Sound processing

Task 1

Fundamental frequency analysis

INTRODUCTION

One of the basic subjective features of a sound is *pitch*. From the objective point of view, the pitch is dependent on the structure of the sound frequency spectrum or – more precisely – on the *fundamental frequency* of the sound. In the simplest case, i.e. for a sound being a pure tone (a tone with a sinusoidal waveform), its frequency is directly equivalent to the subjective pitch. On the other hand, most of the sounds we perceive contain several partials (sine waves) each having different frequencies and amplitudes. If the frequency values of the partials are integer multiples of a given fundamental frequency, they form a harmonic frequency spectrum and are called harmonics. A sound with harmonic frequency spectrum is said to be of definite pitch[1], defined by its fundamental frequency.

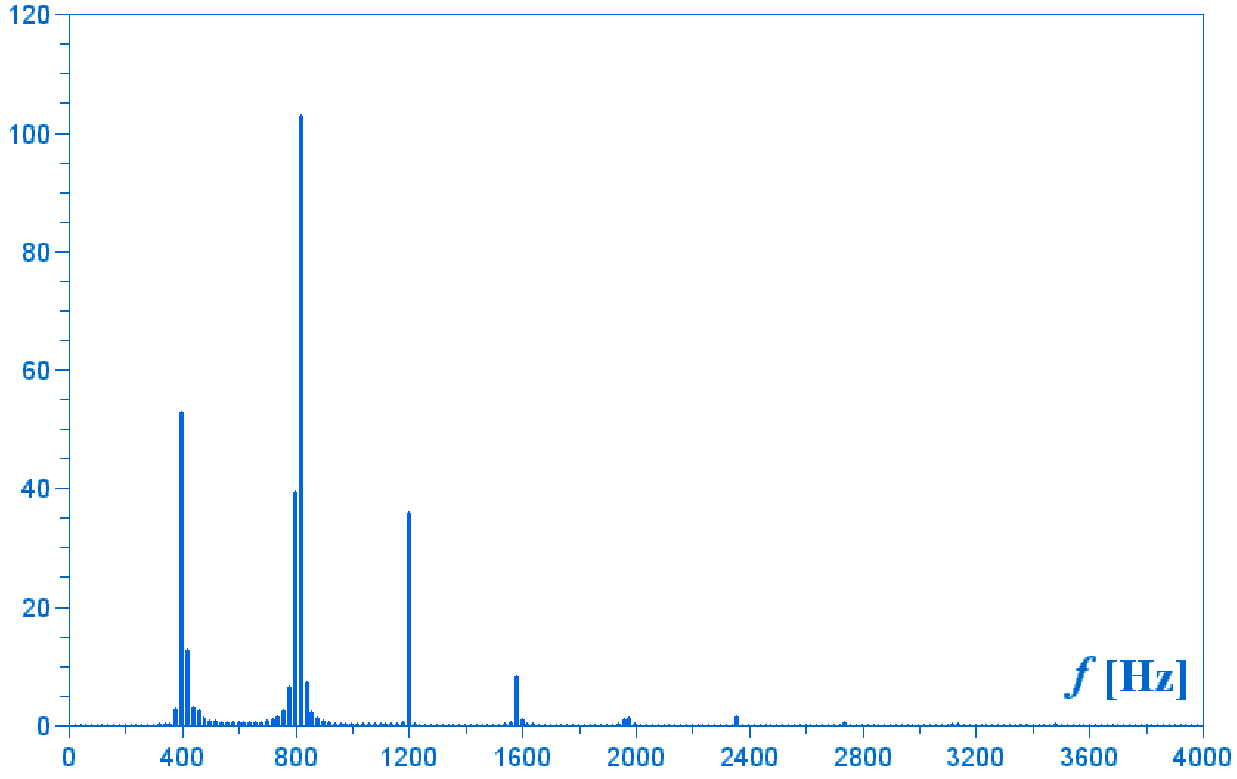

Fig. 1. Amplitude spectrum of a trumpet sound. Fundamental frequency component (f_0=400 Hz) and its multiples visible

For example, a sound with spectrum presented in Fig. 1 will have the same perceived pitch as a pure tone with frequency f_0 = 400 Hz. The harmonics with frequencies 800 Hz, 1200 Hz, 1600 Hz; in general:

$$f_k = k f_0 \,,$$

[1] A sound of indefinite pitch has non-harmonic spectrum (e.g. the sound of a bell) or noise-like characteristics

for $k = 2, 3, 4, ...$ which appear in the spectrum, as well as their relative amplitudes, do not influence the pitch. They may only modify the perception of sound timbre.

