Friedrich-Alexander-Universität Erlangen-Nürnberg

Lab Course

Pitch Estimation

International Audio Laboratories Erlangen

Prof. Dr.-Ing. Bernd Edler

Friedrich-Alexander Universität Erlangen-Nürnberg
International Audio Laboratories Erlangen
Lehrstuhl Semantic Audio Processing
Am Wolfsmantel 33, 91058 Erlangen

bernd.edler@audiolabs-erlangen.de

International Audio Laboratories Erlangen
A Joint Institution of the
Friedrich-Alexander Universität Erlangen-Nürnberg (FAU) and
the Fraunhofer-Institut für Integrierte Schaltungen IIS

Authors:

Stefan Bayer,
Nils Werner,

Tutors:

Nils Werner,
Christian Helmrich,

Contact:

Nils Werner, Christian Helmrich,
Friedrich-Alexander Universität Erlangen-Nürnberg
International Audio Laboratories Erlangen
Lehrstuhl Semantic Audio Processing
Am Wolfsmantel 33, 91058 Erlangen
`nils.werner@audiolabs-erlangen.de`
`christian.helmrich@audiolabs-erlangen.de`

Lab Course

Pitch Estimation

Abstract

When looking at audio signals, one possible signal model is to distinguish between harmonic components and noise like components. The harmonic components exhibit a periodic structure in time and it is of course of interest to express this periodicity via the fundamental frequency F_0, i.e. the frequency of the first sinusoidal component of the harmonic source. This fundamental frequency is closely related to the so called pitch of the source. The pitch is defined as how "low" or "high" a harmonic or tone-like source is perceived. Although strictly speaking this is a perceptual property, and is not necessarily equal to the fundamental frequency, it is often used as a synonym for the fundamental frequency. We will use the term pitch in this way in the remaining text. It is also of interest how the relationships in terms of energy between the harmonic and noise like components of an audio signal are. One feature expressing this relationship is the Harmonic to Noise Ratio (HNR). The estimation of the pitch and the HNR then can be used e.g. for efficiently coding the signal, or to generate a synthetic signal based on this and other information gained from analysing the signal. In this laboratory we will concentrate on a single audio source, and we will restrict ourselves to speech, which is the primary mode of human interaction. We will use this signals to develop simple estimators for both features and compare the results to state-of-the-art solutions for estimating the pitch and the HNR.

1 Pitch Estimation

As stated above, we model an audio, or to be more specific, an speech signal as a mixture of a harmonic signal and a noise signal:

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

where $s(t)$ is the speech signal, $h(t)$ is the harmonic component, and $n(t)$ ist the noise component. For time-discrete signal (and in digital signal processing of course we deal with such time-discrete signals) the equation becomes:

$$s[k] = h[k] + n[k] \tag{2}$$

k being the samples index.

In this section we will have a closer look at the harmonic component $h(t)$, which can be expressed as the sum of its partial tones, which are sinusoidals where the frequencies of the individual partial tones are integer multiples of the fundamental frequency:

$$h(t) = \sum_{n=1}^{N} a_n sin\left(\frac{2\pi n}{F_0} + \phi_n\right) \tag{3}$$

where a_n are the individual amplitudes and ϕ_n are additional phases for the individual partial tones. Unfortunately in real world signals like speech typically neither the amplitudes nor the fundamental frequency stay constant over the whole duration of the signal. But when looking closer at for e.g. speech, we see that this parameters normally only change slowly over time. This behaviour gives us the possibility to assume that the parameters stay constant if we compart the signals into small enough sections in time. Such signals are called *quasi-stationary*. So the first step towards a pitch estimation is to divide the signal into small enough blocks. The length of the block is determined by the lowest pitch we like to detect, for most algorithms at least two periods of the signal should be contained within one block to give a reliable estimate. Table 1 gives a rough overview of the pitch ranges in human speech.

