Transforming Pitch in a Voice Conversion Framework

Zeynep Inanoglu

St.Edmund's College
University of Cambridge
July 23, 2003

Submitted in partial fulfillment of the requirements for the degree of Master of Philosophy.

Abstract

A subtask of voice conversion is to accurately map the pitch contour of a source speaker to a target speaker. So far, the most widely employed method for carrying out this mapping is based on adjusting the pitch range of the source speaker to match the target while keeping the shape of the contour unchanged. In this project, we investigate four alternative algorithms for pitch contour mapping and compare their performance with the popular baseline method in an objective framework as well as through perceptual tests. The first two methods extend the baseline to allow more complex mappings of the pitch range, while the last two methods aim to impart an entirely new contour onto the target. We have found that all four methods improve the baseline for most cases. The amount of improvement, however, varies from method to method and manifests a dependency on the nature of the training data available.

Declaration

This thesis is substantially my own work. References to other sources have been clearly specified in the text and in the bibliography. The source code can be found under ∼zi201/code.

Acknowledgements

I would like to thank my supervisors, Tina Burrows and Steve Young, for responding so quickly to all my questions. Many thanks to Jay Silver and Daniel Andor for taking the time to record the utterances for the speech corpus and to all those who participated in the perceptual tests. Also special thanks to Hui Ye for providing me with the details of his pitch synchronous harmonic model as well as the code for analysis and synthesis of speech signals.

Contents

1 Introduction **5**
- 1.1 Motivation . . . 5
- 1.2 Summary of Algorithms Presented . . . 5
- 1.3 Outline . . . 6

2 Background **7**
- 2.1 What is a pitch contour? . . . 7
- 2.2 Previous Work on Pitch Modelling and Transformation . . . 8
  - 2.2.1 Mean/Variance Linear Transformation . . . 8
  - 2.2.2 Further Work on Pitch Contour Modelling and Conversion . . . 9

3 Speech Corpus **11**
- 3.1 Speakers and Text Material . . . 11
- 3.2 Pitch Extraction . . . 12
- 3.3 Transcriptions . . . 13

4 Experimental Setup **14**
- 4.1 Analysis and Synthesis . . . 14
  - 4.1.1 Frame Boundary Decisions . . . 16
  - 4.1.2 Pitch Synchronous Harmonic Model . . . 16
- 4.2 Pitch Transplantation System . . . 17
  - 4.2.1 Motivation for Pitch Transplantation . . . 17
  - 4.2.2 Phonetic Time Alignment of Utterances . . . 17
  - 4.2.3 Pitch Scaling Using Pitch Synchronous Harmonic Model . . . 20

5 Pitch Conversion Algorithms **22**
- 5.1 Baseline Mean-Variance Linear Conversion . . . 22
- 5.2 Nth Order Conversion Function . . . 24
  - 5.2.1 Training . . . 25
  - 5.2.2 Conversion . . . 26
- 5.3 GMM Conversion Function . . . 28
  - 5.3.1 Brief Introduction to Gaussian Mixture Models . . . 28
  - 5.3.2 Training . . . 29
  - 5.3.3 Conversion . . . 32
- 5.4 Codebook of Utterance Contours . . . 32
  - 5.4.1 Training . . . 32
  - 5.4.2 Conversion . . . 33
- 5.5 Codebook of Voiced Segments . . . 37
  - 5.5.1 Training . . . 37
  - 5.5.2 Conversion . . . 37

6 Evaluation and Results **41**
- 6.1 Objective Evaluation . . . 41
- 6.2 Perceptual Evaluation . . . 46
  - 6.2.1 Similarity Test . . . 46

6.2.2 Preference Test 48

7 Conclusion **51**

A Recorded Text Material **54**

List of Figures

1 Two contours for "Did you buy corduroy overalls?" generated by two distinct speakers 7
2 Similarity of five male OGI speaker contours for one utterance. 12
3 Speech signal and laryngograph signal for the phrase "outdoors" 13
4 Overview of experimental setup. 15
5 Female and male speaker's utterance of "Would a tomboy often play outdoors?" a) without time-alignment b)female speaker's contour time-aligned with male speaker's contour. 18
6 Male speaker's pitch contour for "Did you buy any corduroy overalls?" a)without interpolation. b) with interpolation 19
7 Time alignment of voiced frames for the phone '/aw/'. 20
8 Baseline conversion for "Challenge each general's intelligence" a)source contour in blue, converted contour in red. b)original target contour in blue, converted time-aligned contour in red 23
9 Baseline conversion for "Trish saw hours and hours of movies Saturday" a)source contour in blue, converted contour in red. b)original target contour in blue, converted time-aligned contour in red 23
10 Baseline conversion for "Trish saw hours and hours of movies on Saturday" in the case of non-mimicked recordings a)source contour in blue, converted contour in red. b)original target contour in blue,converted time-aligned contour in red. 24
11 Mean pitch values for a male source speaker and a female target speaker from the OGI data. The cubic approximation is illustrated in blue and the linear mean-variance conversion function in green. 26
12 Mean pitch values for two male Cambridge speakers. 27
13 Comparison with baseline for "The rose corsage smelled sweet" for a female to male conversion. a)Input contour in blue, cubic conversion in green. b) Target contour in blue, baseline conversion in red, cubic conversion in green. 27
14 Comparison with baseline for "Trish saw hours and hour of movies on Saturday" 28
15 Two Gaussian mixtures produced by male Cambridge speaker. 29
16 Training Data for Cambridge male to male conversion - baseline conversion function in green, GMM conversion function in red. 31
17 Conversion results for "The rose corsage smelled sweet": converted contour after pitch transplantation in green, true target contour in blue. Note similarity with Figure 13. 32
18 High level flow diagram of the codebook method. 33

19 Source and target codebook entries for training utterance "Did you buy any corduroy overalls?" 35
20 Codebook conversion for training utterance "Did you buy any corduroy overalls?" a) Input contour in blue. Projected interpolated target contour in green. b) Result of pitch transplantation: original target contour in blue, codebook based conversion in green, baseline conversion in red. . . 35
21 Codebook conversion from a male to a female speaker for "Trish saw hours and hours of movies on Saturday." Original target contour in blue, codebook based conversion in green, baseline conversion in red. 36
22 Codebook conversion from a female to male speaker for "Will you go to the may ball?" a) Input contour of female speaker(blue) and interpolated source codebook entries for female speaker (green) b)Target contour (blue), baseline conversion (red), code book conversion (green) 36
23 Extraction of codebook voiced segments from source and target training data in the case of close neighboring voiced sections. 38
24 Conversion from a male speaker to a female speaker for "The oasis was a mirage". Original target contour (blue), utterance codebook (green), VS codebook (magenta) 39
25 Conversion from a female speaker to a male speaker for "Will you go to the may ball?" a)Input contour (blue), concatenation of best source voiced segments (magenta). b)Original target contour (blue), utterance codebook (green), VS codebook (magenta) 39
26 a) Cubic conversion function(red) and baseline conversion function (green) for a male to female conversion. b)GMM conversion function (red) and baseline conversion function (green). 42
27 Screenshot of similarity test. 46
28 Screenshot of preference test. 48

1 Introduction

Voice conversion algorithms aim to modify the utterance of a source speaker to sound as if it was uttered by a target speaker. Since speaker identity consists of a complex weave of factors including short-term spectral characteristics, prosody and linguistic style, it is important to transform each such factor as successfully as possible. There have been a substantial amount of work in the literature that focuses on the conversion of spectral parameters which are related to the timbre, i.e. how the voice itself sounds. On the other hand, the widely employed methods for modelling and converting speaker dependent prosody are still rather simplistic. The goal of this project is to explore pitch conversion, which falls under the more general area of prosodic transformation. Four methods for converting the pitch contour of one speaker to another have been investigated and their relative performance is evaluated in an objective framework as well as through perceptual tests. The ultimate objective is to examine the extent to which these methods of pitch conversion constitute an improvement over the simplest conversion scheme that has been widely used in the literature.

1.1 Motivation

Pitch is arguably the most expressive manifestation of speaker-dependent prosody, which also includes factors such as phone duration, loudness and pause locations[15]. For instance, different speakers have different pitch ranges (e.g.women's mostly higher than men's), which can be represented by calculating the mean pitch and pitch variance for each speaker. In fact, the simplest and most widely used way of converting one speaker's pitch contour to another is to modify pitch values on a frame by frame basis using a linear transformation based on the mean and variance of each of the speakers.[1][2][3]

However, it turns out that pitch range is not the only speaker specific information that needs to be extracted and transformed in a voice conversion framework. The intonation of an utterance is also represented by variations in pitch pattern. Changing pitch throughout an utterance is usually the most powerful way of expressing emotion or emphasis based on the meaning of the message [15]. Different speakers may utter the same sentence with different intonation patterns and each speaker may have specific habits of expressing certain emotions. In an ideal voice conversion system, it is important to capture these global habits and manipulate the entire pitch contour accordingly while converting from one speaker to another. Modelling speaker dependent intonation patterns is also instrumental in capturing the more stable and long-term intonational differences that stem from the speaker's dialect, socioeconomic group or even gender. Therefore, implanting an appropriate pitch contour is crucial to retain the perceived naturalness of converted speech.

1.2 Summary of Algorithms Presented

This work explores four algorithms of pitch conversion in addition to the standard linear mean-variance transformation. This standard approach is used as a baseline for evaluating the other methods. Each method takes a pitch contour that belongs to the source speaker and uses training data to generate a corresponding target contour. For evalua-

tion purposes, a novel pitch transplantation framework is developed to fit the converted contour onto the identical utterance produced by the target speaker. Such transplantation of pitch contour assumes that everything other than the pitch contour is a perfect conversion and therefore isolates the performance of each method.

The first two algorithms extend the baseline transformation by estimating more sophisticated conversion functions from the training data. However, similar to the baseline approach, these functions map the frequency values on a frame by frame basis. The first algorithm develops a unique Nth order function for a given source-target speaker pair using all the phonetically aligned pitch values of training utterances. The second algorithm models the source speaker pitch as a GMM and trains a conversion function based on GMM parameters. In contrast to the first two methods, the third and the fourth investigate ways of mapping the entire pitch contour of the speaker instead of instantaneous pitch values. The third algorithm explores the construction of a codebook of source and target utterance contours and transforms the contours by picking the most likely target contour directly from the codebook. The fourth method also uses the codebook approach, but instead of entire utterance contours, the codebooks consist of contours of voiced segments, which are subsections of training utterances. In this case, the conversion happens on each voiced segment of the input as opposed to the entire contour.

1.3 Outline

This document can be outlined as follows: Chapter 2 provides a background in pitch modelling and conversion. Chapter 3 introduces the speech corpus. Chapter 4 presents the experimental setup, which includes the model used for speech signal analysis and synthesis as well as the pitch transplantation framework. Chapter 5 explains each investigated algorithm in more depth. Chapter 6 presents and discusses the evaluation methods as well as the results of the objective and subjective tests. Chapter 7 contains concluding remarks for this thesis.

2 Background

2.1 What is a pitch contour?

A pitch contour refers to the rise and fall of the fundamental frequency, f_0, over time. Since f_0 is only defined for voiced sections of a speech signal, the pitch contour has positive pitch values for voiced intonation groups separated by gaps for unvoiced regions of the speech signal. The unvoiced regions corresponds to the condition where the vocal chords are not vibrating, such as the production of certain plosives or fricatives. As noted in the introduction, pitch contours reflect expressions of emotion as well as linguistic feautures such as the sentence type. For instance, most declaratives have an overall declining pitch contour. On the other hand, yes-no questions usually have an extreme final upturn in the last intonational phrase which sometimes causes the overall contour to incline. While such patterns are more or less speaker-independent, there may be intonational trends specific to a speaker or a group of speakers. Figure 1 illustrates the pitch contour for the question "Did you buy corduroy overalls?" spoken by a British male speaker and an American male speaker. The regression lines through both utterances clearly indicate that the first speaker has a continuously rising contour, whereas the second speaker has as many local rises as falls resulting in a flatter declination line. Even though both regression lines have a positive slope, which is typical for a yes-no question, the way the speakers vary their contours are quite distinct. If such distinctions are repetitive between a given pair of speakers, it would be desirable for a good pitch conversion system to learn and replicate these types of speaker specific habits.

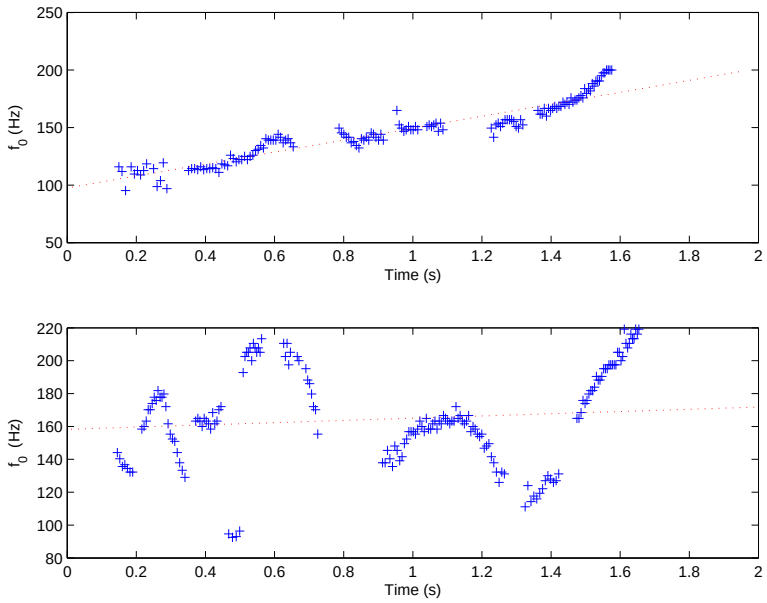

Figure 1: Two contours for "Did you buy corduroy overalls?" generated by two distinct speakers

2.2 Previous Work on Pitch Modelling and Transformation

2.2.1 Mean/Variance Linear Transformation

Substantial work has been undertaken by [1], [2] and [3] on the area of spectral conversion, which involves mapping between spectral envelopes of two speakers at the segmental level. To handle the pitch transformation, all three works employ the mean-variance linear conversion method, with the assumption that "average pitch frequency already carries a great deal of the speaker specific information"[1]. This method involves converting the source f_0 values such that the converted contour matches the average pitch value and the pitch range of the target speaker, while maintaining the intonation pattern of the source. The underlying assumption is that each speaker's f_0 values belong to a Gaussian distribution with a specific mean and variance. A linear transformation can then be defined as follows:

$$t = h(s) = as + b \tag{1}$$

where t is the instantaneous target pitch value and s is the instantaneous source pitch value. The goal is to come up with a and b in terms of the mean and variance values of the two Gaussian distributions.

If the target pitch values have a pdf of p_t with parameters (μ_t, σ_t) and the source pitch values have a pdf p_s with parameters (μ_s, σ_s), the pdf of the linear transformation can be written as follows[17]:

$$p_t(t) = \frac{\partial h^{-1}(t)}{\partial t} p_s\left(h^{-1}(t)\right) \tag{2}$$

Taking the inverse of the linear transformation in Equation 1,

$$h^{-1}(t) = \frac{t-b}{a} \tag{3}$$

$$\frac{\partial h^{-1}(t)}{\partial t} = \frac{1}{a} \tag{4}$$

Therefore Equation 2 can be rewritten as,

$$p_t(t) = \frac{1}{a} p_s\left(\frac{t-b}{a}\right) \tag{5}$$

Replacing both sides by the expression for a Gaussian distribution, we get

$$\frac{1}{\sqrt{2\pi\sigma_t^2}} \exp\left(-\frac{(t-\mu_t)^2}{2\sigma_t^2}\right) = \frac{1}{a}\frac{1}{\sqrt{2\pi\sigma_s^2}} \exp\left(-\frac{(\frac{t-b}{a}-\mu_s)^2}{2\sigma_s^2}\right) \tag{6}$$

Taking the logarithm of both sides we end up with

$$\log\left(\frac{a\sigma_s}{\sigma_t}\right) - \frac{(t-\mu_t)^2}{2\sigma_t^2} = -\frac{(\frac{t-b}{a}-\mu_s)^2}{2\sigma_s^2} \tag{7}$$

The second order terms can then be equated to produce an expression for a in terms of the variances of the source and target pitch distribution.

$$-\frac{1}{2\sigma_t^2} = -\frac{1}{2a^2\sigma_s^2} \tag{8}$$

$$a = \frac{\sigma_t}{\sigma_s} \tag{9}$$

Substituting the value of a into Equation 7 we get

$$b = \mu_t - \frac{\sigma_t \mu_s}{\sigma_s} \tag{10}$$

A fixed number of training utterances can be used to estimate the mean and variance values for each speaker. In the conversion stage, Equation 1 can be applied to the f_0 value of each voiced frame in the input utterance to produce a target contour.