Most of natural instruments generate sound with harmonic spectrum due to the standing wave phenomenon, e. g. in a string or in a column of air. For example, a vibrating string oscillates around its balance point in *all* of the ways presented in Fig. 2 simultaneously. The frequencies of the oscillations are inversely proportional to the lengths of the string segments between adjacent nodes and those lengths form harmonic series (from bottom to top in Fig. 2). Therefore, the resulting sound is a sum of waveforms yielding the harmonic frequency spectrum.

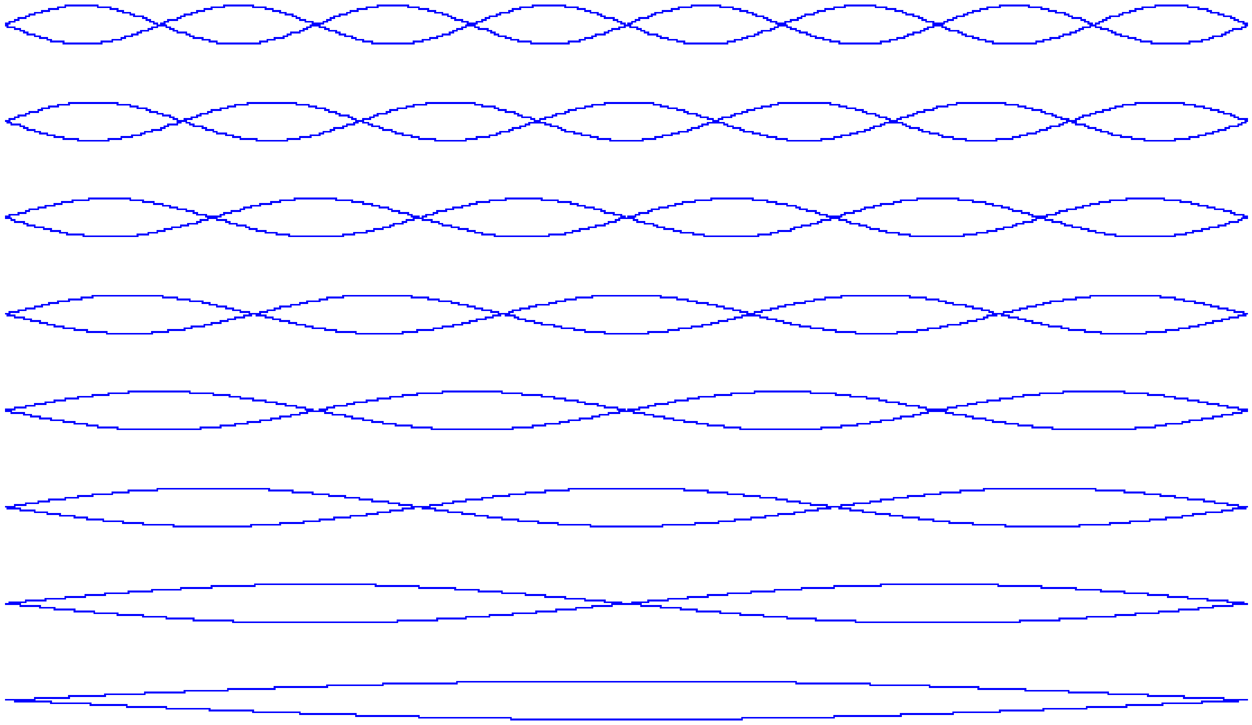

Fig. 2 Standing waves in a string

Due to the significance of the fundamental frequency for human sound perception, many methods have been proposed for computing it automatically. They are applied as a basis for conversion of a recorded sound to symbolic form (MIDI, musical notation, etc.) enabling melodic line recognition and audio material comparison irrespective of the articulation, type of instrument or acoustic conditions. They may also serve as an auxiliary technique in such tasks as speech/speaker recognition or musical instrument type recognition. The existing methods may be divided into algorithms based on time- and frequency-domain, respectively.

