Applying Geometric Source Separation for Improved Pitch Extraction in Human-Robot Interaction

Martin Heckmann, Claudius Gläser, Frank Joublin, Kazuhiro Nakadai

2010

Preprint:

This is an accepted article published in Proc. INTERSPEECH. The final authenticated version is available online at: https://doi.org/[DOI not available]

Applying Geometric Source Separation for Improved Pitch Extraction in Human-Robot Interaction

Martin Heckmann[1]*, Claudius Gläser*[1]*, Frank Joublin*[1]*, Kazuhiro Nakadai*[2]

[1]Honda Research Institute GmbH, D-63073 Offenbach/Main, Germany
[2]Honda Research Institute Japan Co. Ltd., Wako-shi, Saitama 351-0188, Japan
{martin.heckmann, claudius.glaeser, frank.joublin}@honda-ri.de, nakadai@jp.honda-ri.com

Abstract

We present a system for robust pitch extraction in noisy and echoic environments consisting of a multi-channel signal enhancement, a pitch extraction algorithm inspired by the processing in the mammalian auditory system and a pitch tracking based on a Bayesian filter. For the multi-channel signal enhancement we deploy an 8 channel Geometric Source Separation (GSS). During pitch extraction we first apply a Gammatone filter bank and then calculate a histogram of zero crossing distances based on the band-pass signals. While calculating the histogram spurious side peaks at harmonics and sub-harmonics of the true fundamental frequency are inhibited. The grid based Bayesian tracker operating on the resulting histogram comprises a Bayesian filtering in a forward step and Bayesian smoothing in a backward step on a 100 ms time window. We evaluate the system in a realistic human-robot interaction scenario with several male and female speakers. The evaluation is based on the degradation of the pitch tracking results obtained from the signals recorded on the robot to those of a simultaneously recorded clean headset signal. Hereby, we also include the comparison to two well established pitch extraction frameworks, i. e. get_f0 included in the WaveSurfer Toolkit and Praat. Overall the results demonstrate that pitch tracking with small errors is possible in all cases tested and that the proposed system performs better than the two benchmark algorithms.

1. Introduction

Despite many algorithms presented in the past, reliable extraction of fundamental frequency, whose percept is called pitch, in acoustically adverse environments remains difficult. In this paper we present a system for robust pitch extraction in a realistic human-robot interaction scenario where echoes and noise degenerate the speech signal captured by the robot. The system comprises three main building blocks. The first is a Geometric Source Separation (GSS) which enhances the signal. The second step is an algorithm for pitch extraction which takes inspirations from models of human pitch perception. It is based on the calculation of a histogram of zero crossing distances after transformation of the signal in the frequency domain via application of a Gammatone filter bank. The final step is the deployment of a Bayesian tracking algorithm on the resulting histograms. An overview on the system is given in Fig. 1.

In the following we will detail the building blocks of the proposed system for pitch extraction. After this we will give an overview on the human-robot interaction scenario in which we tested our algorithm. We evaluate the performance of the system based on a comparison of the tracking performance obtained on a clean headset signal and the signals recorded on the robot in comparison to two commonly used pitch extraction frameworks. A discussion of the results will conclude the paper.

2. Geometric Source Separation

We used an online version of Geometric Source Separation (GSS) [1] for sound source separation.

A spectrum vector of *M* sources and a spectrum vector of signals captured by the *N* microphones at frequency ω are denoted as $\mathbf{s}(\omega)$ and $\mathbf{r}(\omega)$, respectively. The spectrum vectors are obtained by application of the Fast Fourier Transform (FFT) on the time domain signals $\mathbf{s}(t)$ and $\mathbf{r}(t)$. The source separation is then formulated as

$$\mathbf{y}(\omega) = \mathbf{W}(\omega)\mathbf{r}(\omega), \tag{1}$$

where $\mathbf{W}(\omega)$ is called a *separation matrix*. The separation is defined as finding $\mathbf{W}(\omega)$ which satisfies the condition that output signal $\mathbf{y}(\omega)$ is the same as $\mathbf{s}(\omega)$. In order to estimate $\mathbf{W}(\omega)$, GSS introduces two cost functions, that is, separation sharpness (J_{SS}) and geometric constraints (J_{GC}) defined by

$$J_{SS}(\mathbf{W}) = \|E[\mathbf{y}\mathbf{y}^H - \mathrm{diag}[\mathbf{y}\mathbf{y}^H]]\|^2 \tag{2}$$