While the mean-variance linear conversion is a widely-used method that achieves decent results given its simplicity, [4] found that the resulting target contours frequently fail to possess the fine prosodic structure of the target speaker apart from the proper pitch range. Experimental results on this method are presented in chapter 4. [2] also confirms the importance of modelling detailed prosodic characteristics and emphasizes the difficulty of this problem.

2.2.2 Further Work on Pitch Contour Modelling and Conversion

There have been various studies that have investigated more complex algorithms for modelling and converting pitch contours, some of which will be mentioned in this section.

[4] has suggested a scatterplot pitch model and a sentence codebook method which form the basis for two of the algorithms presented in this paper.

The scatterplot model aims to estimate a higher order mapping function that converts one speaker's pitch values to another without assuming a Gaussian distribution. [4]'s goal was to apply contours acquired this way to the output of a concatenative speech synthesizer. In this project we reimplement and reevaluate it within a voice conversion framework.

The codebook method of [4] consists of constructing a codebook of pitch contours for all source and target utterances seen in the training data. In the transformation stage, the algorithm finds the source codebook entry that is most similar to the input contour and selects the corresponding target contour. While [4] commends the potential advantages of this method, the study does not elaborate on its performance nor its weaknesses. Section 5.3 presents a revised version of this method within the pitch transplantation framework. One important revision carried out was to linearly interpolate among codebook entries as opposed to simply picking one.

[5] suggests a method where speakers are represented by various parameters which model the distribution of their pitch declination lines and the variance of the pitch contours around that line. No results were made available by [5] on the performance of this model.

[6] introduces a powerful tool for modelling pitch contours called the Multi-Space Probability Distribution HMMs, which can model sequences of observation vectors with variable dimensions including zero dimensional observations such as boolean symbols. The relevance of this in terms of modelling pitch stems from the the fact that pitch values do not exist in unvoiced regions of speech, which means that observation vectors ought to consist of a continuous value for voiced sections as well as a discrete symbol that represents the voiced/unvoiced distinction. Therefore while it is difficult to represent pitch patterns with conventional HMMs without making some heuristic assumptions (such as

explicitly modelling unvoiced symbols in continuous HMMs by giving unvoiced frames the value 0), the MSD-HMMs of [6] not only allow for a natural representation of pitch in continuous voiced regions but also handle modelling of a speaker's voiced/unvoiced patterns. [7] presents a method of adapting speaker-independent MSD-HMM pitch models to become speaker-dependent given training data for a particular speaker. This has promising implications for transforming pitch contours of one speaker to another in a voice conversion framework. However, training or adapting the entire model may require more training data than is available in a typical voice conversion application.

3 Speech Corpus

A speech corpus consists of a collection of recordings from a number of speakers as well as all the supporting materials such as pitch mark files, phonetic transcriptions and laryngograph signal files. Constructing a corpus that is appropriate for the task at hand is essential since the corpus is the cornerstone of all experiments. Not only is the data used for training the various transformation methods but also the availability of a representative range of test utterances directly influences the success of the evaluation procedure. The speech corpus for this study contains data from an existing database (OGI-DB) made available by Alexander Kain [8], as well as new data recorded for the particular purposes of this study.

3.1 Speakers and Text Material

Three speakers were used from the OGI database while three speakers were recorded specifically for this project in Cambridge. Fifty sentences were recorded by all six speakers and these were taken from the OGI database. In addition, the three Cambridge speakers recorded eight additional sentences, which were mostly questions. Three of the speakers were female, while the remaining three were male. All speakers except one spoke with a fluent American accent, and one speaker had a mild British accent.

The need for additional speakers and utterances stems from the different requirements of this project and the OGI study. While the OGI database was also created for the specific task of voice conversion and therefore is very favorable in the way it is organized, the focus of the project for which it was designed was centered around spectral conversion and not prosodic transformation, making the corpus suboptimal in various respects: First and most importantly, the goal of the OGI data was to provide a phonetically rich set of sentences with as little intonational variations between speakers as possible so that the actual differences in timbre could be isolated for further processing.[8] In this study, however, the phonetic richness is not instrumental while the availability of a wide range of intonation patterns is critical. To remove the prosodic variations, the speakers of the OGI project were asked to mimic a reference speaker as they were recording the sentences, which makes the set of mimicked data a very constrained training and test sample set for this project. Figure 2 illustrates the contours of a sample mimicked utterance by all speakers in the OGI corpus. Even though the OGI data also contains non-mimicked, natural takes of the same utterances, because the speakers were not specifically asked to be free with their intonation patterns, the non-mimicked recordings improve only slightly in terms of the variety of intonations that could be generated.

Another problem with the OGI data was the absence of questions and exclamations from the text material. Questions and exclamations are usually the types of sentences that contain rather rich speaker specific intonation patterns. Since the goal of [8] was not to capture such patterns but to avoid them as much as possible, the text material was crafted such that most utterances are varying length declaratives.

For the reasons mentioned above, it was essential to record new speakers for this project and come up with additional utterances. A list of utterances can be found in the appendix.

The OGI speakers were recorded at 22kHz while Cambridge speakers were recorded at

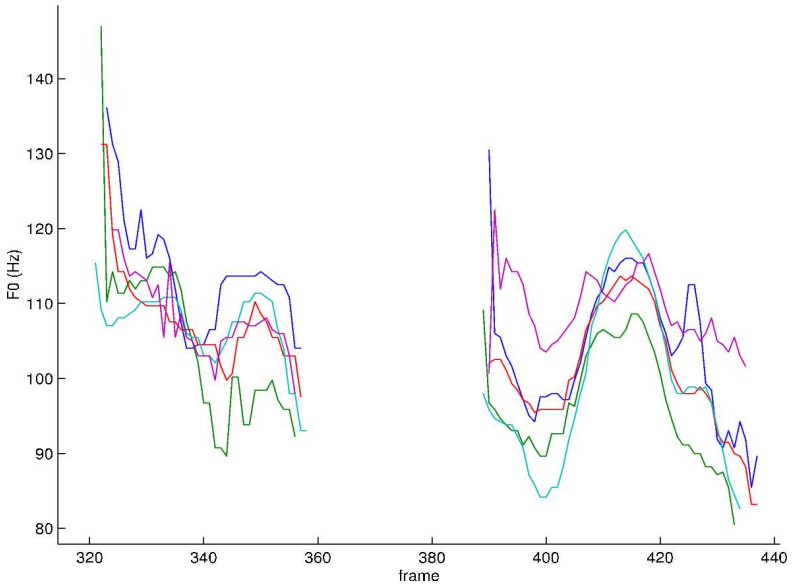

Figure 2: Similarity of five male OGI speaker contours for one utterance.

16kHz. All speakers were connected to a laryngograph processor. The recording quality of the Cambridge data was considerably worse than the OGI data due to the lack of a sound-proof recording booth and a high-quality condenser microphone. There was also a slight amount of ambient noise in the Cambridge recording room. Nevertheless, the laryngograph signals of two male Cambridge speakers turned out to be quite reliable, whereas the female speaker's signal to noise ratio was below optimal.

One advantage of the OGI recordings is the intrinsic time alignment of utterances due to mimicking. Because speaker's were trying to replicate the utterances of the reference speaker in terms of timing and intonation, the length of utterances were more or less the same requiring minimal signal processing for further alignment. Since explicit alignment by dropping/adding too many frames results in unnatural utterances after re-synthesis, the OGI setup for this kind of natural alignment was optimal.

3.2 Pitch Extraction

The laryngograph signal was processed to create pitch mark files for all recordings. Pitch marks indicate the moment of glottal closure and are required by the pitch-synchronous analysis module. The OGI data already included pitch mark files, therefore, only the laryngograph outputs of the new speakers had to be processed during this project. A simple peak-picking algorithm was implemented to detect the pitch marks. The algorithm used on the OGI data may have been more robust and reliable in peak-picking, however the overall results of the simple algorithm used in this project also proved satisfactory. Figure 3 shows the speech signal and the laryngograph signal for the phrase "outdoors". The output of the peak-picking algorithm is illustrated by the red plus signs on the laryngograph signal. The time value for each peak was written to a file for further processing by the pitch-synchronous analysis modules.

While the figure shows that most peaks are identified correctly, there is a peak around 1.8 seconds that fails to be identified because it falls below the noise peak threshold, which

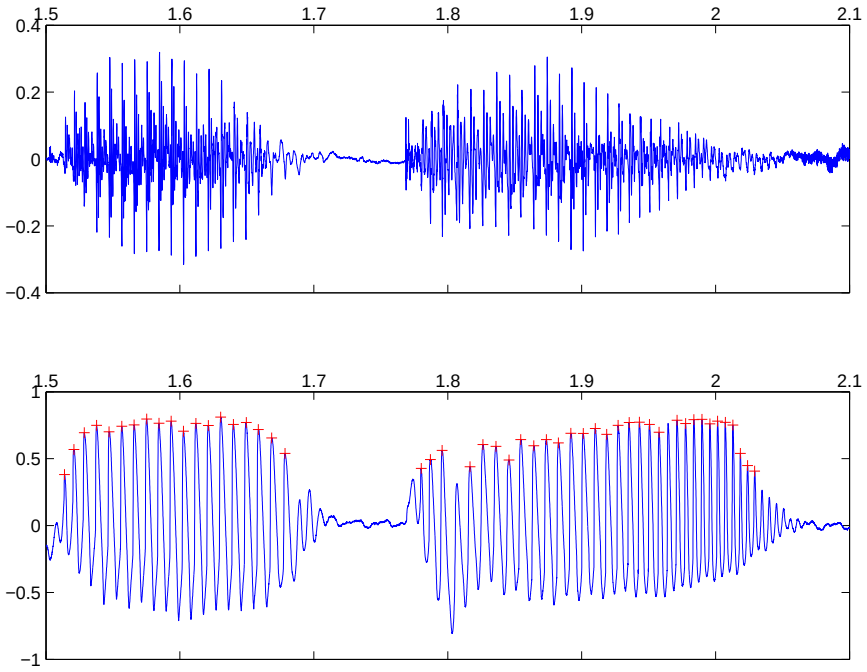

Figure 3: Speech signal and laryngograph signal for the phrase "outdoors"

is set to one standard deviation from the mean. Such errors in peak picking may cause jittering effects when the signal is modified and resynthesized.

3.3 Transcriptions

Transcriptions are necessary for the time alignment of identical utterances produced by different speakers. Time alignment takes place in the process of transplanting the converted contour onto the actual target utterance. It is also needed in the codebook based algorithms for the purpose of aligning source and target codebook entries. Alignment can be carried out using just the voiced/unvoiced decision, however transcriptions provide much more reliable end point constraints for the alignment process.

Transcriptions for the OGI and Cambridge recordings were carried out by running HTK in forced alignment mode on all the data [18]. A large dictionary of English was used for phonetic transcriptions of the words; words that were missing from the dictionary were added. The label files acquired this way were then added to the speech corpus for further processing by the alignment module.

4 Experimental Setup

A flow diagram of the pitch conversion system can be seen in Figure 4. This chapter focuses on the various modules illustrated on this figure, except for the pitch conversion box, which itself takes up all of the next chapter and is the main focus of this project. The steps can be summarized as follows:

- **Analysis:** The incoming source speech signal is analyzed with a pitch-synchronous harmonic model(PSHM), proposed by [9]. The signal is segmented into pitch-synchronous frames and a set of parameters are generated for each frame. These parameters enable convenient yet reliable modification and reconstruction of the speech signal. The parameters of interest to us are the pitch values, f_0^s, and the voiced/unvoiced binary decisions, u/v^s.

- **Conversion:** The pitch contour of the source signal is input into the conversion module. The output of the conversion module is the transformed pitch contour, f_0^c, which is still the same length as the source. Note that if the source parameters were to be resynthesized with the new contour at this point, the resulting speech signal would retain all the original voice characteristics of the source speaker except for the pitch contour which has been converted to the target. This results in a very awkward speech signal, which does not sound similar to either the source or the target and therefore is very difficult to evaluate objectively or subjectively. Hence the need for the pitch transplantation module.

- **Pitch Transplantation:** A pitch transplantation mechanism enables us to take the target utterance for the same test sentence and replace its original pitch values with the converted contour. The clear benefits of this are twofold: first of all the original target pitch contour, which we are replacing, gives us a benchmark against which we can compare the converted pitch values and come up with objective measures of our various algorithms. Secondly, since the resulting speech retains all the original voice characteristics of the target except the pitch contour, we can truly isolate the perceptual performance of the various algorithms under study, given that all else is a perfect conversion. As can be seen in Figure 4, the transplantation module involves two main steps: time alignment of the source and target utterances/contours and pitch scaling of the target pitch values to match the transformed and time-aligned contour, f_0^{c*}.

- **Synthesis:** Once we actually have all the target parameters including the converted pitch contour, all that's left is to resythesize each frame with the synthesis formula provided by the pitch-synchronous harmonic model.

4.1 Analysis and Synthesis

In order to obtain instantaneous pitch values, a pitch-synchronous analysis/synthesis framework is adopted, where each voiced frame consists of one pitch period. This means that frame sizes are not constant but vary over time with the fundamental frequency. Additionally, the analysis and synthesis model of choice needs to be able to allow arbitrary pitch-scaling without altering the more refined spectral characteristics of the signal or

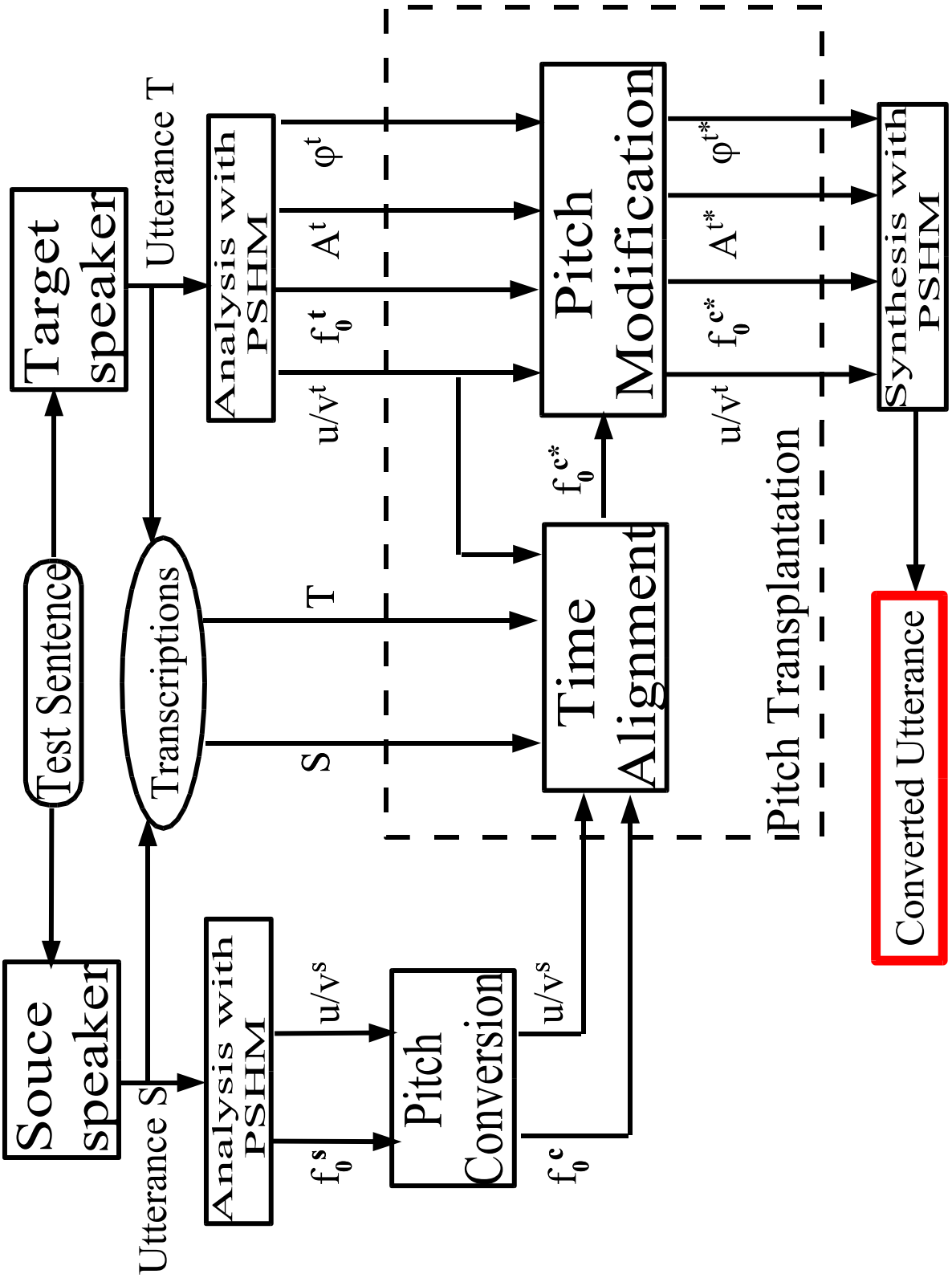

Figure 4: Overview of experimental setup.

causing inter-frame distortions during synthesis. In this project, a new pitch-synchronous harmonic model proposed by [9] has been used. This model is a simplified version of the Analysis-By-Synthesis/Overlap-Add Sinusoidal Model proposed by [10]. It allows for convenient parametric representation and manipulation of the speech signal, particularly for the purposes of pitch modification. The software for the analysis and synthesis of the speech data has been provided by the authors of [9], while some modifications were made for the purposes of this project.