TIME-DOMAIN METHODS

(T1) Zero-crossing rate (ZCR)

It is one of the simplest and also the fastest methods of sound signal analysis. It is based on simply counting sign changes of the signal function in a time unit. In this way we can estimate the number of periods of the signal falling into a single analysis window. It should be noted that the method is not immune to the presence of additive noise and higher partials with significant magnitude (Fig. 3). On the other hand, it may be also used for estimating the presence of noise and some timbral features of the sound.

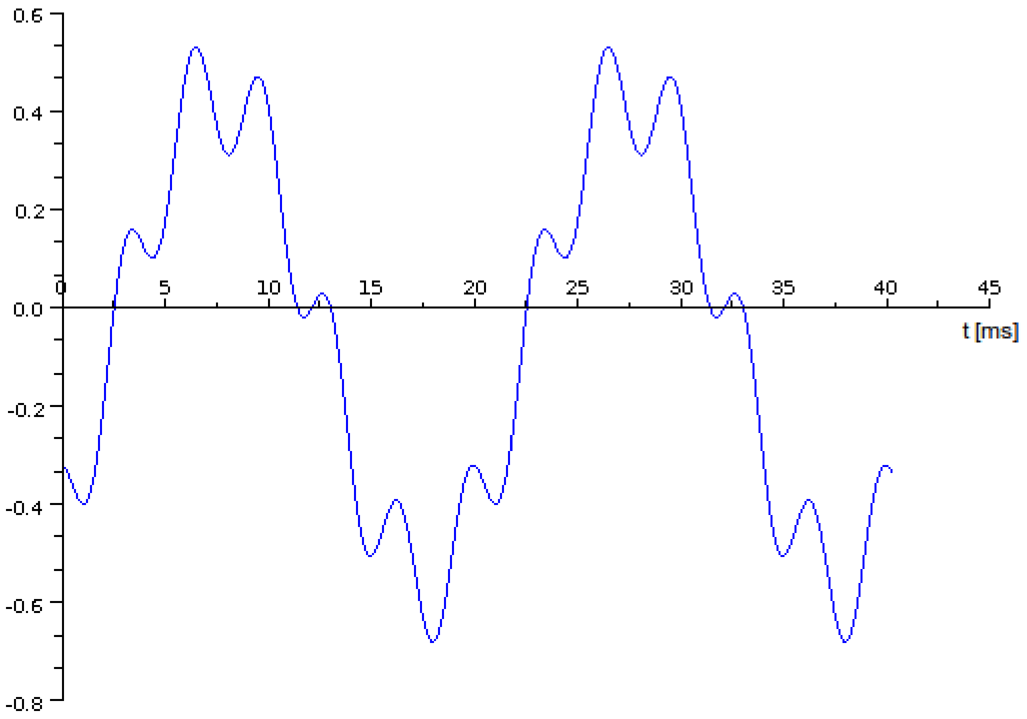

Fig. 3. Sound signal in time domain (analysis window is twice as long as the signal period) for which the ZCR method would improperly recognize a frequency two times bigger than the actual one, because the zero crossing takes place four times – instead of two – in a single period.

The robustness to noise and higher partials may be achieved to a certain extent by initial low-pass filtering (smoothing) or by some modifications of the method. For example, it is possible to count the crossings of the signal value (or its modulus) through a non-zero threshold value. The threshold value may be predefined or automatically computed on the basis of some short-term signal characteristics. It should be high enough to achieve the desired level of noise robustness but also small enough not to "lose" full signal periods. Other potential modifications include peak rate measurements or application of a similar method (zero-crossing rate or peak rate) for the analysis of the slope of the signal instead of considering its original values. In practice, it may be reasonable to apply a moving window and average the values obtained for every displacement.

In realization of this variant, both the basic version of ZCR and a version modified according to the above suggestions should be implemented and the achieved improvement should be documented.