	lower limit	upper limit
male	75 Hz	150 Hz
female	125 Hz	250 Hz
child		600 Hz

Table 1: typical fundamental frequencies in human speech

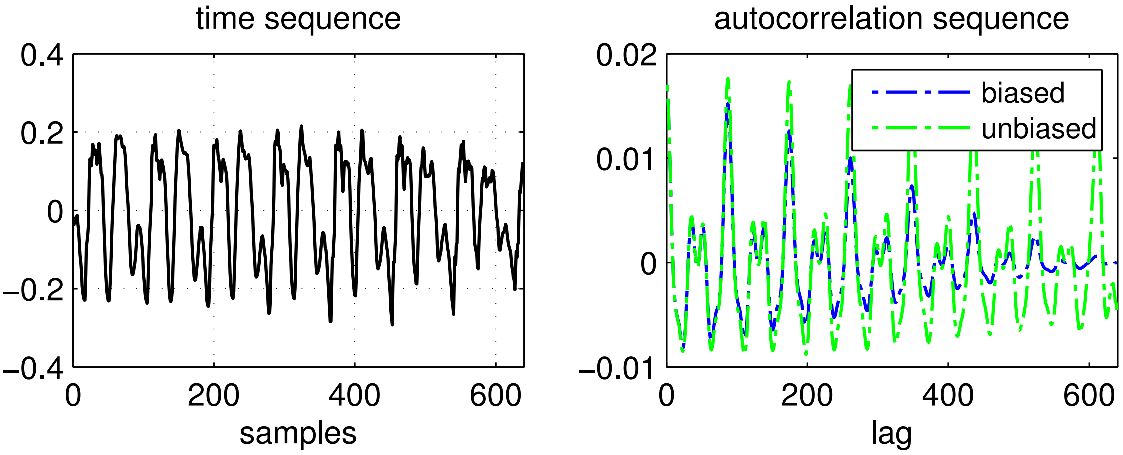


Figure 1: Comparison of the biased and unbiased autocorrelation sequence for a periodic signal (part of a vowel of a male speaker).

The simplest way would now be to just use the zero crossings of the signal. But although this method is very efficient it is not well suited if higher partials have amplitudes or if the noise component is very strong.

So most pitch algorithms are based on other methods, for a simple overview go to [1].

In this laboratory we will develop a estimation algorithm based on the autocorrelation [2]. For discrete time signals the autocorrelation is defined as:

$$R_{xx}[l] = \lim_{N\to\infty} \frac{1}{N} \sum_{k=-N}^{N} x[k]x[k-l] \tag{4}$$

where l is the so called lag. Of course this is the definition for signals of infinite length, but we already divided our signal into blocks of length N each, so the autocorrelation becomes (in its *biased* form):

$$R_{xx}[l] = \frac{1}{N} \sum_{k=l}^{N-1} x[k]x[k-l] \tag{5}$$

We only consider positive lags since the resulting autocorrelation sequence is symmetric around $l = 0$. Another form of the autocorrelation is the so called *unbiased* autocorrelation sequence

$$R_{xx}[l] = \frac{1}{N-l} \sum_{k=l}^{N-1} x[k]x[k-l] \tag{6}$$

The difference between unbiased and biased autocorrelation is that the unbiased takes the decreasing number of samples involved in the summation into account. When looking at figure 1 we observe the difference between the biased and the unbiased autocorrelation, the biased tapers off towards high lags. When we compare the autocorrelation equations with our assumption that the signal is periodic with a periodicity $T_0 = 1/F_0$:

$$x[k] \approx [k + mT_0], m \in \mathbb{Z} \tag{7}$$

we see that for such a signal we can expect local maxima of the autocorrelation sequence for lags that are a multiple of T_0. By finding the maximum of the autocorrelation we get an estimate of the fundamental frequency. Note that the autocorrelation function always has a maximum at $l = 0$, so to not erroneously detecting the zero lag as maximum, it is wise to restrict the search within lags that correspond to the upper and lower limits of the fundamental frequency range under consideration. Also the found global maximum might not be at the lag corresponding to the true fundamental frequency but can possibly be an integer multiple of that. Furthermore note that due to this, the estimate can jump between lags in consecutive frames leading also to jumps in the F_0-estimate. For a more robust estimation this must be taken into account.

Homework Excercise 1

Pitch estimation: Theory

1. Given is the time sequence $x[k] = \{4, -2, -3, 1, 5, -1\}$. Calculate both the biased and unbiased autocorrelation sequences using pen and paper. Sketch the time and the autocorrelation sequence.
2. Calculate the necessary window length (both in ms and in samples for a sampling frequency of $f_s = 16000Hz$) for an autocorrelation based pitch estimator that should detect typical pitches for human speech as given in table 1.
3. Calculate the minimum and maximum lag in the autocorrelation domain for said estimator for the desired F_0 range.
4. What is the relationship between the autocorrelation and the power spectral density (PSD)?
5. Think about strategies to avoid octave jumps and errors in the autocorrelation based pitch estimation.