The first step of signal analysis is to chop up the signal into pitch-synchronous frames,the second is to find all the sinusoidal parameters per frame using the harmonic model.

4.1.1 Frame Boundary Decisions

Voiced sections of the signal are divided into frames that are one pitch-period long. Frame boundaries in unvoiced sections are decided using a default frame size equal to the sampling frequency divided by 100 or however many samples are left between the previous unvoiced frame and the next voiced frame. Pitch mark files (see 3.2) drive the decision of whether a frame is voiced or unvoiced. If the distance between two consecutive pitch marks is longer than twice the default unvoiced frame size, that region between the two marks is labeled as unvoiced. In the cases where this is not the case (i.e. the difference between two pitch marks is shorter than twice the default unvoiced frame size), the region is checked for the number of zero crossings. If the number of zero-crossings is unnaturally high for a voiced pitch period, the region is also labeled as unvoiced. Otherwise a single voiced pitch-period is assumed between two pitch marks.

4.1.2 Pitch Synchronous Harmonic Model

Pitch Synchronous Harmonic Model (PSHM) was proposed by [9] in an attempt to advance the quality of speech modification and handle inter-frame phase incoherence problems that arise due to arbitrary pitch scaling[9]. All voiced and unvoiced frames are represented as a sum of harmonically related sinusoids:

$$\tilde{s}_k(n) = \sum_{l=0}^{L_k} A_l^k \cos\left(l w_0^k n + \phi_l^k\right) \quad n = [0..., N_k] \tag{11}$$

where s_k is the kth frame, L_k is the total number of harmonics for the k_{th} frame and A_l^k, ϕ_l^k are the parameters of the lth harmonic of that frame. In contrast to the Analysis by Synthesis parameter estimation method of ABS/OLA [10], on which PSHM was based, the harmonic model estimates spectral parameters by taking the short-time Fourier transform and selecting the first L_k harmonics in the frequency domain to represent the speech signal. The short time Fourier transform of the original signal $s_k(n)$ is given by:

$$S(l w_0^k) = \sum_{n=0}^{N_k} s_k(n) e^{-j l w_0^k n} \tag{12}$$

The representation for synthesized speech (Equation 11) can then be rewritten as the inverse Fourier transform

$$\tilde{s}_k(n) = \frac{1}{L_k + 1} \sum_{l=0}^{L_k} S(lw_0^k) e^{jlw_0^k n} \tag{13}$$

where the amplitudes and phases can be expresses as follows:

$$A_l^k = \tfrac{1}{L_k+1} |S(lw_0^k)|$$

$$\phi_l^k = \angle S(lw_0^k)$$

Because the analysis process is pitch-synchronous, synthesis frames are the same as analysis frames, and no overlap-add method, such as the one proposed by [10], needs to be employed for smooth synthesis. Pitch modification is directly performed on the pitch-synchronous frames.

4.2 Pitch Transplantation System

Pitch transplantation involves generating utterances that inherit all their acoustic parameters from the target, except for the pitch contour which is replaced by the converted pitch. Variations of prosodic transplantation systems have been suggested in various studies including [11][12].

In this project, transplantation takes place in two stages: first, the transformed source contour is time aligned with the target contour based on phonetic transcriptions and then the target pitch contour is scaled to match the transformed and time-aligned source contour.

4.2.1 Motivation for Pitch Transplantation

The main motivation for a transplantation mechanism is to evaluate the different conversion algorithms within the same framework. Since this project is merely concerned with converting pitch contours, it is critical to be able to isolate the effects of pitch conversion in a voice conversion framework. If we were to use an existing voice conversion system into which the various pitch conversion algorithms were plugged, it would be very difficult to assess the performance of the pitch conversion aspect in isolation, given that most voice conversion systems are far from perfect. For that reason, a pitch transplantation system where one can assume perfect conversion apart from the pitch values is an ideal setup for both objective and perceptual evaluations of the algorithms. The goal of transplantation is, therefore, to provide a fair framework to assess both the audible differences and the measured distortions between the algorithms and learn what aspects of pitch contribute most to perceptual naturalness of converted speech.

4.2.2 Phonetic Time Alignment of Utterances

The goal of the phonetic time alignment is to adjust the source utterance such that each phone has the same duration as its counterpart in the target utterance. In this project a transcription based alignment of the two utterances is implemented. Every phone or section of silence is aligned one at a time until the number of samples in each phonetic region for both the source and the target are the same. Alignment based on transcriptions

is essential, since it is, for instance, undesirable to drop a set of frames which correspond to an entire phone and leave all other phones untouched. The more desirable approach would be to drop a few frames from each phone depending on the rate of each speaker's pronunciation of that phone. It is very important to note that two utterances that are aligned do not necessarily have the same number of frames. This is due to the variable frame sizes assumed in PSHM due to pitch-synchronous analysis.

The first plot in Figure 5 shows two utterances of the question "Would a tomboy often play outdoors?" by a female and a male speaker. The second plot shows the result of time-aligning the female speaker's utterance with the male speaker.

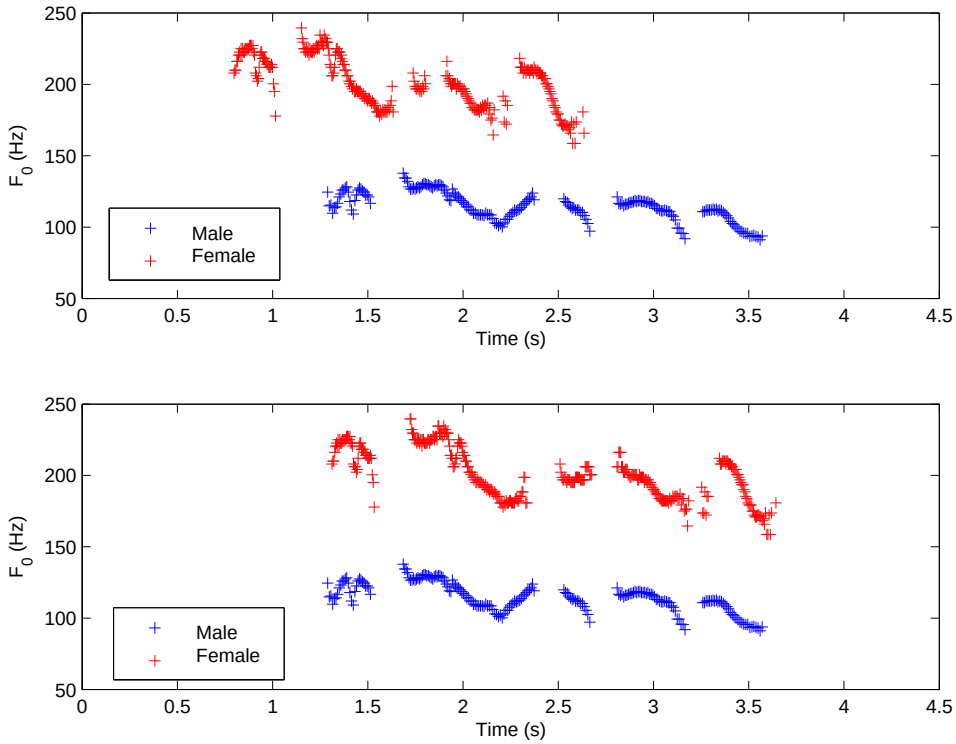

Figure 5: Female and male speaker's utterance of "Would a tomboy often play outdoors?" a) without time-alignment b)female speaker's contour time-aligned with male speaker's contour.

The steps of time alignment implemented in this project can be outlined as follows:

1. Use the phonetic transcriptions to identify the pitch-synchronous frames that fall into each phone. Since HTK label files indicate phone boundaries in 100 nanosecond units, this step involves converting frame boundaries to number of samples and from number samples identifying the set of frames that coincide with each phone.

2. Interpolate the source contour using cubic spline interpolation so that unvoiced frames have meaningful values that represent the voiced regions surrounding them. This is necessary to handle the possibility of unvoiced frames corresponding to voiced frames as a result of alignment. An example of cubic-spline interpolation is shown in Figure 6. Note that interpolation of unvoiced sections in the beginning

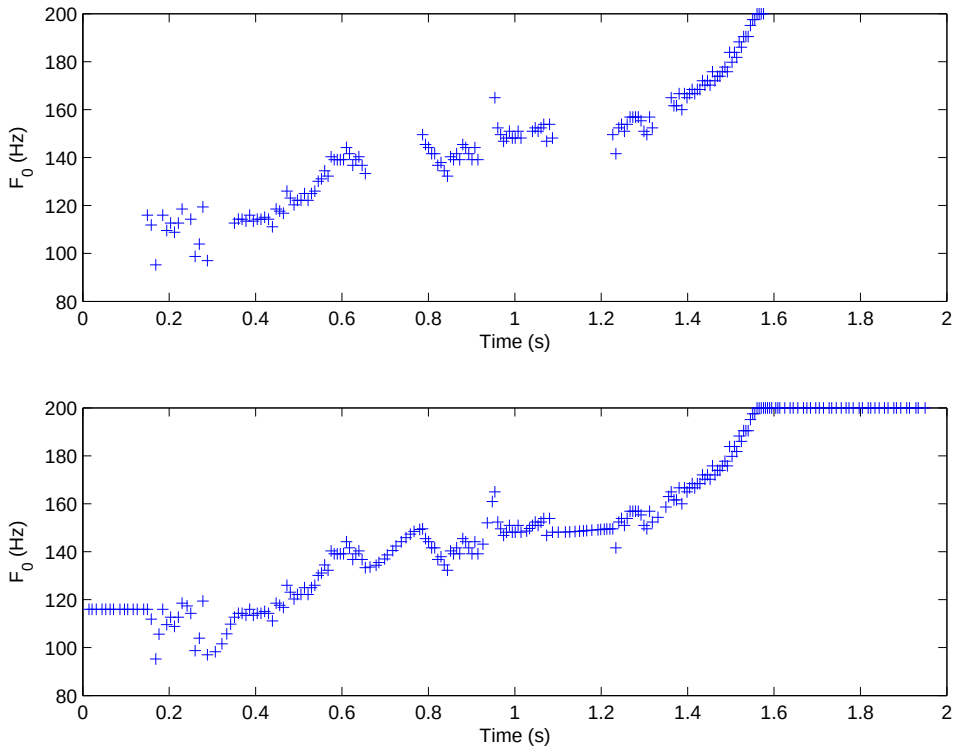

Figure 6: Male speaker's pitch contour for "Did you buy any corduroy overalls?" a)without interpolation. b) with interpolation

and at the end is carried out by repeating the first voiced frame in the beginning and the last voice frame at the end.

3. For each phone or silence in the transcription, the ratio of the number of samples for the target and the source is computed. Based on this ratio, frames of the source utterance are dropped or added uniformly based on a frame selection formula. Note that it is not possible to use a less random approach, such as DTW alignment based on the difference between the true target pitch values and the transformed source pitch values, because we can not assume that we actually have the true target pitch contour in a voice conversion framework.

   Let's say that for the phone **/aw/**, the source contains 6 frames of 50 samples each and the target contains 4 frames of 100 samples each. In this case, even though the target has less frames, due to the lower pitch, it actually has more samples than the source ($100 * 4 = 400 > 50 * 6 = 300$). The goal is to replicate enough frames of the source phone such that it ends up with as close to 400 samples as possible.

   If ts is the ratio of target sample length to source sample length, which is $4/3$ in this case, and N_s is the number of source frames for phone $/aw/$, we can write the number of frames we need to have in the aligned source as

$$N = \text{round}\left(N_s * ts\right), \tag{14}$$

   which gives us $6 * 4/3 = 8$. Now the question is to decide which two frames to

duplicate. If s_x is the source frame index such that $1 <= s_x <= 6$, and s_a is the aligned source frame index such that $1 <= s_a <= 8$, our frame selection formula which selects a source frame s_x for each aligned frame s_a, can be expressed as follows:

$$s_x = \text{round}\left(\frac{s_a - 1}{ts}\right) + 1; \tag{15}$$

Figure 7 visually illustrates which frames of the source are selected using the selection formula of Equation 15. Note that frames are replicated periodically throughout the phone (every three frames in this case) as opposed to just at the beginning, middle or end. Such methods were also investigated but performance of this selection formula was perceived to be the best in terms of preserving phone integrity.

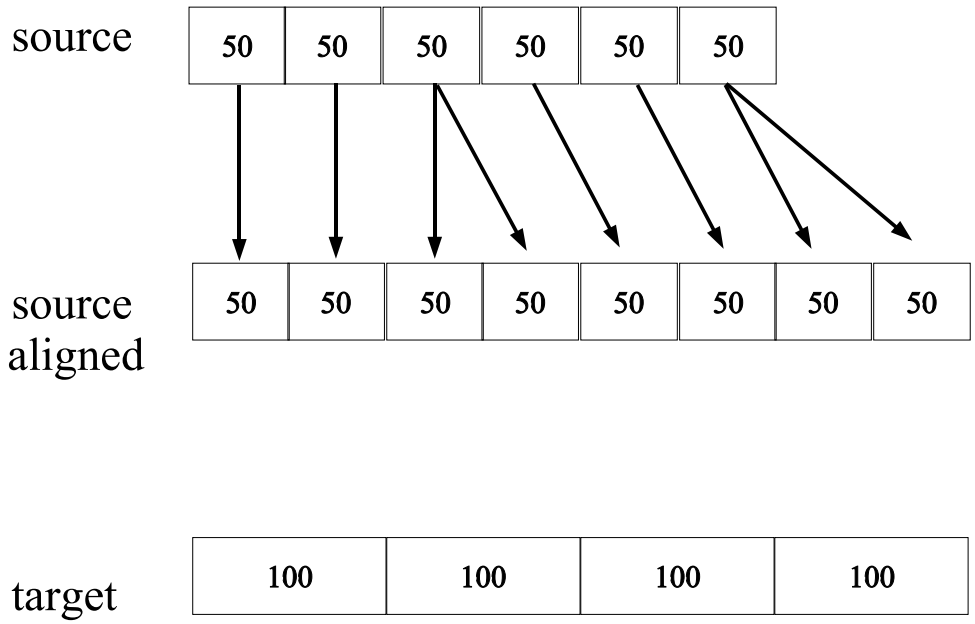

Figure 7: Time alignment of voiced frames for the phone '/aw/'.

The time-aligned source and target utterances will have the same duration but may have a different number of frames. Therefore, the final step of contour alignment is to resample the pitch values in the converted contour to fit the number of frames (pitch values) in the target. This is a trivial process of interpolation.

4.2.3 Pitch Scaling Using Pitch Synchronous Harmonic Model

Once we have the pitch contour that has been converted and time-aligned, all that is left to do is to copy that contour onto the target utterance without changing any of its spectral characteristics. This is achieved through the Pitch Synchronous Harmonic

Model by simply scaling the pitch periods of all voiced frames with the appropriate scaling factor, recomputing all the sinusoidal model parameters per frame given the new pitch values, and undoing the time compression/stretching effects caused by changing pitch periods.[9]

If f_0^{c*} is the converted and time-aligned contour and f_0^t is the pitch contour of the true target utterance, the per frame scaling factor can be written as

$$s(t) = \frac{f_0^{c*}}{f_0^t} \tag{16}$$

for all t where the frame is voiced.