(T2) Phase space analysis

This method is based on cycles detection in a sound signal represented in a properly defined multidimensional (phase) space. The instantaneous value of the signal is presented in the first dimension of this space and the subsequent dimensions represent its higher order derivatives.

For a discrete signal we can alternatively represent it in a pseudo-phase space in which the consecutive dimensions represent the values of the signal delayed by multiples of k samples. For example (Fig. 4), if $f(i)$ is the i-th sample of the signal, then the coordinates of the i-th point in a two-dimensional phase space would be given as:

$$(x(i), y(i)) = (f(i), f(i-k))$$

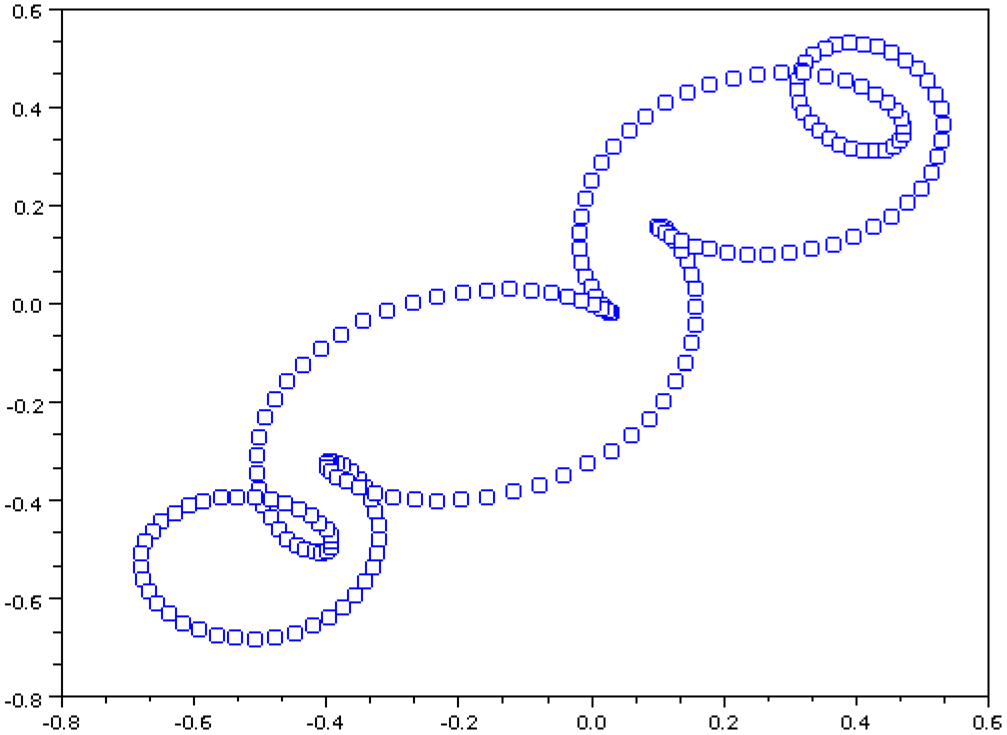

Fig. 4. An example of the signal from Fig. 3 represented in a two-dimensional phase space for *k*=10

Prime period detection occurs when we return to the starting point in the (pseudo)phase space. This fact may be ascertained by analyzing the distances between the points in the phase space. It should be noted that self-intersections occurring in the observed trajectories may result in completing the period prematurely. This problem may be solved by increasing the dimensionality of the space. For example, in order to eliminate the self-intersections visible in Fig. 4 we can add the third dimension, representing the signal shifted by the next *k* samples (i.e. 2*k* in total, w.r.t. the original signal). Fig. 4 would then represent only the projection of the three-dimensional representation and the points in the intersection areas would be in fact significantly distant from each other along the third dimension.

In realization of this variant, the proper number of space dimensions, the value of *k* and the length of the analysis window should be set experimentally. Analyzing parallel trajectories instead of computing the distances from a single point may also help in achieving improved results.

