

SPEAKER TRANSFORMATION USING SENTENCE HMM BASED ALIGNMENTS AND DETAILED PROSODY MODIFICATION

Levent M. Arslan

Entropic Research Laboratory, Washington, DC, 20003

ABSTRACT

This paper presents a new scheme for developing a voice conversion system that modifies the utterance of a source speaker to sound like speech from a target speaker. We refer to the method as Speaker Transformation Algorithm using Segmental Codebooks (STASC). Two new methods are described to perform the transformation of vocal tract and glottal excitation characteristics across speakers. In addition, the source speaker's general prosodic characteristics are modified using time-scale and pitch-scale modification algorithms. Informal listening tests suggest that convincing voice conversion is achieved while maintaining high speech quality. The performance of the proposed system is also evaluated on a standard Gaussian mixture model based speaker identification system, and the results show that the transformed speech is assigned higher likelihood by the target speaker model when compared to the source speaker model.

1 Introduction

There has been a considerable amount of research effort directed at the problem of voice transformation recently [?, ?, ?, ?, ?]. This topic has numerous applications which include personification of text-to-speech systems, multimedia entertainment, and as a preprocessing step to speech recognition to reduce speaker variability. In general, the approach to the problem consists of a training phase where input speech training data from source and target speakers are used to formulate a parametric spectral transformation that would map the acoustic space of the source speaker to that of the target speaker. The transformation is in general based on codebook mapping [?, ?, ?]. That is, a one to one correspondence between the spectral codebook entries of the source speaker and the target speaker is developed by some form of supervised vector quantization method. It is crucial for the success of the mapping to have good alignments between source and target speaker speech. Normally, a phonetic alignment or dynamic time warping algorithm is applied to extract the corresponding speech units from source and target talkers. In this paper, we are introducing a new method for the alignment process using sentence HMMs. The method also adapts to speakers voices with an iterative scheme which results in extremely high quality alignments. Using this method, we were able to improve the quality of our system significantly when compared to our previous approach of using phonetic alignments.

2 Algorithm Description

This section provides a general description of the STASC algorithm. We will describe the algorithm under two main sections: i) transformation of spectral characteristics, ii) transformation of prosodic characteristics.

2.1 Spectral Transformation

For the representation of the vocal tract characteristics of the source and target speakers line spectral frequencies are se-

lected. The reason for selecting line spectral frequencies is that these parameters relate closely to formant frequencies [?], but in contrast to formant frequencies they can be estimated quite reliably. They have been used for a number of applications successfully in the literature [?, ?, ?, ?, ?] In addition, they have a fixed dynamic range which makes them attractive for real-time DSP implementation. In STASC algorithm codebooks of line spectral frequencies are used to represent the vocal tract characteristics of individual speakers. The codebooks can be generated in two ways.

The first method assumes that the orthographic transcription is available along with the training data. The training speech (sampled at 16 kHz) from source and target speakers are first segmented automatically using forced alignment to a phonetic translation of the orthographic transcription. The segmentation algorithm uses Mel-cepstrum coefficients and delta coefficients within an HMM framework and is described in detail in [?]. The line spectral frequencies for source and target speaker utterances are calculated on a frame-by-frame basis and each LSF vector is labeled using the phonetic segmenter. Next, a centroid LSF vector for each phoneme is estimated for both source and target speaker codebooks by averaging across all the corresponding speech frames. A one-to-one mapping is established from the source and target codebooks to accomplish the voice transformation.

The second method does not require the phonetic translation of the orthographic transcription for the training utterances, however it assumes that both the source and target speakers are speaking the same sentences. In this case, short template sentences are selected which are phonetically balanced to be uttered by the source and target speakers. After the training data is collected, the silence regions at the beginning and end of each utterance is removed. Each utterance is normalized in terms of its RMS energy to account for differences in the recording gain level. Next, cepstrum coefficients are extracted along with log-energy and zero-crossing for each analysis frame in each utterance. Zero-mean normalization is applied to the parameter vector to obtain a more robust spectral estimate. Based on the parameter vector sequences, sentence HMMs are trained for each template sentence using data from source and target speakers. The number of states for each sentence HMM is set proportional to the number of phonemes in the phonetic representation. The training is done using segmental k-means algorithm followed by Baum-Welch algorithm. The initial covariance matrix is estimated over the complete training dataset, and is not updated during the training since the amount of data corresponding to each state is not sufficient to make a reliable estimate of the variance. Next, the best state sequence for each utterance is estimated using Viterbi algorithm. The average LSF vector for each state is calculated both for the source and target speakers us-

ing the frame vectors corresponding to the state index. Finally these average LSF vectors for each sentence are collected to build the source and target speaker codebooks. In Figure ??, the alignments to the state indices are shown for the sentence “She had your dark suit in greasy wash water all year” both for the source and target speaker utterances. From the figure, it can be observed that very detailed acoustic alignment is performed very accurately using sentence HMMs. The transformation will be explained in detail later in this section.