This scaling factor is used to modify the pitch period of every frame and recompute a new set of harmonics based on the new f_0 values. It is important to note that because the frames sizes are modified as well, a time restoration process has been provided which add or drop frames until the original length of the utterance is restored. [9]

Once the scaling and restoration is completed, we have a set of parameters which represent an utterance with all the original spectral and durational characteristics of the target apart from the pitch contour which is the same as f_0^{c*}. PSHM has been particularly advantageous for this purpose since it allows arbitrary pitch modification of each consecutive frame without damaging the spectal envelope of the frame or requiring more complex resynthesis methods such as overlap/add.

5 Pitch Conversion Algorithms

In this chapter we introduce the pitch conversion algorithms implemented and evaluated in this project. The first subsection reviews the baseline mean-variance linear conversion method and introduces some sample conversion results as motivation for the rest of the chapter. Each of the remaining subsections introduces one of the algorithms and explains both the training and conversion stages in detail. A number of sample results that demonstrate the relative performance of each algorithm compared to the baseline, are also presented in each subsection. The same set of utterances were used to train all algorithms for fair comparison. A thorough documentation of objective and subjective evaluation results is provided in Chapter 6.

5.1 Baseline Mean-Variance Linear Conversion

The mean-variance linear conversion method described in section 2.2.1 was implemented. Thirty utterances of each speaker were used to extract the mean pitch and pitch standard deviation over all voiced frames. These values were set aside for each speaker. This concludes the training phase of the baseline method.

In the conversion stage, the pitch contour of the new input source utterance was extracted as detailed in 5.1. Pitch values of each voiced frame is then converted using Equation 17.

$$t = \frac{\sigma_t}{\sigma_s} s + \mu_t - \frac{\sigma_t \mu_s}{\sigma_s} \tag{17}$$

where s is the instantaneous source pitch value, (μ_s,σ_s) represents the source parameters and (μ_t,σ_t) represent the target parameters. For a detailed derivation of this conversion formula, see 2.2.1. Once the converted contour is obtained, it is time-aligned and re-sampled to fit the target utterance (see 5.2 for more details on pitch transplantation). Figure 8a shows the original source contour of a male speaker for the utterance "Challenge each general's intelligence" and the result of conversion to a female speaker using the mean-variance method. Note that the converted contour is more or less a scaled version of the source to the appropriate pitch range of the target female speaker. Figure 8b includes the original target contour of the female speaker as well as the time-aligned, resampled version of the converted utterance. For this particular example, the baseline method works rather well since both the source and the target have similar intonation patterns, which means that simply matching the right pitch range already goes a long way in converting from the source contour to the target.

In another example utterance for the same source and target speakers, however, the baseline conversion doesn't perform as well. Figure 9 shows the contours for the utterance "Trish saw hours and hours of movies on Saturday". Not only is the time-aligned converted contour of 9b mostly below the original contour of the target, but it also fails to capture the finer intonational details of target contour. Perceptually, the converted utterance sounds not too far from the target in terms of appropriate pitch range, however, in terms of the actual intonation, the converted contour creates an unnatural sequence of rises and falls when transplanted on the target utterance.

The two speakers whose contours we have examined in these two examples were taken from the OGI speech corpus. As noted in chapter 3, the OGI speakers were asked to mimic

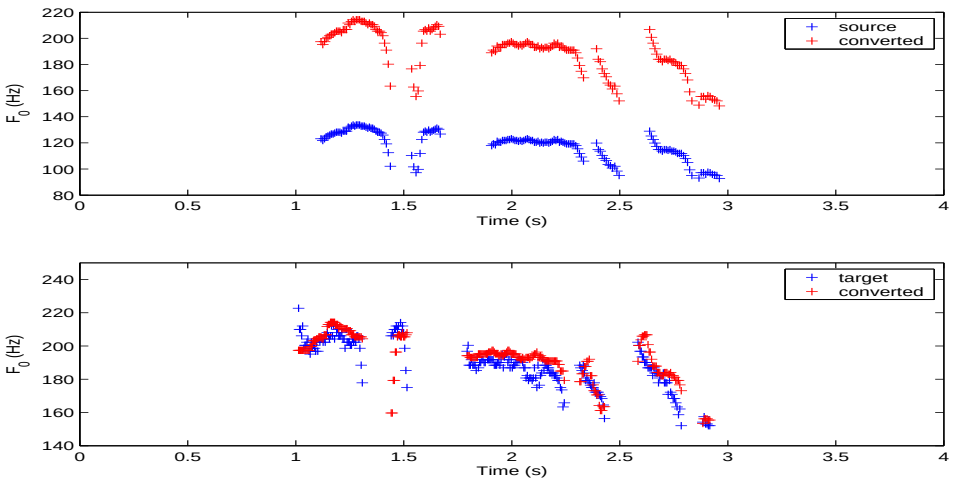

Figure 8: Baseline conversion for "Challenge each general's intelligence" a)source contour in blue, converted contour in red. b)original target contour in blue, converted time-aligned contour in red

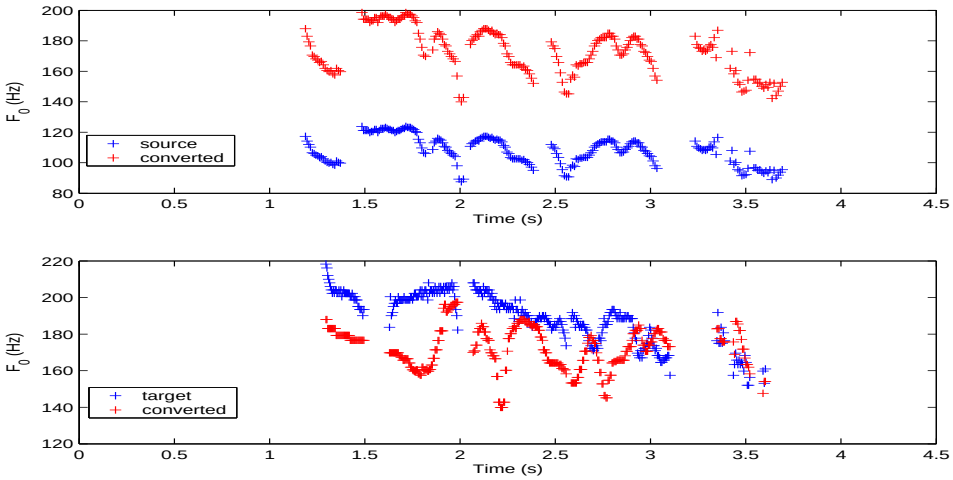

Figure 9: Baseline conversion for "Trish saw hours and hours of movies Saturday" a)source contour in blue, converted contour in red. b)original target contour in blue, converted time-aligned contour in red

a reference speaker during recording in order to minimize intonational differences. Despite this precaution, some of the utterances such as the one mentioned in Figure 9, still display significant variations in intonation patterns. The differences are even more significant in the case of Cambridge speakers who were recorded without mimicking, which is probably a more typical setup for voice conversion applications in general.

Figure 10 illustrates conversion between two Cambridge speakers for the same utterance, "Trish saw hours and hours of movies on Saturday." Once again the source is a male speaker and the target is a female speaker. As we can see from Figure 10b, the baseline conversion fails even more dramatically in capturing the local rises and falls of the original target contour. Perceptually this utterance sounds even more machine-like than its counterpart conversion between the OGI speakers, since even the pitch range is not matched appropriately.

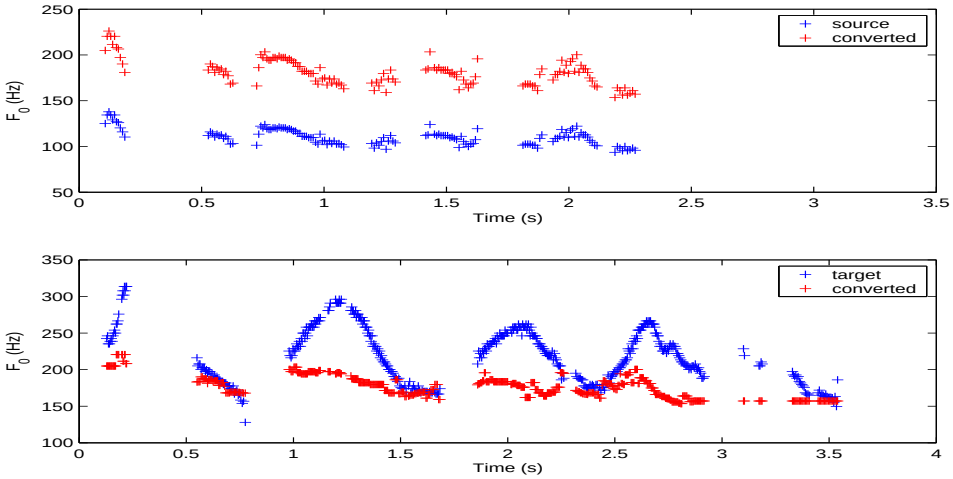

Figure 10: Baseline conversion for "Trish saw hours and hours of movies on Saturday" in the case of non-mimicked recordings a)source contour in blue, converted contour in red. b)original target contour in blue,converted time-aligned contour in red.

These examples are presented as further motivation for exploring alternative algorithms in pitch conversion. They illustrate the cases where the mean-variance method fails to satisfy a basic level of perceptual quality. Objective comparisons of the converted contour with the original target contour support and explain the deficiency in perceptual quality.

5.2 Nth Order Conversion Function

The Mean-variance conversion method relies on the assumption that pitch values belong to a Gaussian distribution. Furthermore the conversion function is constrained to a linear mapping. The Nth order conversion method was investigated in an attempt to relax any assumptions about the distribution of the pitch values as well as to experiment with higher order polynomials as conversion functions. This method is based on the Scatterplot algorithm proposed by [4]. The objective of the scatterplot project, however, was to evaluate the algorithms within a speech synthesis framework by applying converted contours to the output of a concatenative speech synthesis system. In this case, accord-

ing to [4], the artifacts produced by the synthesis system may have prevented the clear detection of pitch changes due to the algorithm. In this project the acquired conversion function is evaluated in isolation using the pitch transplantation system in a voice conversion framework.

5.2.1 Training

The goal of the training phase is to come up with the parameters of the N-th order polynomial, similar to the mean-variance method. The three steps of the training process can be outlined as follows:

1. The first step is to acquire a set of training sentences spoken by both the source and target. This is an additional requirement to the training of the mean-variance linear conversion method, since in that case any set of utterances of the source and target were acceptable. In addition, phonetic transcriptions of each training utterance for both the source and the target are required. We use label files output by HTK in forced alignment mode (see section 3.3). We use the same 30 utterances of the source and target as we used for the mean-variance method.

2. For each phone in each utterance, a mean pitch value is calculated for both the source and the target. The mean value is acquired by averaging the pitch values of all voiced frames that fall within that phone, according to the transcriptions. For a given phone in a specific utterance, the mean value of the source and target are plotted against each other, hence the name 'scatterplot'. The phone level aggregation is necessary, otherwise it is very difficult to gauge what voiced frame in the source corresponds to what voiced frame in the target. Transcriptions allow reliable mapping of source values to target values.

3. Given all the data points, least squares method was used to obtain a Nth order polynomial that best fits the data. In this project, only third and fourth order polynomials were investigated. Because a third order polynomial consistently outperformed a fourth order polynomial for all speaker pairs, the evaluation was conducted on the 3rd order (cubic) conversion function.

Figure 11 shows the mean pitch values per phone of a male speaker and a female speaker plotted against each other. Because these two speakers come from the OGI database, their data points are not too scattered. That is, the mimicked nature of the recordings and the resulting lack of freedom in producing arbitrary intonations manifests itself in terms of a very tight distribution with few outliers. However, when we actually compare the approximation made by the cubic conversion function (in blue) with the linear conversion function of the mean-variance method (in green), we see that the cubic nature of the curve approximates the distribution more closely, by adjusting to local means.

The power of the cubic approximation manifests itself much more clearly when we consider the scatterplot of non-mimicked data. Figure 12 shows the scatterplot for two male Cambridge speakers. As we can see, due to much freer intonational patterns, not only is the distribution more scattered but the overall pitch range is wider for both

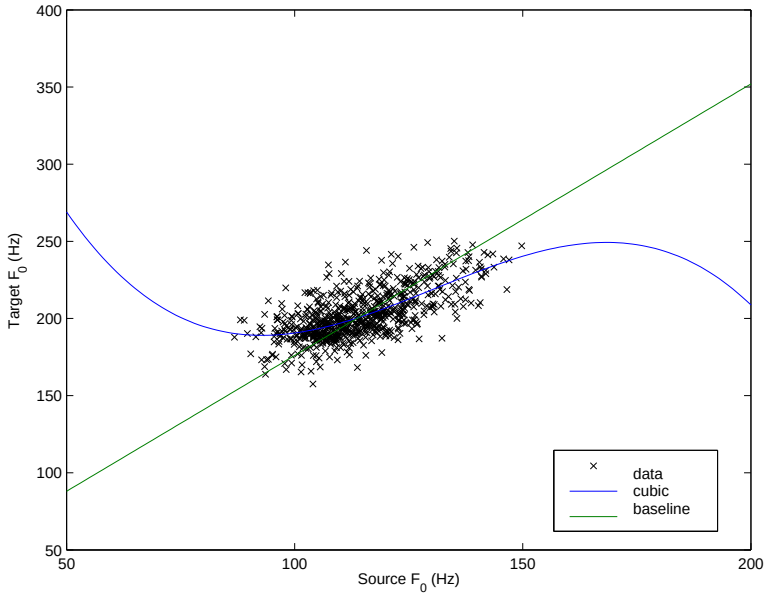

Figure 11: Mean pitch values for a male source speaker and a female target speaker from the OGI data. The cubic approximation is illustrated in blue and the linear mean-variance conversion function in green.

speakers. In this case, the linear approximation of the mean-variance method merely scrapes the lower end of the core pitch range (85Hz-120Hz on the horizontal axis) while the cubic slices the data much more evenly.

5.2.2 Conversion

The Nth order polynomial derived in the training stage is used to transform the pitch values of one speaker to another on a frame-by-frame basis in a similar way to the mean-variance method. Pitch transplantation of the converted contour onto the real target utterance is carried out as before.

Figure 13 shows an example of conversion from a female Cambridge speaker to a male Cambridge speaker for the sentence "The rose corsage smelled sweet". We can see in 13a that the cubic conversion method still maintains the overall intonation pattern of the source speaker. A comparison of the original contour, baseline conversion and cubic conversion in Figure 13b reveals the fact that the cubic conversion moves the pitch contour further towards the correct pitch range than the baseline does.

Reconsidering the example from Figure 10 in section 5.1, where the intonation patterns of the male source and the female target varied dramatically for the utterance "Trish saw hours and hours of movies on Saturdays", Figure 14b plots the result of the cubic conversion for this particular utterance. It is clear from the results that the contour moves closer to the original but is still far from the intonation pattern produced by the target.

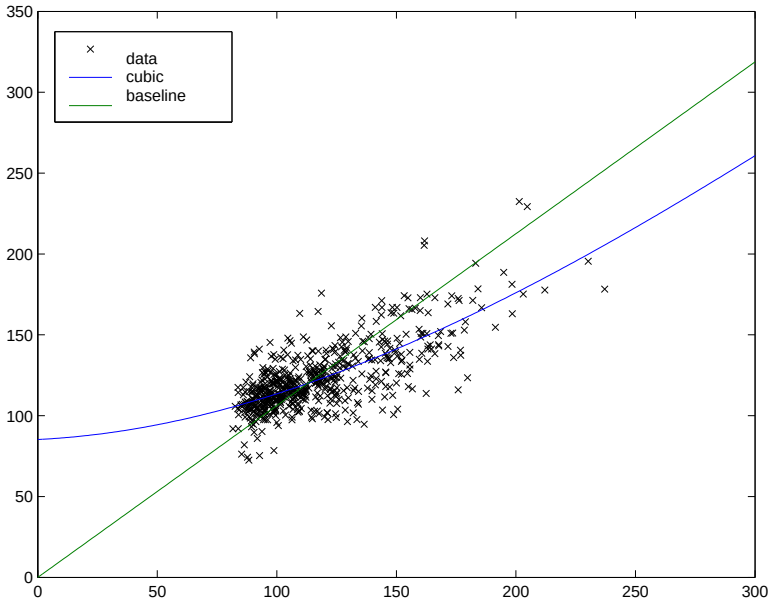

Figure 12: Mean pitch values for two male Cambridge speakers.

