לרכיבים לשוניים שונים ולכן גם לפגיעה באימון פונקצית ההמרה, ובסופו של דבר לפגיעה גם באיכות האותות המומרים. בגישה המוכללת המוצגת כאן, ההתאמה מתבצעת על ידי חיפוש השכן הקרוב בין סדרות של וקטורים ספקטראליים הלקוחות ממסגרות זמן עוקבות באותות הדיבור המקוריים. התאמה זו משפרת את איכות האותות המומרים הן מבחינת איכות השמע והן מבחינת דמיון לדובר המטרה. בנוסף לכך, בעבודה זו מראים כי השערוך העוקב בגישה המוכללת, ולכן גם במקרה הפרטי, הוא למעשה תהליך של מזעור עוקב המביא למינימום מקומי את פונקצית עלות של פונקציות ההמרה וההתאמה בין הדוברים.

רוב הגישות הקיימות לגילוי מילות מפתח מבוססות על אימון מודל סטטיסטי לתהליך יצירת אות הדיבור. גישות אלה דורשים הקלטות רבות וכן סגמנטציה פונטית של ההקלטות לשם אימון המודל, ולכן אינן מתאימות במקרים בהם סט אימון גדול ומסומן אינו זמין. בעבודה זו מוצעת גישה חדשה לגילוי מילות מפתח, המבוססת על מסווגים למילים ומשפטים. מסווגים אלה מאומנים בשיטה מבדלת (discriminative) שהיא סטנדרטית בלמידה חישובית, הניתנת לאימון גם במקרים של מספר דוגמאות קטן במיוחד, אך דורשת ייצוג באורך קבוע של הדוגמאות. בגישה המוצעת, מסווג המילים מבוסס על ייצוג באורך קבוע של מילים על ידי היסטוגרמות המחושבות ביחס למודל מרובה גאוסייאנים. אמנם לתהליך האימון של מודל הגאוסייאנים עצמו נדרשת כמות בלתי מבוטלת של הקלטות, אבל ניתן להשיג אותן בקלות יחסית כיוון שלא נדרשים שום נתונים או סימונים נוספים פרט להקלטות עצמן. בהינתן סדרה של וקטורים ספקטראליים המתאימים למשפט מסוים, מחשבים סדרה של היסטוגרמות, ביחס למודל הגאוסיאנים, על ידי החלקה של חלון מעט ארוך יותר ממוצע של מילת המפתח. מפעילים את מסווג המילים על כל היסטוגרמה בסדרה ומקבלים סדרה של ציונים, המורים

# תקציר

אימון והרצה של מערכות מודרניות לעיבוד אותות דיבור מצריכים משאבי חישוב וזיכרון בקנה מידה גדול וכן מסדי נתונים המכילים הקלטות רבות של אותות דיבור רלוונטיים. התקדמות הטכנולוגיה הביאה לכך שיחידות העיבוד והזיכרון הדיגיטליות כבר אינן מהוות צוואר בקבוק משמעותי בפיתוח מערכות לעיבוד אותות דיבור. עם זאת, גודלו של סט האימון עדיין מהווה אתגר במקרים כגון שפות שאינן מתועדות היטב, דיבור של ילדים, ויישומים למכשירים סלולאריים, בהם המשתמש המצוי אינו מוכן להשקיע זמן רב בהקלטות של קולו. בעבודה זו נעסוק בשתי מערכות לעיבוד אותות דיבור כאשר סט האימון הוא קטן במיוחד: מערכת המרת דובר, בה משפט שנאמר על ידי דובר מקור מומר כך שיישמע כאילו נאמר על ידי דובר מטרה, ומערכת לזיהוי מילות מפתח בה בהינתן משפט, נדרש לומר האם נאמרה בו מילת המפתח או לא.

נהוג לייחס את התפיסה הסובייקטיבית של זהות הדובר למאפיינים הספקטראליים של אות הדיבור וכן לתדר המרכזי, לקצב הדיבור ולעוצמתו. רוב המערכות להמרת דובר מתרכזות בהמרה של הספקטרום, בעוד שאת עקום התדר המרכזי הן ממירות בעזרת המרה ליניארית פשוטה כך שממוצע וסטיית התקן של התדר המרכזי של האות המומר יהיו שווים לאלה של דובר המטרה, וקצב הדיבור ועוצמתו בדרך כלל לא מומרים כלל.

שיטות קיימות להמרת ספקטרום של דובר משתמשות ברובן במודלים סטטיסטיים על מנת לאמן פונקצית המרה. השיטה הקלאסית מבוססת על אימון של המרה ליניארית בעזרת מודל מרובה גאוסייאנים. לאימון זה נדרשות עשרות רבות של משפטים מוקלטים. תהליך ההמרה מתבצע ללא תלות בין מסגרות זמן עוקבות, והמשפטים המתקבלים נשמעים בדרך כלל עמומים עקב עודף החלקה של המעטפות הספקטראליות.