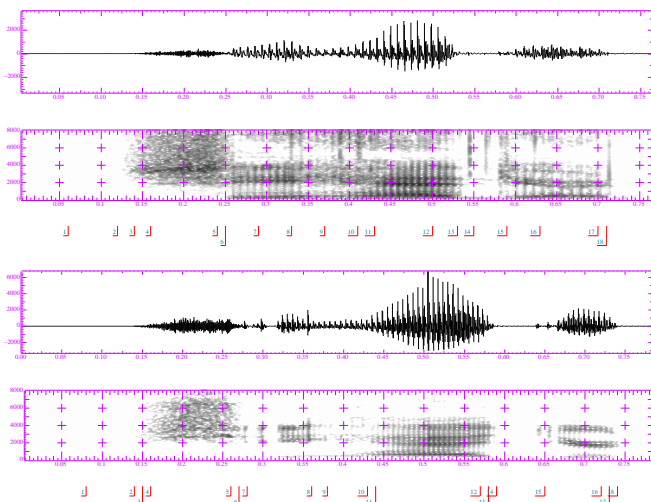


Figure 1: The state alignments for source and target speaker utterances “She had your”.

Another factor that influences speaker individuality is glottal excitation characteristics. The LPC residual can be a reasonable approximation to the glottal excitation signal. It is well known that the residual can be very different for different phonemes (e.g., periodic pulse train for voiced sounds versus white noise for unvoiced sounds). Therefore, we formulated a “codebook based” transformation of the excitation characteristics similar to the one discussed above for vocal tract spectrum transformation. Codebooks for excitation characteristics are obtained as follows: Using the segmentation information, the LPC residual signals for each phoneme in the codebook are collected from the training data. Next, a short-time average magnitude spectrum of the excitation signal is estimated for each phoneme both for the source speaker and the target speaker pitch synchronously. An excitation transformation filter can be formulated for each codeword entry using the excitation spectra of the source speaker and the target speaker. This method not only transforms the excitation characteristics, but it estimates a reasonable transformation for the “zeros” in the spectrum as well, which are not represented accurately by the all-pole modeling. Therefore, this method resulted in improved voice conversion performance especially for nasalized sounds.

The flow diagram of the STASC voice transformation algorithm is shown in Figure ??. The incoming speech is

first sampled at 16 kHz and preemphasized with the filter $P(z) = 1 - 0.95z^{-1}$. Next, 18th order LPC analysis is performed to estimate the prediction coefficients vector \mathbf{a} . Based