(T3) Autocorrelation

For a discrete signal $x(n)$ where $n = 0, 1, \ldots, N\text{-}1$ we compute its autocorrelation with the formula:

$$c(m) = \sum_{n=0}^{N-1} x(n)x(n+m)$$

The location of the first maximum of the resulting autocorrelation function $c(m)$ for $m\neq 0$ gives information about how many samples are contained within one period of the analyzed signal. On this basis, having the knowledge about the sampling frequency of the signal, its actual frequency may be computed.

(T4) Average magnitude difference function (AMDF)

This method is based on a comparison of the original signal with its delayed version:

$$d(m) = \sum_{n=0}^{N-1} |x(n) - x(n+m)|$$

The location of the first minimum of $d(m)$ for $m \neq 0$ yields information about the length of the signal component with the lowest frequency.

FREQUENCY-DOMAIN METHODS

(F1) Fourier spectrum analysis

Due to the characteristic structure of the spectrum of sounds with definite pitch (Fig. 1), spectral methods represent the most natural approach to the problem of fundamental frequency detection. Sound spectrum may be computed with the discrete Fourier transform (DFT):

$$X(k) = \mathrm{DFT}_N\{x(n)\} = \sum_{n=0}^{N-1} x(n) W_N^{nk} \text{ , for } k = 0, 1, ..., N-1$$

where

$$W_N^{nk} = e^{-i2\pi nk/N} = \cos(\frac{2\pi nk}{N}) - i\sin(\frac{2\pi nk}{N}) \text{ , } i = \sqrt{-1}$$

and where $x(n)$, represents the input signal for $n = 0, 1, ..., N-1$.

The output values $X(k)$ are complex numbers so we may consider their magnitudes (magnitude spectrum) and arguments (phase spectrum) separately. Local maxima of the magnitude spectrum correspond to the dominant partials of the signal. The frequency in Hertz of a partial corresponding to $X(k)$ may be computed on the basis of the spectral resolution (f_s / N) of the DFT:

$$f = (f_s / N) \times k$$

where f_s is the sampling frequency in Hertz and N is the number of samples in the analysis window.

Spectral analysis, as opposed to most of the time-domain methods, may take into consideration not only the fundamental frequency but also the harmonics. In this way it is possible to handle a situation when the fundamental frequency is not present at all or its amplitude is relatively small.

The candidates for the f_0 and harmonic frequencies are obtained by thresholding the magnitude spectrum and selecting the distinct amplitude peaks. Having obtained the sequence of dominant frequencies, our goal is to estimate the averaged distance between each pair of consecutive partials, which is equal to the fundamental frequency. In fact, taking an arithmetic mean may not yield proper results because some of the harmonics may fall below the threshold and be excluded from further analysis or, at the same time, some spurious frequency peaks may exceed the threshold and be treated as legal harmonics candidates. Some outliers-immune algorithm, e.g. based on median of the set of frequency distances, would be therefore preferred.

Another method, known as the Schroeder's histogram, is based on analysis of the frequency ratios of the candidate harmonics. For each of the peaks (with frequency f_x), selected after thresholding the spectrum, one counts how many of the remaining selected peaks has frequency equal (with a given tolerance) to some integer multiple of f_x. After computing this count for all the peaks, the one with the highest score is taken as the fundamental frequency.

Several practical issues should be taken into account in frequency spectrum analysis. The value of the threshold should be carefully set, either manually or (better) with some adaptive techniques involving e.g. the intensity of the signal in a given frame. It may be beneficial to boost the upper harmonics which may by achieved by e.g. taking the logarithm of the magnitude spectrum and/or applying the pre-emphasis filter before the DFT computation:

$$y(n) = x(n) - \alpha \times x(n-1)$$

where x and y are the input and output signals in the time domain, respectively, and the value of α should be set experimentally (usually $0.9 < \alpha < 1$). Another operation, applied typically before the DFT, is multiplying the input sequence by a window function, e.g.:

1. Gaussian window:

$$w(n) = e^{-\frac{1}{2}\left(\frac{n-(N-1)/2}{\sigma(N-1)/2}\right)^2}$$

2. Hamming window:

$$w(n) = 0.53836 - 0.46164\ \cos\left(\frac{2\pi n}{N-1}\right)$$

3. Hanning window:

$$w(n) = 0.5\ \left(1 - \cos\left(\frac{2\pi n}{N-1}\right)\right)$$

4. Bartlett window:

$$w(n) = \frac{2}{N-1} \cdot \left(\frac{N-1}{2} - \left|n - \frac{N-1}{2}\right|\right)$$

where N is the number of samples in the current analysis window. Application of the window function in the time domain usually enhances the results of the DFT, reducing the *spectral leakage* effect and making the partials easier to extract.