$$J_{GC}(\mathbf{W}) = \|\mathrm{diag}[\mathbf{W}\mathbf{D} - \mathbf{I}]\|^2 \tag{3}$$

where $\|\cdot\|^2$ indicates the Frobenius norm, $\mathrm{diag}[\cdot]$ is the diagonal operator, $E[\cdot]$ represents the expectation operator and H represents the conjugate transpose operator. $\mathbf{D}$ is a transfer function matrix based on a direct sound path between a sound source and each microphone. $\mathbf{W}$ at the current time step t, $\mathbf{W}_t$, is estimated recursively to minimize these cost functions as follows:

$$\mathbf{W}_{t+1} = \mathbf{W}_t - \mu_{SS}\mathbf{J'}_{SS}(\mathbf{W_t}) + \mu_{GC}\mathbf{J'}_{GC}(\mathbf{W_t})$$

$$\mathbf{J'}_{SS}(\mathbf{W_t}) = 2\mathbf{E}_{SS}\mathbf{W_t}\mathbf{r}\mathbf{r}^H \text{ and } \mathbf{J'}_{GC}(\mathbf{W_t}) = \mathbf{E}_{GC}\mathbf{D}^H,$$

where $\mathbf{J'}(\mathbf{W})$ is an update direction of $\mathbf{W}$ derived from its complex gradient [2]. μ_{SS} and μ_{GC} are step-size parameters.

For further processing the source from the frontal direction is chosen and transformed back into the time domain via application of the Inverse Fast Fourier Transform (IFFT).

3. Pitch Estimation

The algorithm we apply for pitch extraction relies on the calculation of a histogram of zero crossing distances and a subsequent inhibition of side peaks resulting form harmonics and sub-harmonics of the true fundamental frequency [3]. It combines information residing in the spectral and the temporal domain following inspirations from different human pitch perception models [4].

The first step of the pitch extraction is the transformation of the signal resulting from the GSS into the frequency domain via

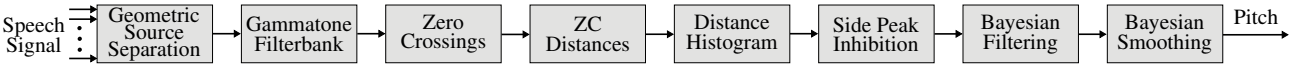

Figure 1: System overview

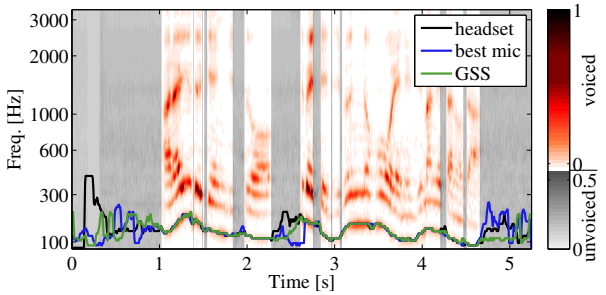

Figure 2: Signal resulting from the GSS after application of the Gammatone filter bank. A male speaker is uttering the Japanese sentence "a r a y u r u g e N j i ts u w o s u b e t e j i b u N n o h o u e n e j i m a g e t a n o d a". The sentence contains a high proportion of vowels and voiced consonants and translates to "every fact was biased towards its preference". Pitch tracks for the clean and noisy signals are shown. Unvoiced regions are marked in gray.

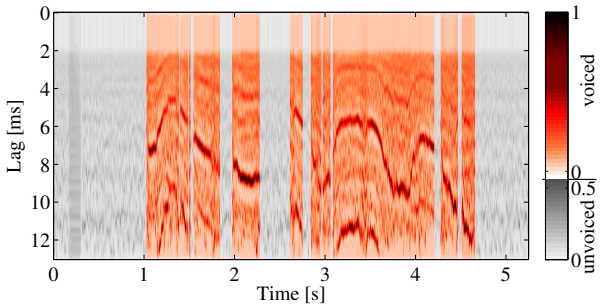

Figure 3: Histogram of the zero crossing distances (ZCD) for the signal in Fig. 2 prior to the inhibition of the side peaks. Unvoiced regions are marked in gray.

a Gammatone filter bank (see Fig. 2).