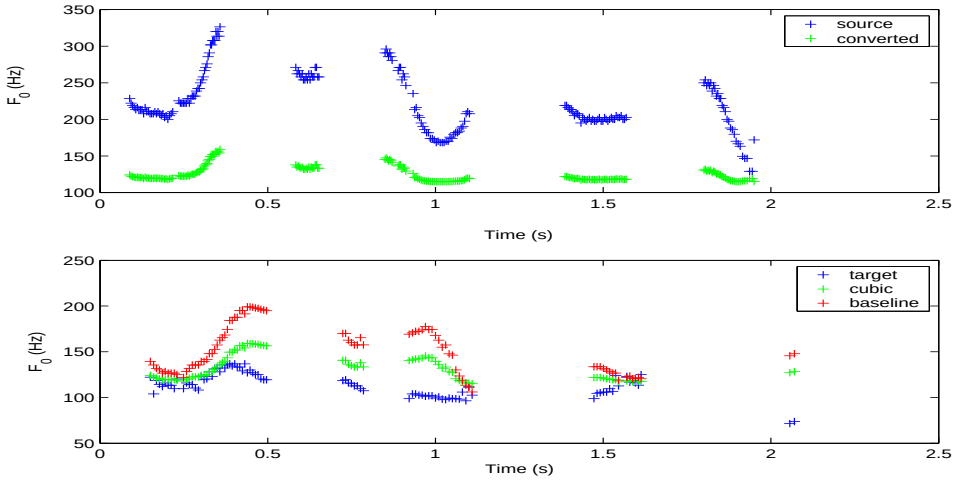

Figure 13: Comparison with baseline for "The rose corsage smelled sweet" for a female to male conversion. a)Input contour in blue, cubic conversion in green. b) Target contour in blue, baseline conversion in red, cubic conversion in green.

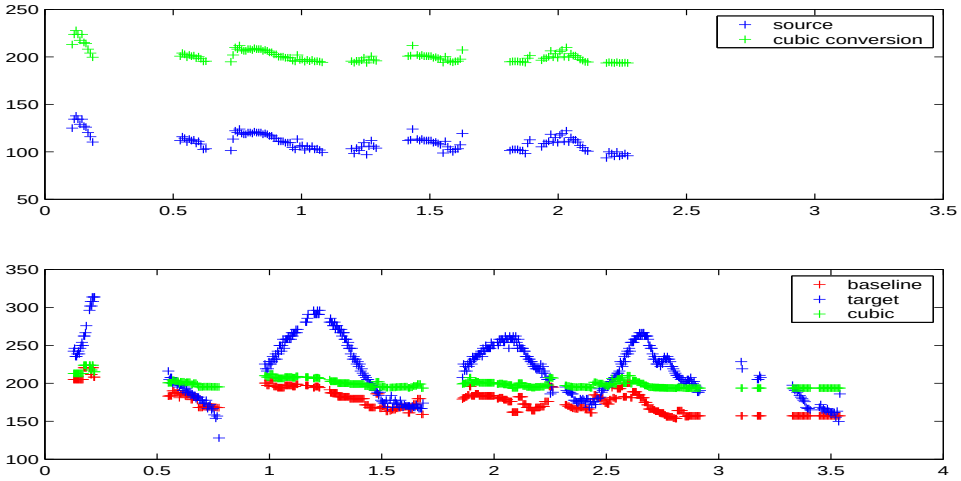

Figure 14: Comparison with baseline for "Trish saw hours and hour of movies on Saturday"

5.3 GMM Conversion Function

The scatterplots of the previous subsection inspired the Gaussian Mixture Model conversion method. Referring back to Figure 12, we were able to identify a core pitch range of more frequent mappings from one speaker to another in the lower ranges of both speakers and a more scattered higher range of outliers. This kind of segregation in the data points to regions in the mapping space where different mapping functions may apply, suggesting a GMM approach for experimentation. In this algorithm, pitch values of the source speaker are represented with a GMM. A conversion function based on the GMM parameters of the source speaker are then trained in similar fashion to the Nth order conversion algorithm.

5.3.1 Brief Introduction to Gaussian Mixture Models

A GMM is a very common parametric model used in various pattern recognition techniques.[1] The GMM is defined in terms of m components each with a component prior α_i, and a Gaussian distribution with parameters μ_i and Σ_i where i represents the ith component. The continuous probability distribution takes the form

$$p(x) = \sum_{i=1}^{m} \alpha_i N(x; \mu_i, \Sigma_i), \tag{18}$$

where $N(x; \mu_i, \Sigma_i)$ represents a p-dimensional Gaussian distribution. In this project, since the goal is to use GMMs to represent pitch values, there is only one dimension. The conditional probability that a given observation, x, belongs to the component C_i of the GMM is given by:

$$P(C_i|x_t) = \frac{\alpha_i N(x; \mu_i, \Sigma_i)}{\sum_{j=1}^{m} \alpha_j N(x; \mu_j, \Sigma_j)} \tag{19}$$

The parameters of the GMM are estimated from the training data pitch values using Expectation Maximization(EM). The EM algorithm iteratively increases the likelihood

of model parameters by maximizing the conditional class probabilities of Equation 19. In this project, a third-party GMM parameter estimation tool was used.[13]

5.3.2 Training

Conversion functions based on GMM parameters have been trained in the past to map spectral vectors of one speaker to another[1]. The methodology used for this purpose in [1] will be applied to the goal of learning a conversion function to map pitch values. The steps of the training stage are summarized as follows:

1. **Fitting the source to a GMM:** The same 30 utterances of the source were used to estimate the GMM parameters as in the previous methods. The parameters of the source GMM were estimated using the third party software, which implements the EM algorithm.[13] The output is a set of Gaussian parameters (μ_{si}, σ_{si}) for each mixture component i of the source pitch distribution. Two and four mixture components were investigated. The use of two components usually outperformed the use of four components, hence two components will be assumed for the rest of this section.

   The two mixture components illustrated in Figure 15 are generated when a male Cambridge speaker's pitch values were fitted to a GMM. The first component has a mean value of about 100Hz and a relatively small variance which indicates a tight distribution while the second component has a higher mean of about 140Hz and much larger variance. In fact, this data belongs to the same source speaker from the scatter-plot of Figure 12. The first component represents the core pitch range of the source speaker while the second peak represents the more marginal pitch values, i.e. the outliers. Because the outliers are more scattered, the variance of the second component is larger.

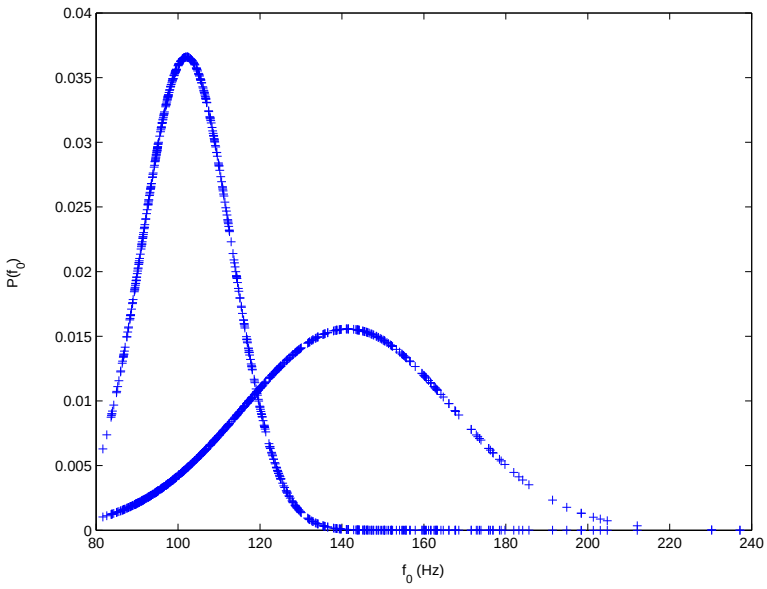

Figure 15: Two Gaussian mixtures produced by male Cambridge speaker.

2. **Form of Conversion Function:** Once a Gaussian Mixture Model is fitted to the source pitch values, the next step is to find a conversion function that transforms each source pitch value to its counterpart in the target. To train the conversion function we need aligned utterances of the source and the target speakers just like we did in the training of the Nth order polynomial. Source and target mean pitch values per phone are computed. If s is the instantaneous pitch value of the source and t is the corresponding instantaneous target pitch value, the parametric form assumed by the conversion function is a single dimensional version of the one used by [1]:

$$t = F(s) = \sum_{i=1}^{m} P(C_i|s) \left[a_i + b_i \frac{(s - \mu_{si})}{\sigma_{si}^2} \right] \tag{20}$$

The conversion function F(s) is completely characterized by the parameters, a_i and b_i for each Gaussian component i. So for a GMM with two mixture components, four parameters need to be estimated and for a GMM with four components, eight parameters need to be estimated. Estimating these parameters constitutes the second step of the training process.

It is worth noting why this particular form for the transformation function was chosen by [1] and adopted in this project. It turns out that this conversion function is an extension of the single-component, single-dimensional limit case, where we can rewrite Equation 20 as

$$F(s) = a + b \frac{(s - \mu_s)}{\sigma_s^2}, \tag{21}$$

where there is a single component with (μ_s, σ_s), and $P(C_i|s) = 1$

If we set a to the mean target pitch value:

$$a = E[t] = \mu_t \tag{22}$$

b to the covariance of source and target and assume that the source and target data are correlated.

$$b = E[(t - \mu_t)(s - \mu_s)] = \sigma_s \sigma_t \tag{23}$$

We can rewrite Equation 21 as

$$F(s) = \mu_t + \sigma_t \frac{(s - \mu_s)}{\sigma_s} \tag{24}$$

Rearranging terms, this reduces to the mean-variance conversion function of Equation 17.

$$F(s) = \frac{\sigma_t}{\sigma_s} s + \mu_t - \frac{\sigma_t \mu_s}{\sigma_s} \tag{25}$$

Therefore the conversion function used in this method is simply an extension of the mean-variance conversion function to support GMMs.

3. **Estimating Function Parameters:** The method of parameter estimation used in [1] is implemented for this project. Since there is only one dimension, i.e. the pitch value, the matrix operations are much less complex and computationally efficient in this case. To turn the problem into a least squares estimation process, we rewrite Equation 20 in matrix form as follows:

$$t = \begin{bmatrix} P & \Delta \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} \tag{26}$$

where a and b are both m by 1 matrices and P and Δ are both n by m matrices, where n is the number of training samples (s, t). P is simply a matrix of all conditional probabilities, where $P_{ij} = P(C_j|s_i)$. Δ, on the other hand depends on the conditional probabilities and the mixture parameters, such that $\Delta_{ij} = P(C_j|s_i)(s_i - \mu_{sj})/\sigma_{sj}$. For a p dimensional version of these matrices, see [1].

Once we can write the conversion function in the matrix form of Equation 26, we can simply solve for the parameters a and b using least squares.

Figure 16 shows the data points for the two Cambridge male speakers as well as the baseline and the GMM conversion functions. The GMM conversion function looks very similar to the cubic approximation of Figure 12. In fact, conversion results show that these two methods yield extremely similar results, since they both try to model the local clusters of data. GMM achieves this by converting the pitch values based on the various components and the cubic fit by using inflection to bend appropriately to fit local clusters of data.

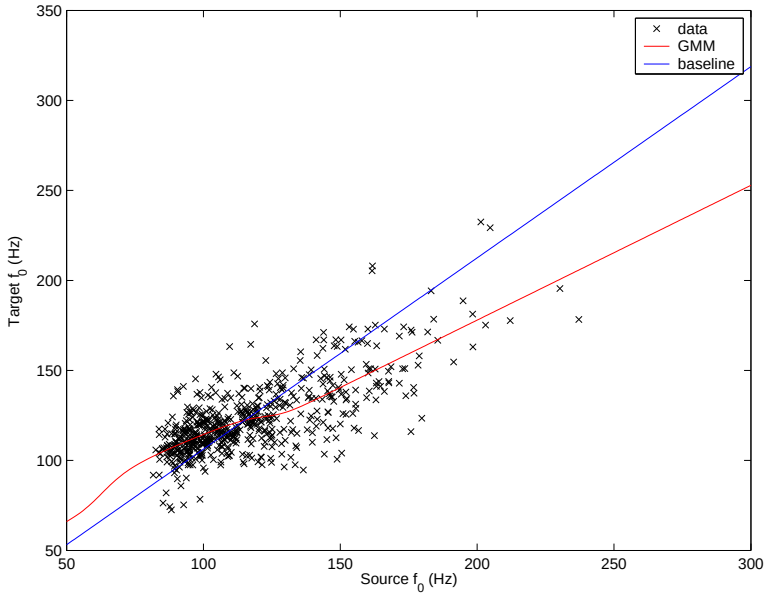

Figure 16: Training Data for Cambridge male to male conversion - baseline conversion function in green, GMM conversion function in red.

5.3.3 Conversion

In the conversion stage, the pitch values of the input source utterance are modified on a frame by frame basis using the conversion function of Equation 20. We show here the same conversion example of the a female and a male speaker from the previous subsection to highlight the similarity between a cubic conversion function and a GMM-based conversion function.

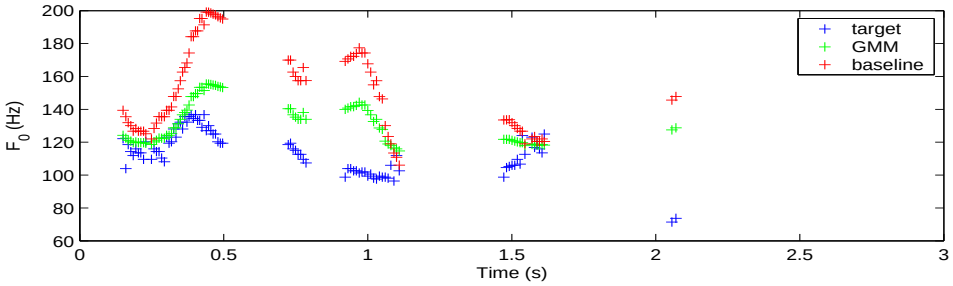

Figure 17: Conversion results for "The rose corsage smelled sweet": converted contour after pitch transplantation in green, true target contour in blue. Note similarity with Figure 13.

5.4 Codebook of Utterance Contours

In contrast to the previous three conversion algorithms, the goal of the codebook algorithm is not just to adjust pitch values on a frame-by frame basis, but to impart an entire pitch contour from a set of target references onto the test utterance. [4] has suggested a codebook based conversion method, where a codebook of source and target pitch contours are set aside for identical utterances. No further processing is required for training. During conversion, the source codebook contour that is most similar to the input source contour is identified using Dynamic Time Warping(DTW). The corresponding target entry is picked as the most likely target contour and transplanted on to the original target utterance. In this project, instead of picking the best contour, we linearly interpolate between all codebook entries. Compared to the single-pick approach, the interpolated contour generally yields a better approximation of the target contour.

It is important to note that the main idea of this method is to impart a pitch contour that was produced by a different utterance of the target onto the input. The assumption is that if the source produces a contour similar to one of the contours he/she produced before (i.e. in training), then the target will also produce a contour similar to that particular training utterance. This is a very strict assumption. Interpolating among codebook contours relaxes this assumption slightly by smoothing over all utterances in the codebook.

5.4.1 Training

There is no preprocessing for training apart from obtaining and setting aside utterance contours for both the source and target speakers. We use the same thirty utterances of the source and target for this purpose. It is important to note that the size of the codebook

as well as the nature of the utterances in the codebook can influence the performance of this algorithm dramatically. For instance, if the codebook consists only of declaratives and we are trying to convert a question, it is very likely that our converted contour will lack the intonational properties of a question, such as a rise at the final intonational phrase. Therefore a codebook that consists of a sufficient number of utterances which cover various sentence types and emotional expressions would be ideal. Unfortunately, it is not always possible impose such specific constraints on the training data for general voice conversion applications. In this project we demonstrate this dependence of the performance on the codebook by assessing it using both mimicked OGI data and non-mimicked Cambridge data.