The size of the analysis window N should be also set properly with respect to the sampling frequency and the range of frequencies under consideration. Finally, it should be remembered that the DFT result for a real signal is symmetric, so only half of the DFT output values should be considered.

(F2) Cepstral analysis

As it may be observed in Fig. 1 the harmonic spectrum is of periodic nature due to the regular repetition of the maxima corresponding to consecutive partials. The period of those repetitions (400 Hz) is just the desired fundamental frequency of the sound. A natural approach would be therefore to compute the Fourier magnitude spectrum again and its maximum (more precisely: the location of this maximum on the frequency axis) would correspond to the reciprocal of the period under consideration.

Such representation of a signal (magnitude spectrum of a magnitude spectrum) is called a *cepstrum*. Cepstral representation of a signal may be also obtained via application of the cosine transform instead of the second Fourier transform[2]. In practice, after obtaining the magnitude spectrum of the input signal, its logarithm is computed before the second DFT is applied.

In realization of this variant, the practical aspects listed in the F1 variant description (i.a. the necessity of rejecting half of the first and also of the second spectrum due to the DFT symmetry for real signals) should be considered.

[2] There exist also a method yielding an invertible transformation due to preserving the phase information which is normally rejected in analysis based on the amplitude spectrum alone.

(F3) Comb-filter method

Harmonic magnitude spectrum may be modeled in a simplified way with a comb function, defined as a sum of equidistant window functions in the *frequency* domain (cf. Fig. 1). The shape of the window function may by arbitrary, e.g. gaussian or triangular (in a simplest case the window may be even reduced to a single impulse). The pitch of the sound corresponds directly to the distance between consecutive maxima of the comb function (the middle points of the consecutive windows).

In order to find the pitch of the analyzed sound, a set of comb functions must be applied, where each comb function has different distance between the consecutive maxima. Each comb function is represented in a discrete form, as a sequence of length equal to the number of elements of the analyzed spectrum and the inner product of the two is computed. In other words, we multiply – element by element – the comb function and the sound magnitude spectrum, summing the obtained products. The comb function for which the greatest sum is obtained is hence the best model of the input signal spectrum (we have "hit" the harmonics of the sound with the maxima of the comb function), so the distance between its maxima corresponds to the fundamental frequency we have been searching for.

THE TASK

An application enabling to read a *.wav file and to detect the fundamental frequency in consecutive time windows should be designed and implemented. The detected frequency should be presented in a numerical form (in Hz) and via generation of a pure tone with the same frequency. In case of a recordings containing a fragment of melodic line composed of several consecutive sounds with different pitches, the result should be a list of frequencies, accompanied by a list of corresponding durations and it should be also presented in the audio form (as a sequence of pure tones with proper frequencies and durations). The provided sound examples should be applied in the testing procedure. It is necessary to implement one of the methods in the time domain and one of the methods in the frequency domain, according to variants assigned by the teacher.

(T1) Zero-crossing rate (ZCR)
(T2) Phase space analysis
(T3) Autocorrelation
(T4) Average magnitude difference function (AMDF)

(F1) Fourier spectrum analysis
(F2) Cepstral analysis
(F3) Comb-filter method

SOURCES

- Gerhard D (2003) Pitch extraction and fundamental frequency: History and current techniques. Tech. rep., Dept. of Computer Science, University of Regina

- Dziubiński M, Kostek B (2004) High accuracy and octave error immune pitch detection algorithms. Archives of Acoustics 29(1):1–21

- Boersma P (1993) Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound. Institute of Phonetic Sciences, University of Amsterdam, Proceedings 17:97–110