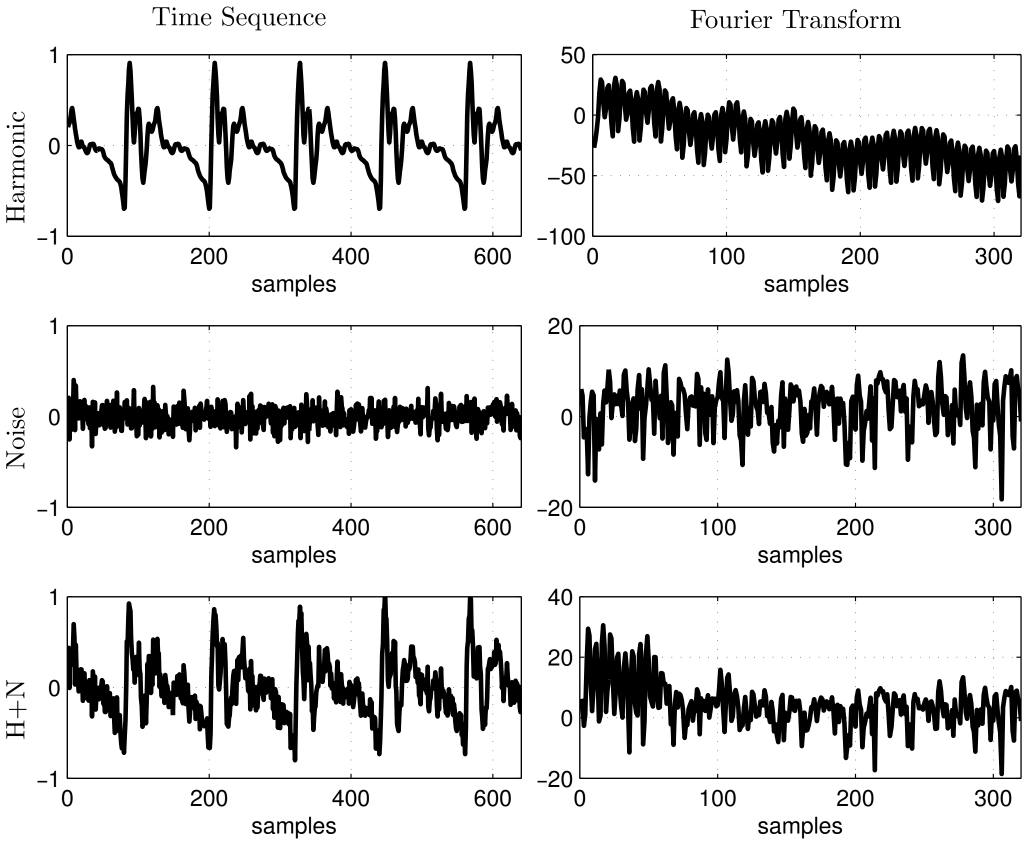


Figure 2: Example of a signal consisting of a harmonic part and a noise part.

2 Harmonic to Noise Ratio Estimation

We now go back to our signal model of equation 2. As already said, the relationship between the harmonic component $h(t)$ and the noise component $n(t)$ is of interest here. One possible characterization of that relationship is the ratio of the energies of both components, which we will call *harmonic to noise ratio.* The same assumption as above, i.e. that the energies of the components and therefore the HNR will vary over time, but slow enough so that it also can be assumed as constant for a small enough amount of time. If we know the exact nature of both $h[k]$ and $n[k]$ the HNR will then be:

$$HNR = \frac{\sum_{k=1}^{N} h[k]^2}{\sum_{k=1}^{N} n[k]^2} \tag{8}$$

Unfortunately, for a real world signal neither $h[k]$ nor $n[k]$ are known, so one has to think about how to estimate the HNR. For example see figure 2 where in the mixture of harmonic and noise components neither in the time sequence representation or the Fourier transformed representation a clear distinction can be seen between the harmonic and noise parts.