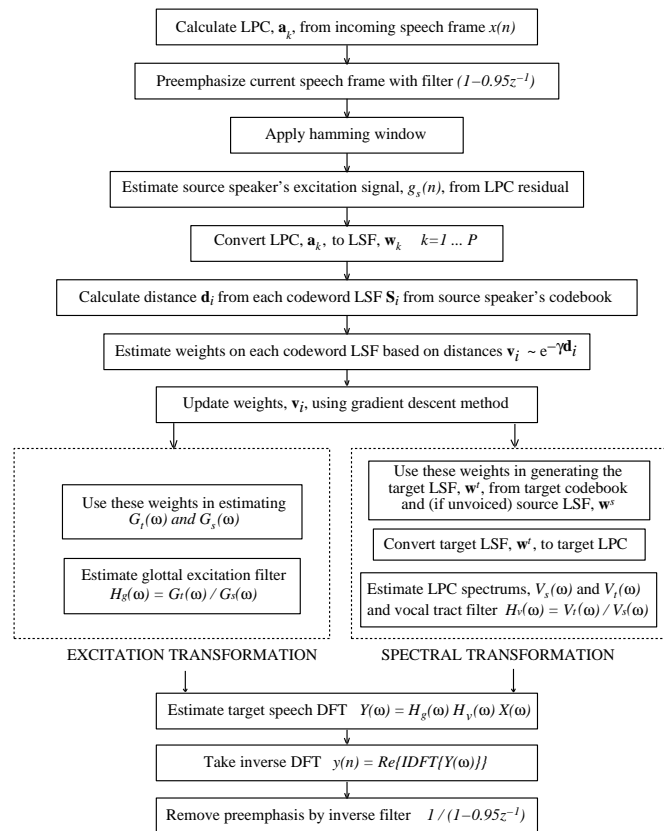


Figure 2: Flow-diagram of STASC voice conversion algorithm.

on the source-filter theory, the incoming speech spectrum $X(\omega)$ can be represented as

$$X(\omega) = G_s(\omega)V_s(\omega), \quad (1)$$

where $G_s(\omega)$ and $V_s(\omega)$ represent source speaker glottal excitation and vocal tract spectrums respectively for the incoming speech frame $x(n)$.

The target speech spectrum $Y(\omega)$ can be formulated as:

$$Y(\omega) = \left[\frac{G_t(\omega)}{G_s(\omega)} \right] \left[\frac{V_t(\omega)}{V_s(\omega)} \right] X(\omega) \quad (2)$$

where $V_t(\omega)$ and $G_t(\omega)$ represent codebook estimated target vocal tract and glottal excitation spectrums respectively. This representation of the target spectrum can be thought of as an excitation filter followed by a vocal tract filter. In the proposed algorithm, the source speaker vocal tract spectrum $V_s(\omega)$ is estimated differently for voiced and unvoiced segments. For voiced segments, in general, the LSF codebook representation can provide a good approximation to the original vocal tract spectrum. Therefore in the above formulation, $V_s(\omega)$ can be

replaced with the spectrum derived from the original LPC vector \mathbf{a} :

$$V_s(\omega) = \frac{1}{1 - \sum_{k=1}^P \mathbf{a}_k e^{-jk\omega}}. \quad (3)$$

However, for unvoiced segments this is not true especially when there are imperfections in the segmentations and when the codebook size is small. In such cases, it is extremely difficult to accurately represent the vocal tract spectrum for unvoiced sections based on the codebook. This leads to a mismatch in the vocal tract filter formulation. In order to provide a reasonable balance in the filter formulation between source and target spectra it becomes necessary to use the LPC vector \mathbf{a} derived from the codebook weighted LSF vector approximation $\tilde{\mathbf{w}}_k$

$$\tilde{\mathbf{w}}_k = \sum_{i=1}^L \mathbf{v}_i \mathbf{S}_{ik} \quad k = 1, \dots, P \quad (4)$$

where \mathbf{S}_i is the i^{th} codeword LSF vector and \mathbf{v}_i represents its weight. For both formulations, the codebook weights need to be estimated for the target spectrum estimate $V_t(\omega)$. The codebook weight estimation procedure is as follows.

Codebook Weight Estimation Method

First, line spectral frequencies, \mathbf{w} , are derived from the prediction coefficients. Line spectral frequency vector \mathbf{w} is compared with each LSF centroid, \mathbf{S}_i , in the source codebook and the distance, \mathbf{d}_i , corresponding to each codeword is calculated. The distance calculation is based on a perceptual criterion where closely spaced line spectral frequencies which are likely to correspond to formant locations are assigned higher weights [?],

$$\mathbf{h}_k = \frac{1}{\text{argmin}(|\mathbf{w}_k - \mathbf{w}_{k-1}|, |\mathbf{w}_k - \mathbf{w}_{k+1}|)} \quad k = 1, \dots, P$$

$$\mathbf{d}_i = \sum_{k=1}^P \mathbf{h}_k |\mathbf{w}_k - \mathbf{S}_{ik}| \quad i = 1, \dots, L \quad (5)$$

where L is the codebook size. In addition to the above weighting, for voiced segments lower order LSFs, and for unvoiced segments higher order LSFs are weighted more by an exponential weighting factor. Based on the distances from each codebook entry, an expression for the normalized codebook weights can be obtained as [?]:

$$\mathbf{v}_i = \frac{e^{-\gamma \mathbf{d}_i}}{\sum_{l=1}^L e^{-\gamma \mathbf{d}_l}} \quad i = 1, \dots, L \quad (6)$$

where the value of γ for each frame is found by an incremental search with the criterion of minimizing the perceptual weighted distance between the approximated LSF vector $\tilde{\mathbf{w}}$ and original LSF vector \mathbf{w} . However this set of weights may still not be the optimal set of weights that would represent the original speech spectrum. In order to improve the estimate of weights a gradient descent algorithm is employed [?].