3.1. Extracting Temporal Information

Commonly the autocorrelation function is used to extract temporal information. As the autocorrelation is very time consuming and not supported by biological data [5] we use in our system the zero crossing distances (ZCD) in the signal. Let $C_i = [t_{i,1}, t_{i,2}, \ldots, t_{i,N}]$ denote the ordered sequence of the time indices of all rising zero crossings, i. e. from negative to positive, in the band pass signals $g_i(t)$ in the i-th channel of the Gammatone filter bank:

$$C_i(m) = t_{i,m} \text{ with } g_i(t_{i,m}-1) < 0 \wedge g_i(t_{i,m}) \geq 0, \forall m. \quad (4)$$

Then the sequence of zero crossing distances is defined by

$$D_i(m) = C_i(m+1) - C_i(m). \quad (5)$$

Based on this, a signal $d_i(t)$ is constructed which has in the interval between two zero crossings as its value the zero crossing distance. Hence $d_i(t) = D_i(m)$ where m is chosen such that $C_i(m) \leq t < C_i(m+1)$. This distance between adjacent zero crossings, more precisely its inverse, codes the frequency of the signal.

3.2. Extracting Spectral Information

We asses the spectral information via a comb filter with teeth at the locations of the harmonics. In a scan through all possible fundamental frequencies $f_{\min} \leq f_0 \leq f_{\max}$ the corresponding comb filters are set up. For each of these comb filters the allocation of the teeth with harmonics of the current fundamental can be checked at each instant in time. The "filter response" of the comb filter is calculated based on the found allocation pattern. The better the found pattern matches the expected pattern the higher the response (see [3] for more details).

3.3. Combining temporal and spectral information

Instead of using the energy in the teeth, i. e. the channels of the Gammatone filter bank, we use the zero crossing distances $d_i(t)$ in the channels to determine if a tooth is set. The ZCDs provide an instantaneous frequency estimate of the band-pass signal. In a harmonic signal the ZCD of the first harmonic are half the distances of the fundamental and the distance of the second harmonic a fourth those of the fundamental. Based on this simple relation one can set up a rule to compare the found distance to an expected one. In the scan through all fundamental frequencies we can compare the ZCD $d_i(t)$ at time instant t and channel i against the current harmonic hypothesis $f_h = k \cdot f_0$ and the corresponding ZCD $d_h = \frac{f_S}{k \cdot f_0}$, with f_S being the sampling rate. The deviation $\Delta = d_h - d_i(t)$ is a measure for the match of the current channel and the hypothesis. To determine if a tooth is set we apply a threshold $\Delta_t = 0.04 d_h$. Summing up over all teeth of the comb filter and dividing by the number of teeth yields a normalized match value $m(t, f_0)$ for the current fundamental frequency hypothesis f_0. The fundamental frequency hypothesis f_0 and the distance hypothesis $d_0 = \frac{f_S}{f_0}$ can be used interchangeably. To facilitate comprehension we will in the following only use f_0. However, one has to keep in mind that the actual scan through all fundamental frequency hypotheses is realized by decrementing the current distance hypothesis d_0 by the minimal decrement determined by the sampling rate $\delta_d = \frac{1}{f_S}$ and then calculating the corresponding f_0. In addition also the test if a tooth is set is performed in the distance domain. The resulting histogram $\mathbf{h}'$ is displayed in Fig. 3. Unfortunately, spurious peaks at harmonics and sub-harmonics of the true fundamental frequency occur due to partial matches of the comb filter at these harmonics and sub-harmonics. This behavior can also be observed when using the autocorrelation (see [3] for more details).

3.4. Inhibition of Side Peaks

To avoid the spurious peaks we introduce an inhibition of these partial matches based on their expected matches. Let $m(t, f_0)$ be the match of the current comb filter calculated as described above (compare also Fig. 4.a). Then $m_{1/2}(t, f_0)$ would be the match of a comb filter set up at $f_0/2$, i. e. corresponding to a true fundamental frequency f_0' of $f_0/2$, but which contains only the teeth that are not covered by the comb filter set up at f_0 (compare also Fig. 4.b). This quantifies to which extent $f_0/2$ would be a better match than f_0. Similar $m_2(t, f_0)$ would be the match of a comb filter for a true fundamental frequency f_0' of $2f_0$ and containing only the teeth which are to be expected to be missing in $m(t, f_0)$ (compare also Fig. 4.c). The inverse $v_2(t, f_0) = 1 - m_2(t, f_0)$ indicates how much $2f_0$ would be a better match than f_0. The final histogram value $h(t, f_0)$ is then