Lets start with some basic assumptions that will make life a little bit easier. We assume that $h[k]$ and $n[k]$ are uncorrelated, furthermore we assume that we already know F_0 of $h[k]$ and that $n[k]$ is white gaussian noise with zero mean. Now we will have a closer look at the autocorrelation and insert equation 2 into 6:

$$R_{xx}[l] = \frac{1}{N-l} \sum_{k=l}^{N-1} (h[k] + n[k])(h[k-l] + n[k-l]) \tag{9}$$

for $l = 0$ we would get the energy of the combined signal. No we look what happens for $l = T_0$ (by expanding the equation above):

$$R_{xx}[T_0] = \frac{1}{N-T_0}\Big(\sum_{k=T_0}^{N-1} h[k]h[k-T_0] + \sum_{k=T_0}^{N-1} h[k]n[k-T_0] + \sum_{k=T_0}^{N-1} h[k-T_0]n[k] + \sum_{k=T_0}^{N-1} n[k]n[k-T_0]\Big) \tag{10}$$

Under the assumptions from above (no correlation, white noise), the last three sums will be approximately zero, which leaves:

$$R_{xx}[T_0] = \frac{1}{N-T_0}\sum_{k=T_0}^{N-1} h[k]h[k-T_0] \tag{11}$$

We now insert the approximation of equation 7:

$$R_{xx}[T_0] = \frac{1}{N-T_0}\sum_{k=T_0}^{N-1} h[k]h[k] \tag{12}$$

and see that the autocorrelation at lag $l = T_0$ is approximately the energy of the harmonic component. Together with $R_{xx}[0]$ we can now calculate an estimation of the HNR:

$$HNR = \frac{R_{xx}[T_0]}{R_{xx}[0] - R_{xx}[T_0]}. \tag{13}$$

We now have found a nice estimate of the HNR that can be implemented very straightforward. In the literature many other approaches can be found, feel free to search for different algorithms and get some of the ideas, whether be it time-domain, time/frequency-transform based, or methods using the cepstrum [3].