5.4.2 Conversion

The flow diagram for the conversion stage is illustrated in Figure 18. The basic steps of the conversion can be detailed as follows:

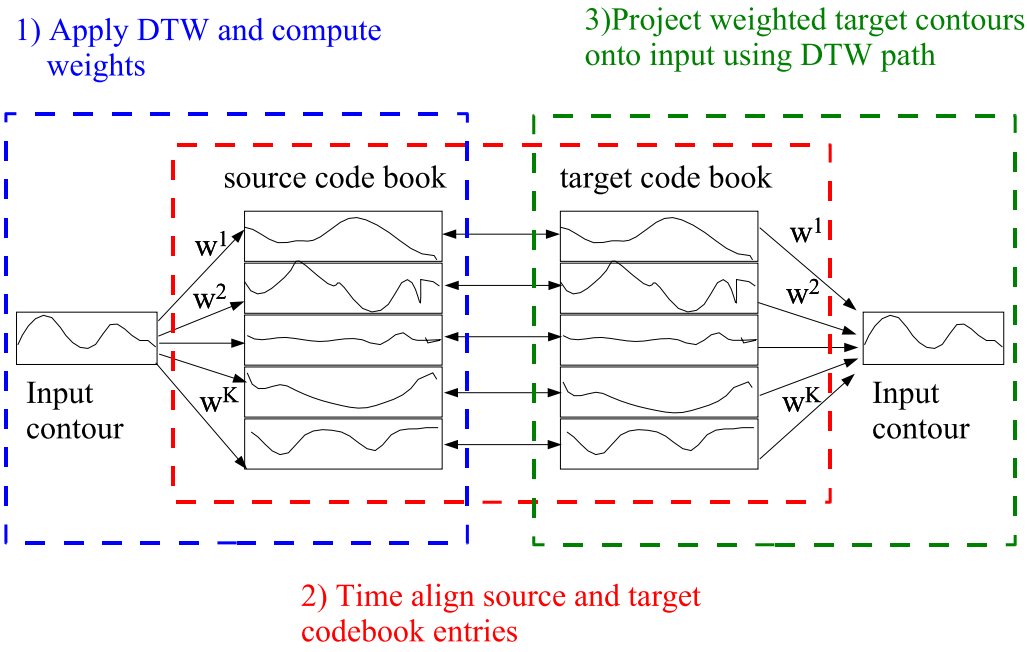

Figure 18: High level flow diagram of the codebook method.

- For an input source pitch contour, DTW is used to find individual time-warping paths with each of the codebook contours. DTW needs a cost function to minimize as it tries to find the best path. In this project, the Euclidean distance between instantaneous pitch values is used as the cost function. The result of running DTW on all codebook entries and the input is a final distance d^i for each codebook contour, i, indicating its similarity to the input.

- Normalized weights for each codebook entry are computed based on the normalized distance between the input contour and each codebook entry. The distances and weights are normalized to sum to unity. d_N^i denotes the normalized distance between the input and the ith codebook entry, and w_N^i denotes the normalized weight of each entry. Such weight computation methods have been used extensively in the literature.[2][12]. If we have K codebook entries:

$$d_N^i = \frac{d^i}{\sum_{j=1}^{K} d^j} \tag{27}$$

$$w_i = \exp(-Ad_N^i) \tag{28}$$

$$w_N^i = \frac{w^i}{\sum_{j=1}^{K} w^j} \tag{29}$$

  The constant A in Equation 27 decides how many codebook entries will influence the resulting contour. A high value of A ensures that only codebook entries that are highly similar to the input are interpolated while the weights of the rest are set to 0. In this project A is set to 500.

- The target codebook entries are time-aligned with their corresponding source codebook entries using phonetic transcriptions.

- The time-aligned target codebook contours are all projected onto the source utterance using the same DTW path computed in the first step. The final converted contour is a weighted sum of all projected target contours:

$$f_0^c = \sum_{j=1}^{K} w^j f_0^{tj} \tag{30}$$

- Once the converted pitch contour, f_0^c is obtained, it is further processed for pitch transplantation as before.

The strength of the code book method becomes apparent when we consider running the conversion on an actual training utterance i.e. the utterance is in the codebook. Figures 19 and 20 illustrate this case with the utterance "Did you buy corduroy overalls?", which is taken from the training data. Figure 19 shows two male Cambridge speakers' utterances of this sentence and can be viewed as a single entry in the codebook for a male to male conversion. As we can see, the intonation patterns of the two speakers are quite distinct.

The first speaker's utterance is input into the conversion system. The DTW process compares it with all the source codebook contours and assigns the highest normalized weight ($w = 0.999$) to the identical version of the utterance in the code book. The target codebook entries are time-aligned and their interpolated sum is projected onto the source utterance. Obviously, in this case, most of the contribution comes from the target's actual version of "Did you buy corduroy overalls?". Figure 20a shows the input contour

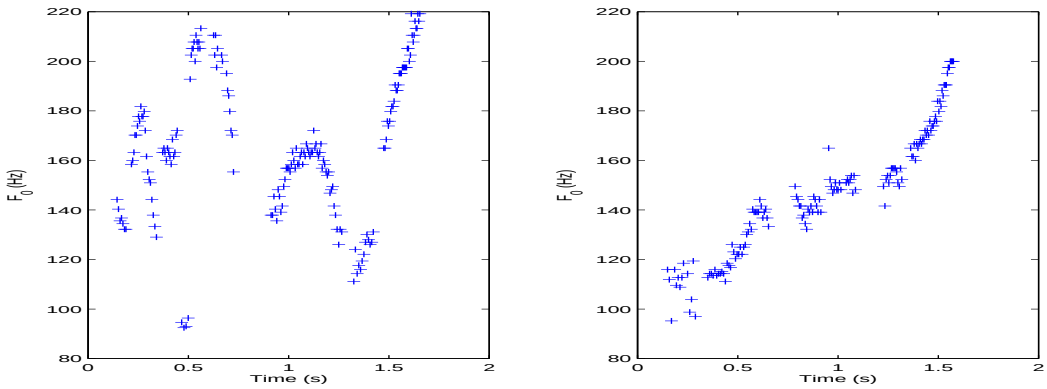

Figure 19: Source and target codebook entries for training utterance "Did you buy any corduroy overalls?"

and the projected target contour. As opposed to the previous methods we have studied, in this case the target contour is not a mere scaling of the source but has the target's unique intonation. Figure 20b compares the transplanted contour with the original and the baseline conversion. Even though the utterance was also in the training data for the baseline system, we can see that the codebook method yields a close to perfect conversion while the baseline is still following the intonation pattern of the source, which is very different from the target, particularly in the first intonational phrase.

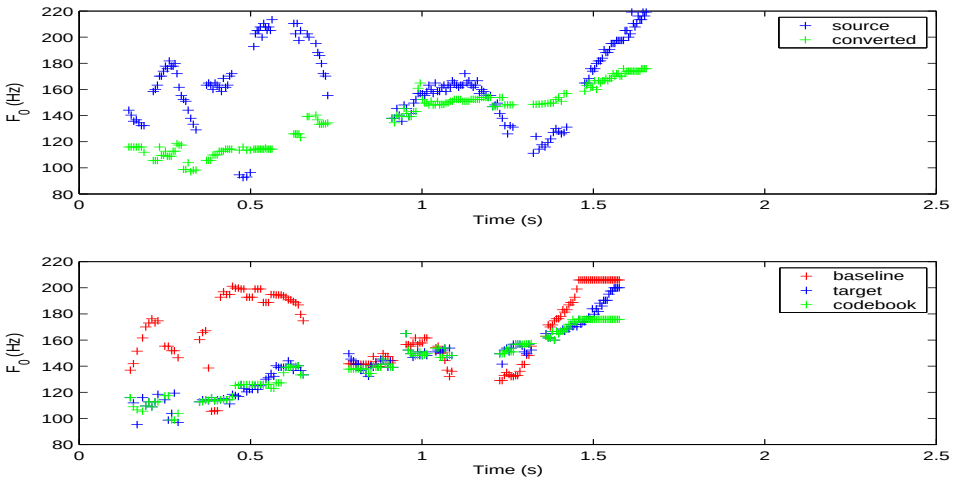

Figure 20: Codebook conversion for training utterance "Did you buy any corduroy overalls?" a) Input contour in blue. Projected interpolated target contour in green. b) Result of pitch transplantation: original target contour in blue, codebook based conversion in green, baseline conversion in red.

We can now go back to the test utterance "Trish saw hours and hours of movies on Saturday" that we have been looking at in the previous sections and see what the results of codebook based conversion are on this 'unseen' utterance. Figure 21 suggests that even though the codebook method does not capture the intonational pattern of the target completely, simply because such a contour does not exist in the codebook, it does move the contour closer to the real one than the baseline, for instance covering at least

the second intonational peak in the target utterance.

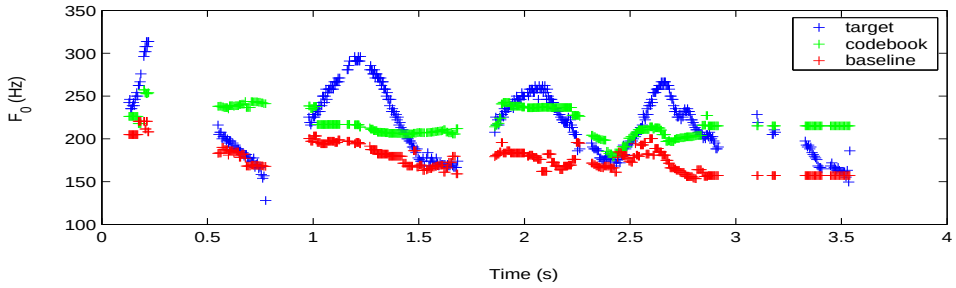

Figure 21: Codebook conversion from a male to a female speaker for "Trish saw hours and hours of movies on Saturday." Original target contour in blue, codebook based conversion in green, baseline conversion in red.

On the other hand, there are cases where the codebook method does not perform better than the baseline. This is mainly due to the fact that an input source utterance may have a contour that is not similar to any of the contours in the training data. Figure 22 illustrates this case. This time 22a shows the weighted interpolation of all source codebook contours (green) and the input contour itself (blue). While the first voiced phrase of the input is matched relatively well by the interpolated codebook contours, the fall and rise of the final voiced phrase is not captured at all by the source codebook. This carries over to the interpolated target contour (green in 23b) which matches the first voiced phrase of the original target (blue) much better than the baseline (red), however underestimates the range and shape of the final contour by a considerable amount.

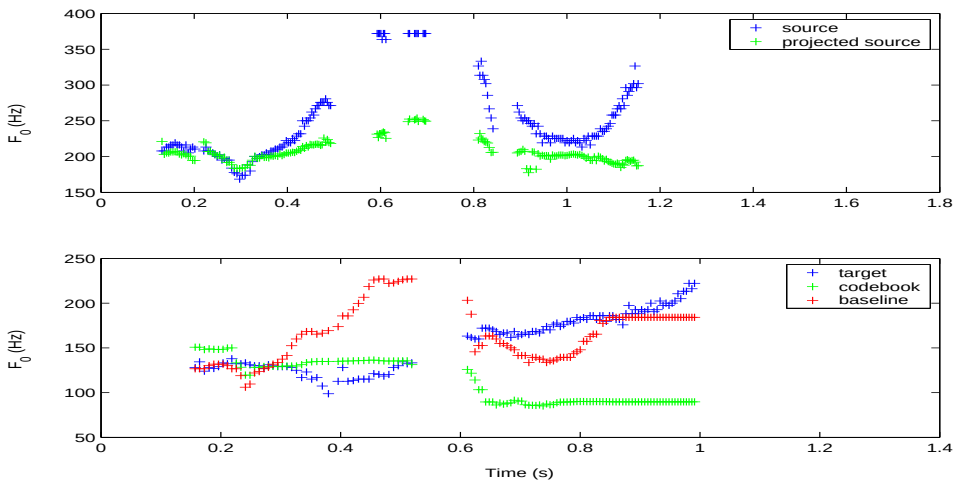

Figure 22: Codebook conversion from a female to male speaker for "Will you go to the may ball?" a) Input contour of female speaker(blue) and interpolated source codebook entries for female speaker (green) b)Target contour (blue), baseline conversion (red), code book conversion (green)

The problem with whole utterances as codebook entries is that we are not making optimal use of our training data. We can think about an utterance level codebook as one end of an extreme, where the other extreme is the frame-by frame conversion which

we have been investigating in the previous three methods. In frame-by-frame conversion, there is no awareness of how to map the overall intonational pattern of the source utterance: instead, the intonation pattern of the source is adjusted to the right range. While we gain some insight into how we can map the entire contour with utterance code books, limiting ourselves to training utterances is not very optimal and can result in awkward intonations in the case of an input which is very different from all source codebook entries. As a natural next step we suggest our next algorithm, which constructs a codebook of voiced segments by chopping up utterances. This way we make better use of our training data by concatenating various pitch segments to form a target contour.

5.5 Codebook of Voiced Segments

A codebook of voiced segments lies between frame-by-frame conversion (which assumes the intonation pattern of the source) and utterance codebooks (which impart the pitch contour of an entirely different target utterance onto the input). The main idea is to create a code book by chopping each utterance into voiced segments and making each source and corresponding target voiced segment an entry in the codebook. The conversion stage is very similar to the previous method, except that we repeat the mapping for each voiced segment of the input, instead of doing it once for the entire pitch contour. This way we impart a pitch contour which is concatenation of segments from various different target codebook utterances, as opposed to a linear interpolation of entire utterance-level contours.

5.5.1 Training

Training consists of constructing the codebooks. This is mainly a process of extracting the voiced segments in the source and finding the corresponding segments in the time-aligned target utterance. To handle the case of voiced frames in the source corresponding to unvoiced segments in the target, all target training utterances are interpolated before extracting segments.

One aspect of the training process that should be pointed out is the treatment of voiced sections which are separated by a very small number of unvoiced frames. Such separation may be due to a short break by the speaker or simply an error in pitch extraction. Therefore we not only add codebook entries for each voiced segment but also an entry for an interpolated concatenation of two adjacent sections. See Figure 23 for an illustration of this process. Using trial and error, we set the threshold for the number of unvoiced frames to 10, which means that unvoiced sections of more than 10 frames are considered to be a proper break and no interpolation between adjacent voiced sections takes place. We found that treating extremely close voiced segments like this improved the performance of this method considerably.

5.5.2 Conversion

In the conversion stage, the following steps are followed:

- The voiced sections of the input voice contour are identified.

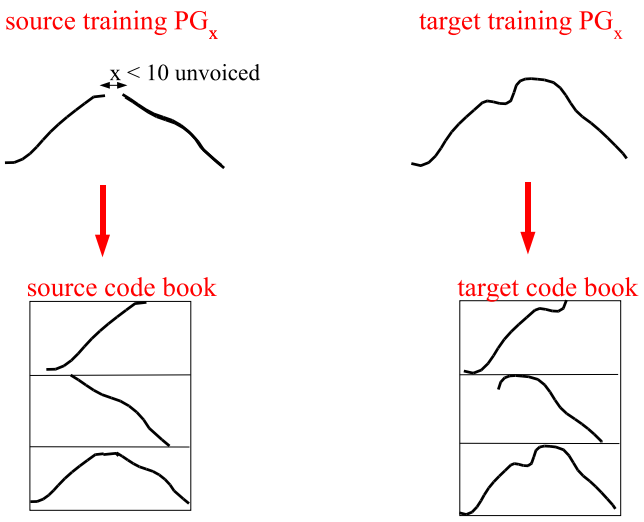

Figure 23: Extraction of codebook voiced segments from source and target training data in the case of close neighboring voiced sections.

- For each voiced segment(VS) in the input that is separated from the neighboring voiced segments by more than 10 frames:

  1. We use DTW to find the most similar source codebook VS.
  2. We take the corresponding time-aligned target codebook VS.
  3. We project the time-aligned target codebook VS onto the source VS using the DTW path.

  It is important to note that there is no linear interpolation in this case, we simply pick the best voiced segment from the codebook.

- For voiced segments that are separated from neighboring voiced segments by less than 10 unvoiced frames, a more complex treatment is performed. Different permutations of voiced segment boundaries are all run through the DTW process and the permutation that yields the codebook entry with the minimal distance is chosen.

Figure 24 illustrates a sample utterance, "The oasis was a mirage", where the voiced segment conversion improved on the utterance codebook method. In this example, both the range and the shape of the contour is mapped much more successfully using the VS codebook, particularly in the first voiced phrase, where the rise is captured much more clearly.

Similarly, taking the problematic example of the previous section, "will you go to the may ball?" and converting it using a PG code book, we get much better results. The input contour is now matched perfectly by the concatenated source codebook entries(Figure 25a). This carries on to the target and now the final voiced segment of the converted contour mimicks the original much more closely than the utterance based codebook conversion (Figure 25b).