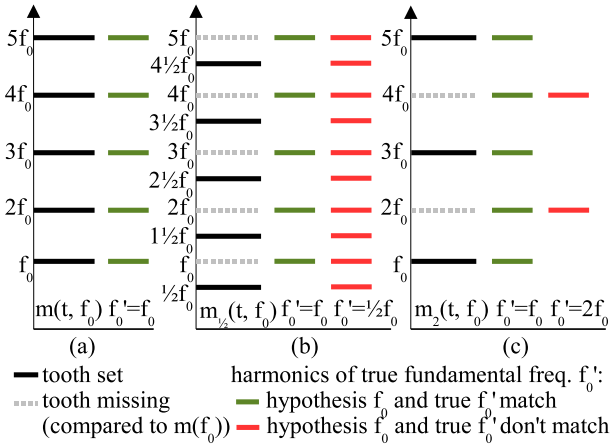

Figure 4: Visualization of the different comb filter patterns used to inhibit side peaks.

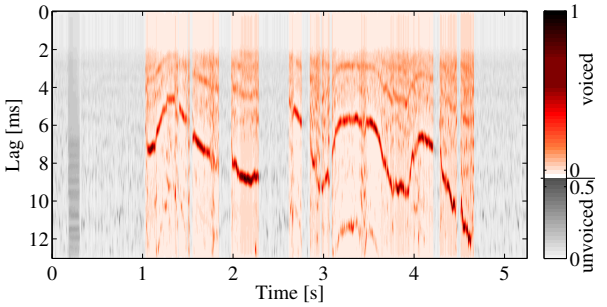

Figure 5: Histogram of the zero crossing distances (ZCD) for the signal in Fig. 2 after inhibition of side peaks. Unvoiced regions are marked in gray.

the full match reduced by the maximum of the partial matches:

$$h(t,f_0) = m(t,f_0) - w(f_0) \cdot \max[m_{1/2}(t,f_0), v_2(t,f_0), m_{1/3}(t,f_0), v_3(t,f_0), \dots]. \quad (6)$$

with $w(f_0) = \sqrt[3]{f_{0_{\min}}/f_0}$ being an inhibition weight dependent on the current f_0, hypothesis. When comparing Figs. 3 and 5 one can see that this inhibition step makes the true fundamental frequency much better visible and successfully reduces the most prominent side peaks.

4. Pitch Tracking

On the histogram $\mathbf{h}$ we apply a tracking algorithm based on Bayesian filtering [6]. We originally developed this algorithm for formant tracking and adapted it in [7] also to pitch tracking.

Bayesian trackers sequentially estimate the state x_t at time t conditioned on all information contained in the sensor data z_t [8]. Uncertainty is introduced by a probabilistic distribution over x_t, called the belief $Bel(x_t) = p(x_t|z_1, \dots, z_t)$.

Let $Bel^-(x_t)$ denote the predicted belief at time t which can be obtained via the application of the pitchs' underlying dynamics $p(x_t|x_{t-1})$. Then the belief at time t is calculated by correcting the predicted belief according to the observation from the pitch histogram $p(z_t|x_t)$ and a normalization factor α.

Since we want to estimate pitch locations on a discrete grid defined by the evaluated zero crossing distance values, a grid-based approximation of the belief is chosen. Thus, assuming that N distances are evaluated, the state space at time t can be written as $X_t = \{x_{1,t}, x_{2,t}, \dots, x_{N,t}\}$ which leads to the following Bayesian filter recursion:

$$Bel^-(x_{k,t}) = \sum_{l=1}^{N} p(x_{k,t}|x_{l,t-1}) Bel(x_{l,t-1}) \quad (7)$$

$$Bel(x_{k,t}) = \frac{p(z_t|x_{k,t}) Bel^-(x_{k,t})}{\sum_{l=1}^{N} p(z_t|x_{l,t}) Bel^-(x_{l,t})} \quad (8)$$

When operating in noisy conditions a subsequent backward pass on the already obtained filtering distributions $Bel(x_{k,t})$ is recommended since it significantly enhances the noise robustness of the algorithm. Bayesian smoothing provides such a mechanism. It aims to recursively estimate a smoothed version $\widehat{Bel}(x_{k,t})$ of the belief, thereby depending on both past and future observations. Its essence is the application of the filtering equations from Eqs. 7 in a given time window in reverse time direction.