Homework Excercise 2

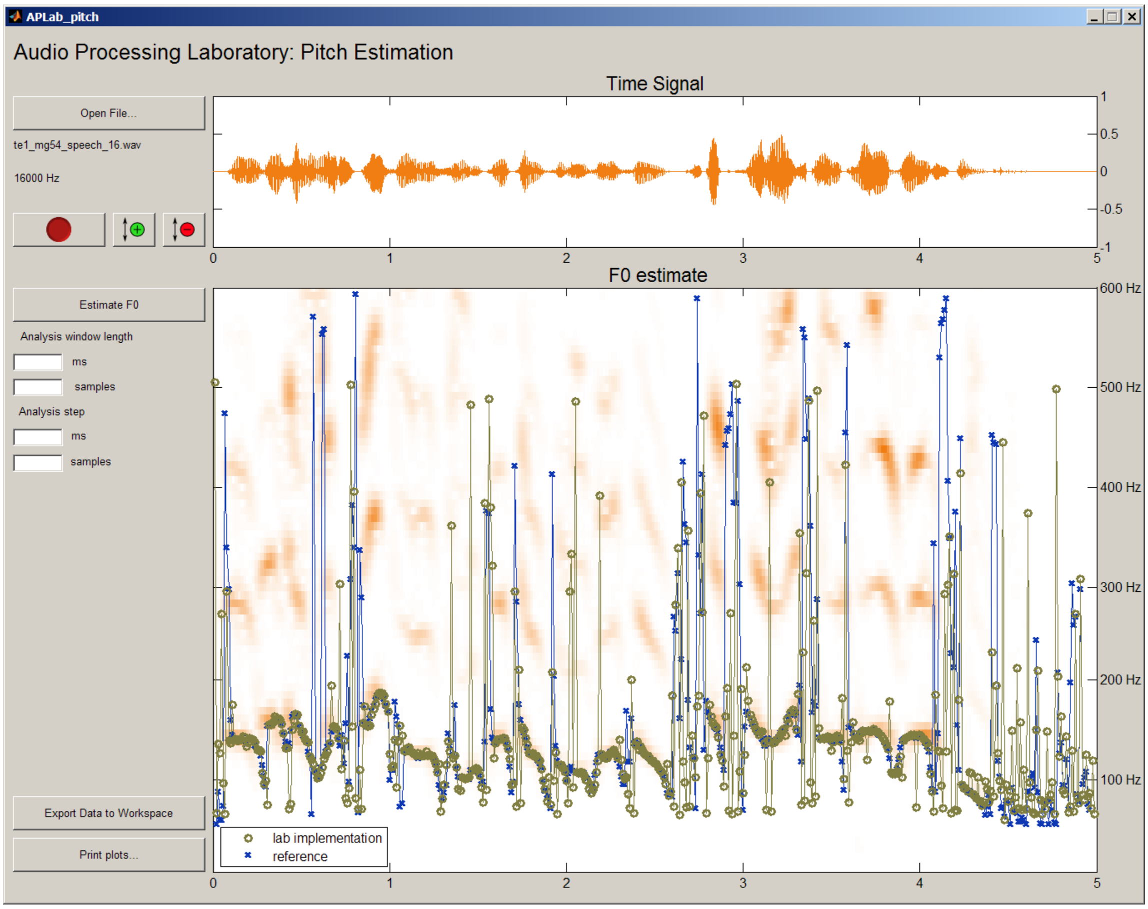
Harmonic to Noise Ration: Theory

1. Why can we assume that the last three sums in equation 10 are approximately zero under the stated terms that the noise is white and the noise and the harmonic component are uncorrelated?
2. Which autocorrelation should be used for the HNR estimation, the *biased* or the *unbiased*? Why?
3. Estimate the HNR for the sequence given in home work part 1 using the calculated autocorrelation and the estimation of equation 13 (Hint: take the position first maximum of the autocorrelation as T_0). If the result seems to be not in line with the theory find an explanation for that.
4. Search for or think about other possibilities to estimate the HNR.

3 The Experiment

3.1 Matlab based estimation

The Matlab directory contains stubs for the autocorrelation function, the F_0-estimation function, and the HNR estimation function called `autcorr.m`, `f0_estimation.m`, and `hnr_estimation.m`.



APLab_pitch

Audio Processing Laboratory: Pitch Estimation

te1_mg54_speech_16.wav

16000 Hz

Figure 3: Screenshot of the Matlab GUI for comparing the implemented pitch estimation against the given reference.

Furthermore for the evaluation of the pitch estimation against a given reference, a GUI called `APLab_pitch.m` exists. A screenshot of the GUI can be seen in figure 3. A similar GUI for the HNR estimation exists, called `APLab_hnr.m`. The subdirectory audiofiles contains several example audio files, you can bring your own files. Additionally, the GUIs allow to make recordings on the fly.

3.2 Exercises

Lab Experiment 1

Pitch Estimation: Instructions

1. Implement the autocorrelation of equations 5 and 6 in Matlab and compare the results for different signals to the Matlab function xcorr(). If the results differ, find an explanation for the difference.
2. Open f0_estimate.m in the Matlab editor.
3. Implement a first version of the F_0-estimator based on the comments in f0_estimate.m.
4. Compare the results using the APLab_pitch GUI to the results of the reference F_0 estimator.
5. Implement a refinement to reduce octave errors and jumps.
6. Compare the results using the APLab_pitch GUI to the results of the reference F_0 estimator.
7. Explain your solution.

Lab Experiment 2

Harmonic to Noise Ration Estimation: Instructions

1. Implement the HNR estimation derived in section 2 as Matlab function. For this use the already implemented functions for the pitch estimation.

2. Load the files synth_vowel_1.wav and synth_vowel_2.wav into the Matlab workspace. Both files contain synthetic vowels with the same HNR and same F_0. Calculate the HNR estimates for both signals using your implemented HNR estimation (Fs=16000, F0=100) on the complete items, if they differ, find an explanation. Note that for this exercise you should not use the APLab_HNR tool. (Hint: plotting the signals for inspection is always a good idea).

3. Compare the estimate to the reference estimate using the APLab_HNR tool.

4. Both the F_0 and HNR estimates are not reliable for certain signal portions, i.e. large variations can be observed. Why is this? And what solutions might be found to overcome this problem? Implement your solution.

References

[1] Wikipedia. Pitch detection algorithm. [Online]. Available: https://en.wikipedia.org/wiki/Pitch_estimation

[2] ——. Autocorrelation. [Online]. Available: https://en.wikipedia.org/wiki/Autocorrelation

[3] ——. Cepstrum. [Online]. Available: https://en.wikipedia.org/wiki/Cepstrum