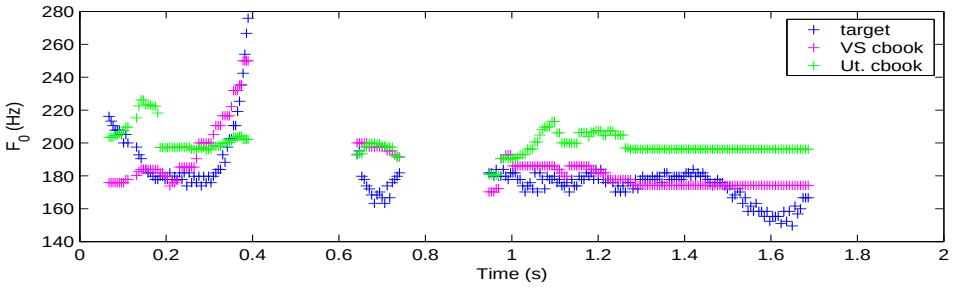

Figure 24: Conversion from a male speaker to a female speaker for "The oasis was a mirage". Original target contour (blue), utterance codebook (green), VS codebook (magenta)

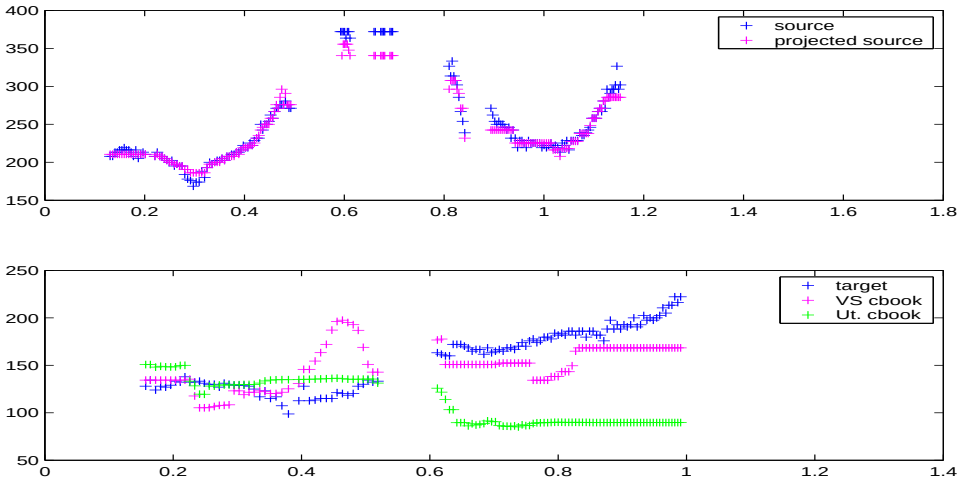

Figure 25: Conversion from a female speaker to a male speaker for "Will you go to the may ball?" a)Input contour (blue), concatenation of best source voiced segments (magenta). b)Original target contour (blue), utterance codebook (green), VS codebook (magenta)

However, this method does not always yield contours that look similar to the original target or ones that always improve on the other methods. Even when the algorithm picks source codebook contours that are almost identical to the input, we found that this doesn't necessarily mean that the target contours will also be similar. Even in Figure 25 we see that while the first voiced segment of the source is matched perfectly by a codebook segment, the corresponding target voiced segment has a peak in its contour that the orignal target does not have. In a sense, there is an aspect of randomness in pitch contour generation that no algorithm can fully capture unless the meaning of the message and its context are somehow incorporated into the algorithm. On the other hand, in some cases, despite the fact that the converted contour looks very different from the original target contour, when resynthesized it may sound very natural and perfectly acceptable as an utterance that could have been produced by the target. This brings us to the question of how reliable our objective measures are. We discuss this in more detail in the next chapter.

6 Evaluation and Results

6.1 Objective Evaluation

The objective measure we chose relies on the pitch transplantation system. Once we have the interpolated, time-aligned converted contour, we measure its distance from the original target contour on a frame by frame basis. The average per-frame distortion, d_u, of an utterance u can then be measured by:

$$d_u = \frac{1}{N_{voiced}} \sum_{i=\{voiced\}} |f_0^c(i) - f_0^t(i)| \tag{31}$$

where N_{voiced} is the number of voiced frames in the original target utterance and i represents each voiced frame. Note that this measure treats the original target contour as the absolute ideal contour, which is not neccessarily true. However it does provide us with a way of assessing in what direction the various algorithms move the pitch contour.

As mentioned in the previous chapter, 30 utterances were used for training for each method. For Cambridge speakers, 28 test utterances were run through all the methods and for OGI speakes 20 test utterances were used. Tables 1 and 2 summarize the objective results. Each table entry represents the average distortion, D, over all test for a given speaker pair. If N_u is the number of test utterances, we can write D as

$$D = \frac{1}{N_u} \sum_{u=1}^{N_u} d_u \tag{32}$$

Algorithm	jay→dan	dan→jay	zey→jay	zey→dan	dan→zey
Baseline	23.86	28.24	31.06	32.67	46.99
Cubic	21.40	23.75	21.75	22.08	39.77
GMM	21.50	23.56	21.37	22.07	38.54
CB(utterance)	23.16	23.98	22.82	22.09	42.32
CB(VS)	23.88	26.19	27.96	25.31	43.92

Table 1: Distortion results from five Cambridge speaker pairs.(in Hz)

Algorithm	nad→jal	jal→leb	leb→nad	zey→leb	zey→jal
Baseline	7.19	10.73	9.81	20.08	13.37
Cubic	6.41	9.52	9.51	13.82	8.86
GMM (2)	6.44	9.58	9.52	13.96	8.94
CB(utterance)	6.53	10.49	11.41	12.29	7.12
CB(VS)	9.55	12.81	13.23	17.68	12.61

Table 2: Distortion results from three OGI Speaker pairs as well as two conversions between a Cambridge source and an OGI target. (in Hz)

The speakers are referred to with their shortened names for convenience: The Cambridge speaker are jay(M), dan(M), zey(F). The OGI speakers are jal(M), leb(F), nad(F). The second table also contains the results of conversions from a Cambridge female speaker (zey) to an OGI female speaker (leb) and an OGI male speaker (jal).

It is immediately apparent from a comparison of the two tables that conversions between OGI speakers in general produce less distortion. This is expected due to the restrained nature of the recordings and reflects the neutral intonation that most OGI speakers assume.

Here is a brief evaluation of the methods based on the objective measures:

- **Cubic and GMM-based conversion functions:** Comparison between the cubic and GMM-based conversion methods show that these two algorithms yield very similar results consistently across all speakers. The reason for this is apparent, when we consider the approximations obtained from these two methods for a male-to-female conversion(dan→zey). Figure 26a shows the cubic approximation and Figure 26b shows the GMM-based approximation of Equation 20 for 2 mixture components .

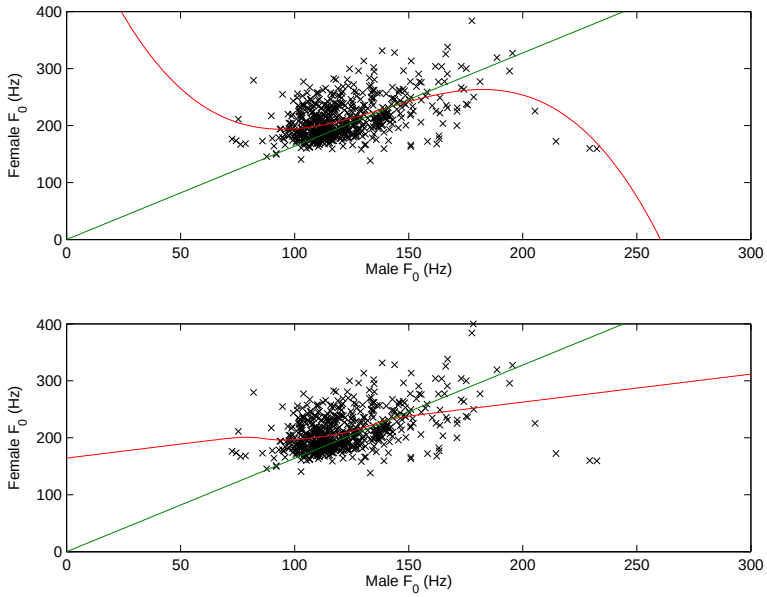

Figure 26: a) Cubic conversion function(red) and baseline conversion function (green) for a male to female conversion. b)GMM conversion function (red) and baseline conversion function (green).

In the core pitch range of the two speakers (80-200Hz for the male speaker and 150-300Hz for the female speaker), these two methods yield very similar mappings from the source to target. This is because they both adjust to local means of the data as opposed to the mean-variance method which uses the global mean to estimate the linear conversion function. The cubic conversion achieves this by inflection and the GMM-based conversion by explicitly modeling two regions of the mapping space. The analogy also manifests itself when we get slightly worse results after increasing

the number of mixture components for the GMM case or the order of the polynomial for the Nth order conversion.

It is worth noting that the outliers are modelled more explicitly by the cubic conversion function. This, however, results in very unrealistic pitch mappings. Figure 26 shows clearly how different the two approximations behave outside the core pitch range of the speakers. While the GMM-based conversion simply becomes a linear mapping function, the cubic approximation bends towards the outliers, resulting, for instance, in mappings of very high source pitch values to very low target pitch values. This is a clear weakness of the cubic conversion function.

Fortunately, it turns out that speakers rarely use the range that is modelled incorrectly. In the end, both the cubic and GMM-based conversions introduce an improvement over the baseline conversion across all speakers. While sometimes these conversions produce approximations that are linear in nature and very similar to the mean-variance conversion, other times approximations diverge from linear more dramatically. The results show that in any conversion task these methods can safely be preferred to the baseline. Table 3 shows the number of times, out of 28 test utterances, the cubic conversion improved the baseline for Cambridge speakers. Table 4 shows the number of improvements, out of 20 test utterances, for the OGI speakers. We see that even in the more restrained OGI pitch distributions, the cubic conversion improves in most cases. In the case of Cambridge to OGI conversions, (zey→leb and zey→jal), the number of improvements are even higher.

Cambridge Speakers	jay→dan	dan→jay	zey→jay	zey→dan	dan→zey
number of improvements over baseline	22	23	26	23	22

Table 3: Improvements introduced by cubic conversion out of 28 test utterances

OGI/Cam speakers	nad→jal	jal→leb	leb→nad	zey→leb	zey→jal
number of improvements over baseline	16	17	10	18	19

Table 4: Improvements introduced by cubic conversion out of 20 test utterances

However, it is important to note the additional requirement of aligned source and target training data for the cubic/GMM methods. In the baseline mean-variance method, we were able to use any data for training, whereas in the cubic and GMM methods, a set of identical utterances produced by both the source and target is required. This may be difficult to obtain in some conversion applications where it is not possible to control the training text of either speaker.

- **Codebook of Utterances:** For conversions between Cambridge speakers (Table 1), the utterance codebook method is also a consistent improvement over the baseline and yields overall results that are nearly as low as the GMM and cubic conversions. The nature of the improvements, however, are quite different. When we look at the number of improvements over the baseline introduced by the codebook

method (Table 5), we see that the numbers are slightly less than their counterparts for the cubic conversion.(Table 3) However, it turns out that more dramatic improvements are introduced by the codebook method on fewer utterances as opposed to smaller improvements introduced by the cubic/GMM conversions on slightly more utterances. Table 6 illustrates this by showing the average improvement (i.e. the total amount of improvement divided by the number of improved utterances) for all Cambridge speaker pairs. It is clear that the values for the codebook method are consistently higher than the cubic.

Cambridge Speakers	jay→dan	dan→jay	zey→jay	zey→dan	dan→zey
Number of improvements over baseline	13	20	21	23	24
Number of improvements over cubic	10	12	12	15	10

Table 5: Improvements introduced by the codebook method. (out of 28)

Algorithm	jay→dan	dan→jay	zey→jay	zey→dan	dan→zey
Cubic	3.82	5.81	10.02	14.03	12.50
GMM	3.70	6.24	9.68	13.95	12.64
Codebook	8.59	8.37	12.51	15.14	13.01

Table 6: Average improvements over the baseline (in Hz).

For OGI speakers, however, the code book method performed either very similarly or worse than the baseline(Table 2). Once again, this is not unexpected, due to the mimicked nature of the OGI data. In many cases the OGI speakers have more or less the same intonation pattern because they are mimicking the same reference speaker. Therefore the baseline conversion works quiet well to begin with since it is based on the assumption that the target will have the same intonation pattern as the source.

On the other hand, when we tried to convert a Cambridge female speaker to an OGI female speaker (zey→leb), the utterance codebook approach immediately showed more improvements over the baseline (Table 2). For instance, for a conversion between two female OGI speakers (leb→nad), the utterance codebook method improved the baseline for only 6 of the 20 test utterances. (Table 7) However, for the case of zey→leb, the codebook method improved the baseline for 16 cases out of 20. It also improved the cubic conversion in 13 out of 20 utterances, while the improvement introduced in the (leb→nad) conversion was merely 3 out of 20.

This shows how much the performance of an algorithm depends on the nature of the training data available. To relate the dependence on training data to a real-world voice conversion application, we can think of converting between two speakers, where for one we have speech data from a real conversation with a wide range of possible intonation patterns and for the other we have a recording of a news broadcast, where the intonation is somewhat neutral and predictable. In this case,

OGI/Cam Speakers	leb→nad	zey→leb
Number of improvements over baseline	6	16
Number of improvements over cubic	3	13

Table 7: Improvements introduced by the codebook method.(out of 20)

the findings of this project would suggest the use of an utterance-based codebook method over a mean-variance linear or cubic/GMM conversion function.

- **Codebook of Voiced Segments:** The voiced segment conversion does yield some improvements over the baseline for Cambridge speakers, however not as often as the other three methods. (Table 1) When compared with the utterance based codebook method, it performs consistently worse. It turns out that when we chop up an utterance into its various voiced segments, we lose the embedded utterance level context information that is very important for intontation. However, interpolating between all voiced segments may yield better results just as it did in the case of utterance level codebooks.

  Table 8 displays the number of improvements over the baseline and the utterance based codebook introduced by matching voiced segments. We see that the improvements over the baseline are fewer than their counterparts in the other methods.

Cambridge Speakers	jay→dan	dan→jay	zey→jay	zey→dan	dan→zey
number of improvements over baseline	13	17	15	21	17
number of improvements over codebook(ut.)	10	8	4	7	14

Table 8: Number of VS codebook improvements out of 28 test utterances

As illustrated in section 5.5.2, in some cases, this method allows us to get very good matches of the target intonation and improve on the utterance based codebook method. Furthermore, in most cases the source pitch contour is matched close to perfection with a concatenation of source code book voiced segments, even in cases where the corresponding concatenated target segments is very far from the real target contour. Therefore it is appropriate to question the effectiveness of the general concept, given that the mechanics seem to performing well.

Finally, it is worth raising the following question about our objective measure: Does the fact that the original target contour is considerably different from the converted contour necessarily mean that the conversion is bad? Clearly, there are various ways a single speaker can utter the same sentence. In fact even in consecutive attempts, it is quite hard for any speaker to utter the same sentence with the same exact intonation pattern. In general, intonation is influenced by conditions such as the speaker's mood, the context and meaning of the utterance and the speaker's physical well-being. Our objective measure, on the other hand, takes one such possible version of an utterance and makes

it the absolute benchmark against which we measure the success of the converted pitch contour. This may not be the most reliable measure, since a very different contour may actually sound natural. Therefore, it is very important to conduct perceptual tests to accompany the objective ones and see to what extent the objective tests reflect the actual success of the conversion and under what conditions the perceptual results contradict the objective measures of distortion.

6.2 Perceptual Evaluation

The quality of the various algorithms were also assessed during formal listening tests. Two listening tests were designed and nineteen listeners participated in both tests. The listening tests were put online so that listeners were able to do them in their own time.

6.2.1 Similarity Test

To assess the similarity of the converted contours to the original target contour, the unmodified target utterance was presented to the listeners along with four options. Each option was a version of the same utterance after pitch transplantation by one of the algorithms. The four options were randomized for each sample presented. The options contained contours modified by the baseline conversion, cubic conversion, utterance code books and the voiced segment codebooks. Because of the similarity between GMM-based and cubic converison, GMM-based conversion was not presented to the listeners. The listeners were asked to mark the option that sounded most similar to the original. Eight sets of utterances were presented. Figure 27 is a screenshot of the web interface for the similarity test.