Glottal Excitation Spectrum Mapping

The estimated set of codebook weights can be regarded as information about the phonetic content of the current speech frame. It can be utilized in two separate domains: i) transformation of the glottal excitation characteristics, ii) transformation of the vocal tract characteristics. For transformation of

the glottal excitation, the set of weights is used to construct an overall filter which is a weighted combination of excitation codeword filters:

$$H_g(\omega) = \sum_{i=1}^L \mathbf{v}_i \frac{\mathbf{U}_i^t(\omega)}{\mathbf{U}_i^s(\omega)} \quad (7)$$

where $\mathbf{U}_i^t(\omega)$ and $\mathbf{U}_i^s(\omega)$ denote average target and source excitation spectra for the i^{th} codeword respectively.

Vocal Tract Spectrum Mapping

The same set of codebook weights (\mathbf{v}^i , $i = 1, \dots, L$) are applied to target LSF vectors (\mathbf{T}_i , $i = 1, \dots, L$) to construct the target line spectral frequency vector $\tilde{\mathbf{w}}^t$:

$$\tilde{\mathbf{w}}_k^t = \sum_{i=1}^L \mathbf{v}_i \mathbf{T}_{ik}, \quad k = 1, \dots, P \quad (8)$$

Next, target line spectral frequencies are converted to prediction coefficients, \mathbf{a}^t , which in turn are used to estimate the target LPC vocal tract filter:

$$V_t(\omega) = \left| \frac{1}{1 - \sum_{k=1}^P \mathbf{a}_k e^{-jk\omega}} \right|^{\frac{1}{2}}. \quad (9)$$

The weighted codebook representation of the target spectrum results in expansion of formant bandwidths. In order to cope with this problem a new bandwidth modification algorithm is used and is described in [?].

Combined Output

The vocal tract filter and glottal excitation filters are next applied to the magnitude spectrum of the original signal to get an estimate of the DFT corresponding to the preemphasized target speech:

$$Y(\omega) = H_g(\omega) \frac{V_t(\omega)}{V_s(\omega)} X(\omega). \quad (10)$$

Next, inverse DFT is applied to produce the synthetic target voice,

$$y(n) = \text{Real}\{\text{IDFT}\{Y(\omega)\}\}. \quad (11)$$

Finally preemphasis is removed from the speech by applying inverse preemphasis filter:

$$P^{-1}(z) = \frac{1}{1 - 0.95z^{-1}}. \quad (12)$$

2.2 Prosodic Transformation

In STASC algorithm a frequency domain pitch synchronous analysis synthesis framework is adopted in order to be able to realize both spectral and prosodic transformations simultaneously. In addition to the spectral transformation discussed in the previous section pitch, duration, and amplitude is modified to mimic target speaker prosodic characteristics. Each analysis frame length is set to be constant for unvoiced regions. For voiced regions the frame length is set to two or three pitch periods depending on the pitch modification factor. It is observed that when the pitch modification factor

is less than one using smaller frame lengths reduces artifacts introduced by the modification.

Pitch-Scale Modification

The pitch modification involves matching both the average pitch value and range for the target speaker. This is accomplished by modifying the source speaker fundamental frequency, f_0^s , by a multiplicative constant a and an additive constant b :

$$f_0^t = af_0^s + b \quad (13)$$

The value for a is set so that the source speaker pitch variance σ_s^2 , and target speaker pitch variance σ_t^2 match, i.e.,

$$a = \sqrt{\frac{\sigma_t^2}{\sigma_s^2}} \quad (14)$$

Once the value for a is set, the value for the additive constant b can be found by matching the average f_0 values.

$$b = \mu_t - a\mu_s \quad (15)$$

where μ_s and μ_t represent source and target mean pitch values. Therefore, the pitch scale modification factor β at each frame can be set as

$$\beta = \frac{af_0^s + b}{f_0^s} \quad (16)$$

in order to achieve the desired target speaker pitch value and range.