על מנת לשפר את האיכות של האותות המומרים לאחר הסינטזה, מוצגות כאן שתי שיטות להגברת השונות הגלובלית של הרכיבים הספקטראליים של אותות מומרים. השיטה הראשונה מבוססת על המערכת הקלאסית־ליניארית להמרת דובר, כאשר בזמן האימון מאלצים את השונות הגלובלית של רכיבי הספקטרום של האות המומר להיות שווה לשונות הגלובלית המתאימה בדובר המטרה. בשיטה זו הגברת השונות הגלובלית נעשית על ידי אימון מחודש של פונקציה ההמרה. השיטה השנייה מבוססת על יחידת עיבוד נפרדת המופעלת לאחר ההמרה עצמה, ולכן מתאימה למערכות בהן נדרש לשפר את איכות השמע של האותות המומרים מבלי לשנות את מערכת ההמרה עצמה. בהינתן אות דיבור מומר, היחידה הנפרדת שואפת להגביר את השונות הגלובלית של הרכיבים הספקטרליים, כך שהמרחק הספקטראלי בין האות המוגבר והאות המומר לא יעלה על סף שהגדיר המשתמש. לפי מבחני השמע שנערכו, שתי השיטות מביאות לשיפור איכות השמע של האותות המומרים, בהשוואה לגישת ההמרה הקלאסית.

המחקר בוצע בהנחייתם של פרופ' דוד מלאך ופרופ' קובי קרמר בפקולטה להנדסת חשמל.

חלק מן התוצאות בחיבור זה פורסמו כמאמרים מאת המחבר ושותפיו למחקר בכנסים ובכתבי־עת במהלך תקופת מחקר הדוקטורט של המחבר, אשר גרסאותיהם העדכניות ביותר הינן:

- H. Benisty and D. Malah, “Voice Conversion Using GMM with Enhanced Global Variance.” *in Proc. INTERSPEECH*, 2011, pp 669-672.
- H. Benisty, D. Malah, and K. Crammer, “Modular global variance enhancement for voice conversion systems.” *in Proc. EUSIPCO*, 2012, pp. 370-374.
- H. Benisty, D. Malah, and K. Crammer, “Non-parallel voice conversion using joint optimization of alignment by temporal context and spectral distortion.” *in Proc. ICASSP*, 2014, pp. 7909-7913.
- H. Benisty, D. Malah, and K. Crammer, “Sequential voice conversion using grid-based approximation.” *in Proc. IEEEI*, 2014.
- H. Benisty, D. Malah, and K. Crammer, “Grid-Based Approximation For Voice Conversion In Low Resource Environments”, *EURASIP Journal on Audio, Speech, and Music Processing*, Jan. 2016.
- H. Benisty, K. Crammer, D. Malah, K. Crammer, and I. Kats, “Discriminative Keyword Spotting For Low Resource Applications”, *to be submitted after submission of a patent*, 2015.

## תודות

ברצוני להודות לפרופסור דוד מלאך ולפרופסור קובי קרמר על הנחייתם המסורה. אני מרגישה בת מזל שזכיתי ללמוד מהם הן ברמה המקצועית והן ברמה האישית. תודה רבה על ההכוונה, העצות הטובות, השאיפה לשלמות ובעיקר על הדחיפה קדימה, גם כשחשבתי לוותר.

תודה לצוות המעבדה לעיבוד אותות ותמונות: נמרוד פלג, יאיר משה, זיוה אבני ואבי רוזן שמייצרים סביבת עבודה מעולה ואוירה משפחתית ונעימה.

ברצוני להודות לחברים שזכיתי להכיר בטכניון: רונן, מירי, איתמר, תמר ונורית. תודה על השיחות והעצות המקצועיות והאישיות כאחד ועל כך שהפכו את השהות בטכניון למהנה כל כך.

לבסוף, ברצוני להודות לבעלי היקר והמתוק שהאמין בי עוד בתחילת התואר הראשון והמשיך להאמין ולתמוך בי עד לסיום התואר השלישי. אהבתך היא מקור כוחי. תודה מיוחדת למשפחתי האהובה: להוריי איתן וברוריה, לאחיי: תומר ואלון, על העזרה תמיכה והאהבה שהעניקו לי.

תודה לחברותי המקסימות שהן כמו משפחה עבורי: נגה, טלי, אלה, מיכל וליזו על תמיכתן ואהבתן.

אחרונים והכי חביבים, תודה לילדיי האהובים והמתוקים: טליה, דניאל ורעות, שמעניקים לי יום יום אהבה אינסופית, שמחה ונחת.

אני מודה לטכניון על התמיכה הכספית הנדיבה בהשתלמותי. חלק מעבודת המחקר נעשתה בתמיכתם של משרד המדע הכלכלה ישראלי, מענק 06305, ומשרד המדע הישראלי, מענק ISF - 1567/10

# שיטות סטטיסטיות לעיבוד אותות דיבור במערכות דלות משאבים

חיבור על מחקר

לשם מילוי חלקי של הדרישות לקבלת התואר
דוקטור לפילוסופיה

**הדס בן אייסטי**

הוגש לסנט הטכניון – מכון טכנולוגי לישראל
שבט ה'תשע"ו חיפה ינואר 2016

# שיטות סטטיסטיות לעיבוד אותות דיבור
# במערכות דלות משאבים

הדס בן איסטי