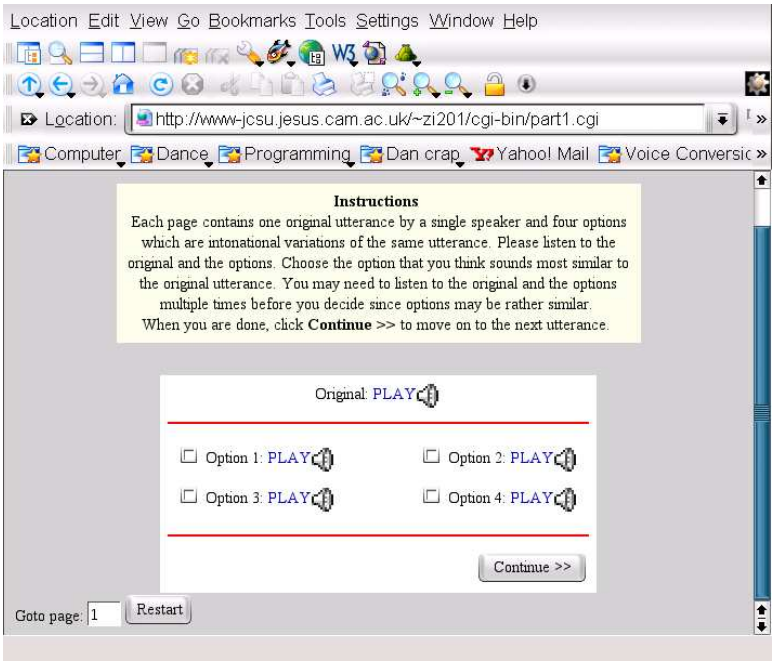

Figure 27: Screenshot of similarity test.

Out of the 8 utterances used in the similarity test, two of them were training utterances

User	1	2	3	4	5	6	7	8
Amir Katz	4	3	4	4	3	3	4	3
Anna Ritchie	4	3	4	2	3	3	3	3
Ben Medlock	1	3	3	3	2	3	3	4
Bjorn Stenger	2	3	2	1	3	3	4	3
Fabre Lambeau	4	2	1	2	3	3	4	3
Henry Robinson	4	3	4	1	2	3	4	3
Nergis Erturk	3	3	3	3	3	3	4	2
Surapa T.	2	3	1	4	3	3	4	3
Thorstein Bostad	4	3	3	2	3	3	3	3
Zeynep Fetvaci	2	3	4	1	3	1	3	4
aslavin	3	3	3	4	3	3	4	4
da209	4	3	3	3	3	3	3	4
erikalopez	4	1	4	1	2	3	4	3
james allen	1	3	4	2	2	3	4	3
jay silver	4	2	3	2	3	3	4	4
jmahowald	4	3	1	1	4	3	3	3
leah hoffmann	4	3	2	4	3	3	3	3
vidura	1	3	2	1	3	3	4	3
zeynepwindows	3	3	3	2	3	3	4	3

Table 9: Results of the similarity test

and the rest of them were taken from the test set. Care was taken to select utterances that have different sentence types(question vs. declarative) and a variety of conversions (male to male, male to female, female to male and female to female).

Table 9 displays the results of the similarity test. The names of the participants and their choices for each test question is shown. The different algorithms were enumerated from 1 to 4, where 1 is the baseline conversion, 2 is the cubic, 3 is the utterance codebook and 4 is the voiced segment(VS) code book. Samples 6 and 7 were training utterances. 3, 4, 5 and 8 were questions while 1,2,6,7 were declaratives.

In the similarity test, 152 choices were made. The breakdown of the number of choices per method is as follows:

	Baseline	Cubic	CB (utterance)	CB (VS)
Number of times method was picked as first choice	14	19	81	38

The utterance based codebook method was the most popular choice in this test. Particularly on the training utterances 6 and 7 , none of the listeners but one picked methods 1 or 2, stressing the success of the codebook based methods on training data. For six of the utterances out of eight, the algorithm that was picked by most listeners was also the one with the least objective distortion. This gave us more confidence in our objective measurements. There was one case, (utterance 1) where the voiced segment

based codebook method was favored more than the utterance codebook despite the fact that the latter had a slightly lower distortion (11.96Hz) than the former(16.80Hz). The reverse happened in utterance 8, where this time the utterane code book method was favored slightly even though its distortion (43.29Hz) measure was slightly higher than the VS codebook method (36.92Hz). This illustrates that in the cases where the objective results are realtively close, perceptual differentiation is harder.

6.2.2 Preference Test

A preference test was also conducted, which simply presented four options, each representing a conversion, without presenting the original target utterance. The goal here was to get a sense of which conversion sounds most natural without biasing the listeners with the original target contour. Since there is no single correct intonation pattern, an utterance that sounds very different from the target may actually sound more natural than an utterance similar to the target. In a sense, this test tries to capture the cases where the objective measures may be contradicted by perceptual evaluations. Eight sets of utterances were presented to listeners in this test. Figure 28 shows a screenshot of the preference test.

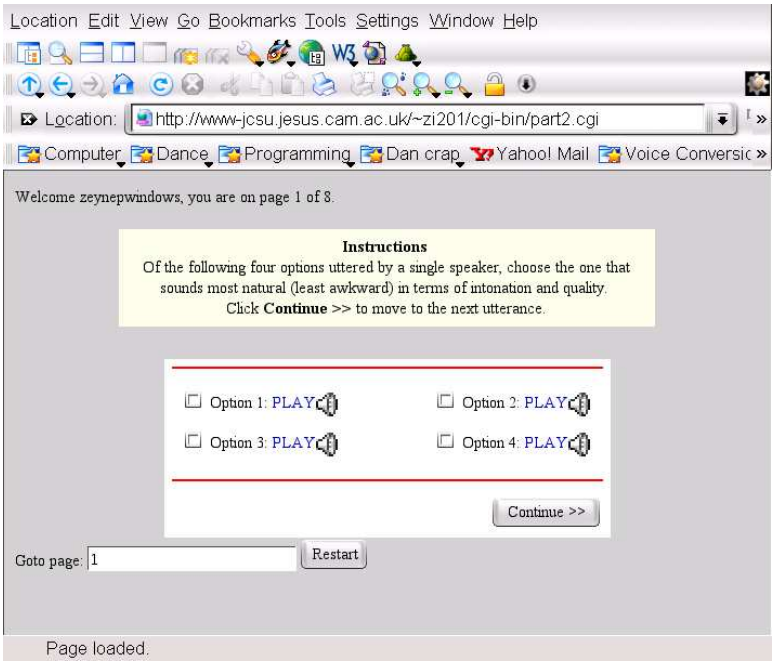

Figure 28: Screenshot of preference test.

The preference test was somewhat more challenging for the listeners since there was no original to compare with. They had to listen to four different versions of the same utterance and decide on the most natural one. In certain cases the utterances had very similar contours in which case the choices were more random than in the first test (Table 10). In this test, care was taken to pick a set of utterances whose objective values are not biased towards one method over the other.

User	1	2	3	4	5	6	7	8
Amir Katz	3	2	3	4	4	1	4	3
Anna Ritchie	2	2	2	2	1	3	4	2
Ben Medlock	3	3	1	1	3	2	2	3
Bjorn Stenger	3	3	1	2	3	4	4	1
Fabre Lambeau	1	2	3	3	2	2	3	1
Henry Robinson	4	3	2	1	3	3	4	1
Nergis Erturk	3	2	3	3	4	2	3	
Surapa T.	4	1	1	2	2	4	3	2
Thorstein Bostad	1	1	3	3	2	2	1	1
Zeynep Fetvaci	3	3	3	4	2	4	4	1
aslavin	1	2	3	2	3	3	4	1
da209	2	3	1	3	2	3	3	1
erikalopez	1	2	4	2	3	3	3	1
james allen	1	1	2	1	2	4	4	2
jay silver	2	3	3	4	4	4	4	4
jmahowald	1	2	1	1	3	2	4	2
leah hoffmann	4	3	4	2	3	3	3	2
vidura	4	1	3	1	3	3	3	2
zeynepwindows	1	2	2	4	3	3	4	1

Table 10: Results of the Preference Test

In the preference test, 151 choices were made (one person stopped before the last utterance). The breakdown of the number of choices per method is as follows:

	Baseline	Cubic	CB (utterance)	CB (VS)
Number of times method was picked as first choice	33	39	50	29

As expected, the distribution of choices are slightly closer to uniform in this test. However, there is still a clear preference for the utterance code book method. Even in cases where the cubic conversion function produced a lower objective distortion (e.g.utterance 3), most listeners preferred the code book method over the cubic. (8 preferences versus 4)

Another interesting result to note is that, even though the cubic conversion improved the baseline consistently in objective measurements, these improvements were clearly not perceptually distinguishable since the number of baseline choices and cubic conversion choices were close to equal.

While the utterance based codebook method was favored by most listeners, the voiced segment based codebook method was not selected very often apart from utterance 7 which was a training utterance. This could be due to a number of reasons: First of all, interpolating between different voiced segments may yield better results as it did for the utterance based approach. Secondly, utterances converted using this method may have

unexpected intonations due to the concatenative nature of the algorithm. Because we lose utterance level context, the converted contour ceases to be a true target contour but a concatenation of target intonation phrases, which may end up producing awkward sequences. This problem of course challanges the main idea behind the algorithm by showing that matching the source contours perfectly doesn't always result in generating a reasonably natural target intonation.

Overall, we conclude that the perceptual tests support objective measurements in the cases where listeners are asked to compare conversions to the original target contour. When they are simply asked to pick the most natural contour, listeners were leaning towards the conversions output by the utterence code book method even in cases where its objective distortion measure was not the lowest of all methods. This is possibly due to the fact that imparting a real target contour is probably more natural than adjusting the pitch range of the source speaker. Given that with a codebook of more utterances, the performance of this algorithm can only improve, we would strongly suggest using the utterance code book method, where there is sufficient training data available.

7 Conclusion

This project implemented and evaluated four pitch conversion algorithms in a pitch transplantation framework. The first two aimed to convert pitch values on a frame by frame basis, while the last two attempted to capture finer intonational details of the target speaker by imparting an entirely new contour onto the input utterance. While most of the algorithms improved the baseline mean-variance method according to objective measures, the rates of improvement were closely dependent on the nature of the training data. We saw that when the training utterances of the source speaker contain much freer intonation patterns, and the target much more restrained ones, the utterance based code book method performed significantly better than all other methods. Thereofore, for voice conversion applications hoping to take advantage of the more complex pitch conversion algorithms introduced here, we suggest that training data is classfied in terms of the range and variation of intonational patterns before choosing an algorithm. This process of classification could be an independent project on its own.

More work can be done to improve to perceptual naturalness of conversions based on a codebook of voiced segments. Interpolating among codebook entries is one such next step. Another step could be to parametrise some of the global characteristics of the utterances, such as the slope of the declination line, and map these as well as pitch groups. This would potentially implant some utterance level cohesion to the contour and possibly improve the problem of random concatenation of voiced segments.

In general, we found that mapping fine details of the intonation patterns from one speaker to another is a very difficult problem due to the variability of contours that can be generated by any given speaker. Therefore, without the meaning of the message or the context in which it was uttered, there is only so much that can be done to map entire contours. However, if syntactic and semantic information were to be included in this process, this may result in greater improvements.

References

[1] Yannis Stylianou, Oliver Cappe, Eric Moulines, "Continous Probabilistic Transform For Voice Conversion", IEEE Transactions On Speech and Audio Processing, vol. 6, No.2, March 1998.

[2] Alexander Kain, Michael W.Macon, "Spectral Voice Conversion For Text-To-Speech Synthesis.",Proc IEEE ICASSP, Seattle

[3] Levent Arslan, "Speaker Transformation Algorithm using Segmental Codebooks (STASC).", Speech Communication, 1999.

[4] David T. Chappell, John H. L. Hansen, "Speaker Specific Pitch Contour Modelling and Modification", Proc. ICASSP, Seattle, USA, May 1998, pp. 885-888.

[5] Tim Ceyssens, Werner Verhelst,Patrick Wambacq, "On the Construction of a Pitch Conversion System", Proc. EUSIPCO, Toulouse, France, September 2002.

[6] K. Tokuda, T. Masuko, N. Miyazaki, T. Kobayashi, "Hidden Markov Models Based on Multi-Space Probability Distribution For Pitch Pattern Modeling", Proc. ICASSP-99, Mar 1999, pp. 229-232.

[7] M.Tamura, T.Masuko, K.Tokuda, T.Kobayashi, " Adaptation of Pitch and Spectrum for HMM-based Speech Synthesis Using MLLR", Proc. ESCA/COCOSDA Third International Workshop on Speech Synthesis, 1998.

[8] Alexander Kain "High Resolution Voice Transformation", PhD Thesis, OGI School of Science and Engineering at Oregon Health and Science University.

[9] Hui Ye, Steve Young "A Simplified Pitch Synchronous Harmonic Model for Speech Synthesis and Modification", Eurospeech.

[10] B. George, M. Smith, M. J. T., "Speech Analysis/Synthesis and Modification Using an Analysis-by-synthesis/overlap-add Sinusoidal Model", IEEE Trans. on Speech and Audio Proc., vol.5, no. 5, pp. 389-406, September 1997.

[11] Werner Verhelst, Tim Ceyssens, Patrick Wambacq, "On Inter-Signal Transplantation of Voice Characteristics", Proc. 3^rd IEEE Benelux Signal Processing Symposium (SPS-2002), Leuven, Belgium, March 21-22,2002.

[12] Oytun Turk, "New Methods For Voice Conversion", PhD Thesis, Bogazici University.

[13] Olivier Cappé, **H2M Matlab Toolbox**, downloadable at http://www-sig.enst.fr/ cappe/h2m/h2m.html

[14] Werner Verhelst, Tim Ceyssens, Patrick Wambacq, "A strategy for pitch conversion and its Evaluation", Proc. 3^rd IEEE Benelux Signal Processing Symposium (SPS-2002), Leuven, Belgium, March 21-22,2002.

[15] Xuedong Huang, Alex Acero, Hsiao-Wuen Hon, "Spoken Language Processing. A Guide to Theory, Algorithm and System Development", Prentice Hall, 2001.

[16] Thomas F. Quatieri, "Discrete Time Speech Signal Processing, Principles and Practice", Prentice Hall, 2002.

[17] John A. Rice, "Mathematical Statistics and Data Analysis", Duxbury Press, 1995.

[18] S. Young; J. Odell; D. Ollason; V. Valtchev; P. Woodland, The HTK Book (Version 2.1), Cambridge University Department and Entropic Research Laboratories Inc., 1997.

Appendix

A Recorded Text Material

To further his prestige he occasionally reads the wall street journal.
Smash light bulbs and their cash value will diminish to nothing.
Gregory and Tom chose to watch cartoons in the afternoon.
Would a tomboy often play outdoors?
The advertising verse of plymouth variety store never changes.
Roy ignored the spurious data points in drawing the graph.
As a percaution the outlaws bought gunpowder for their stronghold.
Jeff thought you argued in favor of a centrifuge purchase.
Put the butcher block table in the garage.
The haunted house was a hit due to outstanding audiovisual effects.
Shell shawk caused by schrapnel is sometimes cured through group therapy.
That noise problem grows more annoying each day.
The legislature met to judge the state of public education.
Cliff was soothed by the luxurious massage.
It's healthier to cook without sugar.
Al received a joint appointment in the biology and the engineering departments.
Approach your interview with statuesque composure.
Our plans right now are hazy.
Herb's birthday occurs frequently on thanksgiving.
Young children should avoid exposure to contagious diseases.
Gus saw pine trees and redwoods on his walk to *seqolia* national forest.
Read verse outloud for pleasure.
Why buy oil when you always use mine?
The baby puts his right foot in his mouth.
Steve wore a bright red cashmere sweater.
The beauty of the view stunned the young boy.
Which church do the Smiths worship in?
John's brother repainted the garage door.
A huge tapestry hung in her hallway.
The small boy put the worm on the hook.
The ground hog clearly saw his shadow but stayed out only a moment.
The rose coursage smelled sweet.
A moth zigzagged along the path through Otto's garden.
The baracuda recoiled from the serpent's poisonous fangs.
The oasis was a mirage.
The rich should invest in black zercons instead of stylish shoes.
Trish saw hours and hours of movies saturday.
The eastern coast is a place for pure pleasure and excitement.
Challenge each general's intelligence.
Dunk the stale biscuits into strong drinks.
Employee lay-offs coincided with the company's reorganization.

Even I occasionaly get the monday blues.
Footprints showed the path he took up the beach.
The dark merky lagoon wound around for miles.
Did you buy any corduroy overalls?
Cement is measured in cubic yards.
The patient and the surgeon are both recuperating from the lengthy operation.
I gave them several choices and let them set the priorities.
Alice's ability to work without supervision is noteworthy.
The angry boy answered but didn't look up.
What does it smell like in the garden?
Will you go to the may ball?
I couldn't believe he said that to you.
Put your hands down and leave me alone.
I bought a few apples, bananas, oranges and even mangos.
Where did you buy your shoes?
Is your friend a vegeterian?
Do you think you will be ready for the presentation at noon?