Duration-Scale Modification

The duration characteristics can vary across different speakers significantly due to a number of factors including accent or dialect. Although modifying the speaking rate uniformly to match the target speaker duration characteristics reduces timing differences between speakers to some extent, it is observed that this is not sufficient in general. In Figure ??, comparison of duration statistics of monophones for two speakers in our database are given. It can be seen from the table that the proportion of average durations are quite different among different phonemes. For example, the average duration of /aa/ vowel is 100 ms for source speaker, and 67 ms for target speaker. On the other hand, for the /uh/ vowel the target speaker has a longer average duration (64 ms versus 37 ms). Although on the average the target speaker has 1.2 times longer average duration than the source speaker, there exists a significant number of phonemes that the target speaker uses shorter duration for.

Based on the previous set of results it can be concluded that the variation in duration characteristics between two speakers is heavily dependent upon context. Therefore it is highly desirable to develop a method for automatically estimating the appropriate time-scale modification factor in a certain context. In STASC algorithm a codebook based approach to duration modification is implemented. The phonetic codebooks used for spectral mapping can also be used to generate the appropriate duration modification factor for a given speech frame. In order to accomplish this, first duration statistics are estimated for both the source speaker and the target speaker

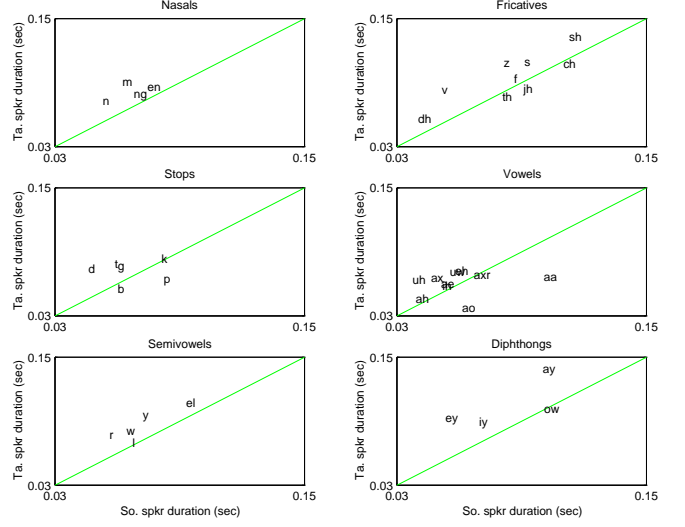


Figure 3: Comparison of duration statistics between a source speaker and a target speaker.

for all the phonemes in the codebook. Then the same codebook weights developed for spectral mapping can be used to estimate the appropriate time-scale modification factor γ :

$$\gamma = \sum_{i=1}^L \mathbf{v}_i \frac{d_i^t}{d_i^s}, \quad (17)$$

where d_i^t and d_i^s represent average source and target speaker durations for the i^{th} phone in the codebook.

A major application for current time-scale modification algorithms is to slow down the speech for accurate transcription by humans. The problem with most of those systems is that they use a constant time scale modification factor when changing the speaking rate. However, not all the phonemes are scaled to the same extent when a speaker modifies his/her speaking rate. Therefore, the same approach proposed here for transforming duration characteristics across speakers can be applied to speaking rate modification algorithms if the statistics for slow, normal and fast speaking styles are generated prior to the application.

Stress Modification In addition to pitch and duration, stress is another important component which characterizes the prosody of a speaker. In order to match target speaker's stress characteristics we applied a codebook based amplitude mapping as well. The RMS energy is scaled with a variable η at each time frame. The scaling factor can be expressed as follows:

$$\gamma = \sum_{i=1}^L \mathbf{v}_i \frac{e_i^t}{e_i^s}, \quad (18)$$

where e_i^t and e_i^s represent average source and target speaker energies for the i^{th} phone in the codebook.

Finally, the pitch-scale modification factor β , the time-scale modification factor γ , and energy scaling factor η are

used to perform prosodic modification with pitch-synchronous overlap-add synthesis.

The next section discusses the evaluations conducted to test the performance of the STASC algorithm.