The final calculation of exact pitch values $P(t)$ can easily be done by picking the peaks of the smoothed beliefs (see [6] for more details). In Fig. 2 the result of the Bayesian tracking are overlaid in blue on the spectrogram (the two other curves result from the two other setups explained in detail in the following section). During the tracking we model the a priori distribution $p(x_{k,0})$ and the pitch dynamics $p(x_{k,t}|x_{l,t-1})$ with normal distributions.

5. Evaluation

To asses the performance of our algorithm we evaluated it in a human-robot interaction scenario. Different people spoke to the Honda humanoid robot at a natural interaction distance of 1.5 m in a 4 m × 7 m room with RT_{20} = 300 ms[1]. 2 female and 6 male speakers were uttering a total of 90 utterances with 10-16 utterances per speaker. The speech signals was captured by an 8 channel microphone array mounted on the head of the robot. We compared the performance of our algorithm to the two publicly available and commonly used pitch tracking frameworks get_f0 from ESPS in the implementation of the WaveSurfer toolkit [9, 10] and praat [11]. Both frameworks are based on an autocorrelation calculated from the full-band signal. They also include a voicing detection and output pitch only for voiced segments. Because the voicing detection is rather unreliable for noisy speech we changed the parameterization such that the whole segment was classified as voiced and hence pitch was continuously calculated.

For the evaluation we also simultaneously recorded the speech signals with a headset and used this signal to calculate the ground truth information for the fundamental frequency. The following results are given as deterioration of the tracking results relative to this assumed ground truth. Hence the validity of the results partially depends on the correctness of the pitch extracted from the headset signal. However, visual inspection of the extracted pitch showed that it is extracted very accurately from the headset signal. As pitch is only present in voiced regions of speech an additional voiced/unvoiced detection is necessary for the performance evaluation. To detect voiced regions we use the voicing detection algorithm included in get_f0. In order to increase the robustness of the detection we additionally rejected segments with very low energy ($\approx 0.5\%$ of the mean energy). We applied this algorithm on the headset signal and used this information also for the noisy signals recorded on the robot. Consequently pitch tracking results were only evaluated in regions where voicing was detected in the headset signal.

After application of the GSS signals were downsampled to

[1] RT_{20} is better suited for measurements in noisy environments. It gives the decay measured at 20 dB extrapolated to 60 dB decay

16 kHz. For the pitch tracking we used a 100 channel Gammatone filter bank in the implementation according to [12] with frequencies in the range from 80-5000 Hz. The range of possible fundamental frequencies was set to 80-500 Hz. We calculated zero crossing distances up to the order 7 and used a comb filter with 15 teeth. The Bayesian smoothing operated on a 100 ms time window.

To differentiate the impact of the multi-channel signal enhancement from the pitch extraction and tracking algorithm we compared two different setups. In the first setup we use the microphone signal with the highest SNR. As all speakers were speaking approximately from the front to the robot the SNR was always highest for the microphone mounted on the front (referred to in the following as *best mic*). A typical SNR value for this setup is $\approx 15\,\mathrm{dB}$ (compare to $\approx 35\,\mathrm{dB}$ for the headset).[2] In the second setup we evaluate the pitch tracking after the application of the GSS algorithm. The GSS improved the SNR $\approx 4\,\mathrm{dB}$ compared to the best mic condition.

In Table 1 the tracking errors relative to the headset signal are shown. The tracking performance of each algorithm in the noisy conditions is evaluated against the headset condition extracted by the same algorithm. Tracking errors are ceiled to 100%, i. e. errors larger than 100% are set to 100%.

Table 1: Pitch tracking errors relative to the headset signal in %.

	best mic	GSS	GSS+Post Filter
get_f0	2.6	7.1	7.2
praat	2.6	3.5	4.6
proposed	2.1	1.5	2.2

Additionally, we also evaluated the so called Gross Pitch Error (GPE) [13]. It measures how much of the pitch track deviates more than e_t from the true pitch. In our case we set $e_t = 20\%$. The corresponding values are given in Table 2.

Table 2: Gross pitch errors ($> 20\%$) relative to the headset signal in %.

	best mic	GSS	GSS+Post Filter
get_f0	1.8	2.7	2.9
praat	1.0	1.3	2.5
proposed	0.7	0.3	1.0

The results show that the tracking errors already for the best mic configuration are very good for all algorithms. The GSS preprocessing notably reduces the errors for our algorithm. However, the results for get_f0 and praat were deteriorated by the GSS. When using the GSS as preprocessing combined with our algorithm the errors are very small and only very little gross pitch errors occur.

The GSS based signal enhancement proposed in [1] also includes a multi-channel post filtering step. The post filter is applied after the GSS and has as its purpose to reduce the noise still present after the GSS step. In addition to the stationary components of the noise it also estimates non-stationary components and subtracts them from the signal. We investigated a setup where we included the post filter as described in [1]. When comparing Table 1 and 2 one can see that the post filtering is not beneficial for the pitch tracking for all algorithms.

[2] We calculated the SNR as the ratio of the energy of the segments containing only speech to those containing only noise. Signal distortions due to reverberations are hereby not taken into account.

6. Conclusion

The evaluation showed that the pitch extraction already yields good results without the preprocessing for all algorithms. The combination of GSS and our pitch tracking algorithm further improved the results significantly. For get_f0 and praat it deteriorated the results. The phase changes resulting from the GSS might be the reason for this behavior. In contrast our algorithm is insensitive of the phase of the different harmonics. The application of an additional post filter decreased performance in all cases. This might be due to distortions resulting from the post filtering, e. g. musical tones due to incorrect estimation of either noise or signal energy.

In the best case, i. e. using the GSS but without post filtering and our pitch extraction framework, we obtain relative errors averaged over all speakers below 2% and gross pitch errors of only 0.3%. From this we conclude that the system we propose robustly extracts the fundamental frequency and hence lays the foundation for a prosodic analysis of the speech signal.

7. Acknowledgments

We want to thank Dr. Shun'ichi Yamamoto for support with the GSS algorithm and for designing and performing the recordings.

8. References

[1] S. Yamamoto, K. Nakadai, J.M. Valin, J. Rouat, F. Michaud, K. Komatani, T. Ogata, and HG Okuno, "Making a robot recognize three simultaneous sentences in real-time," in *Proc IEEE/RSJ Int. Conf. on Robots and Intell. Syst. (IROS)*, Edmonton, Canada, 2005, pp. 4040–4045.

[2] D.H. Brandwood, "A complex gradient operator and its application in adaptive array theory," *IEE Proc.*, vol. 130, no. 1, pp. 251–276, 1983.

[3] M. Heckmann, F. Joublin, and C. Goerick, "Combining rate and place information for robust pitch extraction," in *Proc. INTERSPEECH*, Antwerp, 2007, pp. 2765–2768.

[4] A. de Cheveigne, "Pitch perception models," in *Pitch*, C. Plack and A. Oxenham, Eds. Springer, Cambridge, U.K., 2004.

[5] C. Kaernbach and L. Demany, "Psychophysical evidence against the autocorrelation theory of auditory temporal processing," *Journal of the Acoustic. Soc. of America*, vol. 104, pp. 2298–2306, 1998.

[6] C. Gläser, M. Heckmann, F. Joublin, and C. Goerick, "Combining auditory preprocessing and bayesian estimation for robust formant tracking," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 2, pp. 224–236, 2010.

[7] M. Heckmann, C. Gläser, M. Vaz, T. Rodemann, F. Joublin, and C. Goerick, "Listen to the parrot: Demonstrating the quality of online pitch and formant extraction via feature-based resynthesis," in *Proc. IEEE/RSJ Int. Conf. on Intell. Robots and Systems (IROS)*, Nice, 2008, IEEE-RSJ.

[8] D. Fox, J. Hightower, L. Liao, D. Schulz, and G. Borriello, "Bayesian Filtering for Location Estimation," *IEEE Pervasive Computing*, vol. 2, no. 3, pp. 24–33, 2003.

[9] D. Talkin, "A robust algorithm for pitch tracking (RAPT)," *Speech coding and synthesis*, vol. 495, pp. 518, 1995.

[10] K. Sjölander and J. Beskow, "Wavesurfer-an open source speech tool," in *Sixth Int. Conf. on Spoken Lang. Proc. (ICSLP)*, 2000.

[11] Paul Boersma and David Weenink, "Praat: doing phonetics by computer (v. 5.1.21) [computer program].," November 2009.

[12] M. Slaney, "An efficient implementation of the Patterson-Holdsworth auditory filterbank," Tech. Rep., Apple Computer Co., 1993, Technical report #35.

[13] L. Rabiner, M. Cheng, A. Rosenberg, and C. McGonegal, "A comparative performance study of several pitch detection algorithms," *IEEE Trans. on Acoustics, Speech and Signal Proc.*, vol. 24, no. 5, pp. 399–418, 1976.