3 Evaluations

In order to evaluate the performance of the STASC algorithm we performed a subjective listening experiment. While informal listening tests showed that the transformation of speaker characteristics was successful, we wanted to test whether the transformation process introduced a degradation in intelligibility. This was necessary, since the most important application (i.e., text to speech personafication) relies heavily on the level of intelligibility. The test material was 150 short nonsense sentences. For example One of the sentences used in the test was “Shipping gray paint hands even”. The main purpose of using nonsense sentences was to limit the ability of the listener to derive words from context. Two conditions, transformed speech and natural speech, were presented to the listeners with random order. We used three inexperienced listeners to transcribe the test material. Listeners were allowed to listen each sentence up to three times. The transformation tested in this experiment was from a male speaker to another male speaker. The result of the experiment was surprising. The phone accuracy for natural speech (93.4%) was slightly lower than it was for the transformed speech (93.8%). The reason for the slight increase in intelligibility might be due to measurement noise. Another possible reason might be that the target speaker was more intelligible than the source speaker, and the transformation algorithm took advantage of that. Of course, the transformation between different speaker combinations may reveal different results. When the acoustic characteristics of two speakers are extremely different (e.g., male to female transformation), then we may expect degradation in intelligibility. Our future plans include testing other speaker conditions.

4 Conclusion

In this study, several improvements to our previous voice conversion system are described. First a new concept, sentence HMM, is introduced to refine the alignments between source and target speaker utterances. Sentence HMMs can provide more robust and finer detail alignments when compared to previous methods using DTW or phonetic alignments. In addition they have the advantage of being vocabulary independent over phonetic alignment method that we used in our previous system.

In terms of prosodic characteristics, the previous algorithm was only adjusting mean pitch level and speaking rate. Now, in addition to mean pitch level the pitch range is adjusted to match the target talker. Moreover, codebook based duration and energy modifications are performed to capture context dependent prosodic characteristics. The enhancements to algorithm resulted in better characterization of the target speaker speech. Finally, subjective tests verified that additional processing did not introduce degradation in intelligibility scores for the transformed speech.

References

- [1] M. Abe, S. Nakamura, K. Shikano, and H. Kuwabara. “Voice Conversion through Vector Quantization”. In *Proc. IEEE ICASSP*, pages 565–568, 1988.
- [2] L.M. Arslan, A. McCree, and V. Viswanathan. “New Methods for Adaptive Noise Suppression”. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, volume 1, pages 812–815, Detroit, USA, May 1995.
- [3] L.M. Arslan and D. Talkin. “Voice Conversion by Codebook Mapping of Line Spectral Frequencies and Excitation Spectrum”. In *Proc. EUROSPEECH*, volume 3, pages 1347–1350, Rhodes, Greece, September 1997.
- [4] G. Baudoin and Y. Stylianou. “On the transformation of the speech spectrum for voice conversion”. In *Proceedings ICSLP*, pages 1405–1408, Philadelphia, USA, 1996.
- [5] D.G. Childers. “Glottal source modelling for voice conversion”. *Speech Communication*, 16(2):127–138, February 1995.
- [6] J.R. Crosmer. *Very low bit rate speech coding using the line spectrum pair transformation of the LPC coefficients*. PhD thesis, Elec. Eng., Georgia Inst. Technology, 1985.
- [7] J.H.L. Hansen and M.A. Clements. “Constrained iterative speech enhancement with application to speech recognition”. *IEEE Trans. on Signal Processing*, 39(4):795–805, 1991.
- [8] F. Itakura. “Line spectrum representation of linear prediction of speech signals”. *J. Acoust. Soc. Amer.*, 57(S35(A)), 1975.
- [9] N. Iwahashi and Y. Sagisaka. “Speech spectrum conversion based on speaker interpolation and multi-functional representation with weighting by radial basis function networks”. *Speech Communication*, 16(2):139–151, February 1995.
- [10] H. Kuwabara and Y. Sagisaka. “Acoustic characteristics of speaker individuality: Control and conversion”. *Speech Communication*, 16(2):165–173, February 1995.
- [11] R. Laroia, N. Phamdo, and N. Farvardin. “Robust and Efficient Quantization of Speech LSP Parameters Using Structured Vector Quantizers”. In *Proc. IEEE ICASSP*, pages 641–644, 1991.
- [12] K.S. Lee, D.H. Youn, and I.W. Cha. “A new voice transformation method based on both linear and nonlinear prediction analysis”. In *Proceedings ICSLP*, pages 1401–1404, Philadelphia, USA, 1996.
- [13] C. Wightman and D. Talkin. *The Aligner User’s Manual*. Entropic Research Laboratory, Inc., Washington, DC, 1994.