Audio Engineering Society
Convention Paper
Presented at the 116th Convention
2004 May 8–11 Berlin, Germany

Octave-Error Proof Timbre-Independent Estimation of Fundamental Frequency of Musical Sounds

Alicja Wieczorkowska[1] and Jakub Wróblewski[1]

[1] *Polish-Japanese Institute of Information Technology, 02-008 Warsaw, Poland*

Correspondence should be addressed to Alicja Wieczorkowska (alicja@pjwstk.edu.pl)

ABSTRACT

Estimation of fundamental frequency (so called pitch tracking) can be performed using various methods. However, all these algorithms are susceptible to errors, especially octave ones. In order to avoid these errors, pitch-trackers are usually adjusted to particular musical instruments. Therefore problem arises when one wants to extract fundamental frequency independent on the timbre. Our goal was to elaborate method of fundamental frequency extraction, which works correctly for any timbre. We propose multi-algorithm approach, where fundamental frequency estimation is based on results coming both from a range of frequency tracking methods, and additional parameters of sound. Also, we propose frequency tracking based on direct analysis of signal and its spectrum. We explain the structure of our estimator and the obtained results for various musical instruments and sound articulation techniques.

1. INTRODUCTION

Processing of music data often requires estimation of fundamental frequency of recorded sounds. Most popular application is extraction of pitch of consequent notes of the recorded melody, i.e. pitch tracking. Also, sound analysis for the purpose of the timbre description may require estimation of fundamental frequency, since the timbre of sounds of definite pitch is usually described in terms of the contents of harmonic partials in spectrum. Therefore, pitch extraction is important to many users of audio data.

The issue of pitch estimation is not new and it has been investigated in many papers. Most of pitch extraction methods come from speech analysis. There exist numerous methods to perform this task, based on various analyzes of audio data. The most popular approaches include such methods as autocorrelation, methods based on zero-crossings of sound wave, maximum likelihood, Average Magnitude Difference Function (AMDF), spectral and cepstral analysis, and so on, see [1, 3, 5, 6, 7, 8]. However, these methods may not work correctly unless their implementations are adjusted to the specific features of sounds, i.e. to the timbre. When applying the frequency estimation algorithm to sounds of various timbre, for instance various musical instruments, correctness decreases. Apart from inexactness of estimation, octave errors may appear and this kind of error is quite common here.

In our work, we decided to apply multi-algorithm approach, where a range of classic fundamental frequency estimation methods is used. We also propose pitch estimation based on direct analysis of signal course. Additionally, we label each sample with sound descriptors to facilitate final choice of correct value of fundamental frequency. We investigated sounds from the McGill University Master Samples (MUMS) CD collection, containing isolated monophonic sounds of various musical instruments [13]. In our research, we used samples of brass, woodwind, and string instruments of contemporary orchestra, played with various articulation techniques. We tested wide variety of timbres, since our data set included strings played vibrato and pizzicato, woodwinds, also with vibration, and brass, played without and with muting. Frequency range in our research was very wide, since we experimented with 5 full octaves, from C2 to C7 in MIDI notation, i.e. from 65.41 Hz to 2093 Hz (nominal frequencies).

The approach that we propose consists in direct using sound course and a few basic sound descriptors for fundamental frequency estimation, together with classical pitch extraction methods, and to build the approximation model. Following this idea, we apply artificial intelligence (AI) methods, to be able to use the values of sound descriptors as the constraints leading to more effective local estimators, as well as to select the best approximation from the list of candidates calculated using particular pitch tracking methods, and finally for synthesizing the results obtained from the classical techniques. Using this approach, one can make use of information about sound and its timbre to calculate more correctly the fundamental frequency. As a result, correct pitch recognition can be performed for sounds of various timbre features, i.e. coming from various musical instruments and articulation techniques, because sound characteristics can guide correct extraction of fundamental frequency for any sounds.

2. PITCH ESTIMATION

There exist numerous algorithms for fundamental frequency estimation, based on various methods of sound analysis. Most of pitch extraction methods originate from speech processing. Methods based on spectral analysis directly derive fundamental frequency of the sound, for instance Harmonic Product Spectrum (HPS), cepstrum analysis, Cepstrum-Biased HPS (CBHPS), constant-Q spectral transform, and Maximum Likelihood (ML) [2], [4], [7], [9]. Time-domain based methods are used to calculate the period T_0 of sound, and fundamental frequency f_0 is then calculated as

$$f_0 = \frac{1}{T_0} \tag{1}$$

These methods are based on observation of zero crossings, autocorrelation, also in "narrowed" and weighted version [3], [7], Average Magnitude Difference Function (AMDF) and maximum likelihood [4], [5], [10], [12]. All these techniques are usually applied in case of monophonic pitch tracking; for polyphonic sounds, further methods are introduced, see for instance [11], [16], [17]. However, as we already mentioned, octave errors are common problem

in pitch tracking for musical instrument sounds, because of harmonicity of musical signal. To cope with this problem, pitch trackers are sometimes adjusted to characteristics of the analyzed sounds, if the algorithm is to be applied to the specific instrument, or to the frequency range. In such case, various parameters of pitch tracker, like thresholds necessary in the implementation etc., are set to work with this instrument, but not with any other. Our goal is to elaborate an algorithm that works with any instrument of definite pitch, applying several algorithms for estimation of fundamental frequency, and AI methods guided by basic sound parameters.

Our frequency estimator considers several methods for the fundamental frequency calculation. It is based on: AMDF, Fourier analysis, autocorrelation, and observation of signal values, including extreme values and zero-crossings. For each of them, the derived results are added to the list of candidate fundamental frequencies.

2.1. AMDF

AMDF applied in our research has the following form [5], [7]:

$$AMDF(i) = \frac{1}{N}\sum_{k=0}^{N-1} |x(k) - x(i+k)|^j \qquad (2)$$

where N is the last sample in the interval taken for estimation and $x(k)$ is the value of the signal sample. For simplicity reasons, we used Equation 2 with $j = 1$. The period for the analyzed sound is determined as i corresponding to the minimal value of $AMDF(i)$. We evaluate the approximate period length within the stable part of the sound, and 100 ms frame is taken for analysis. The beginning of the useful part of the signal is set as the moment when the sound envelope reaches 10% of maximal amplitude. The end of the analyzed part of the sound is found analogously from the ending side of the sound. We select the most stable part of the sample, where peaks of signal are closest to the level determined by maximum of signal amplitude in this 100 ms frame. Selection of analyzing frame is illustrated in Figure 1.

Unfortunately, AMDF analysis of signals is susceptible to octave errors, since global minimum is at the beginning of analyzing frame, and local minima can be misleading. A few examples of AMDF function for selected string sounds are shown in Figure 2.

2.2. Autocorrelation

Autocorrelation function for a signal frame of length N is defined as

$$R(i) = \frac{r(i)}{r(0)}, \quad i = 0, \pm 1, \pm 2, \ldots \qquad (3)$$

where

$$r(i) = \frac{1}{N}\sum_{k=0}^{N-i-1} x(k)x(i-k), \quad i = 0, 1, 2, \ldots \quad (4)$$

Sound period is found as the first local maximum of the autocorrelation function. In our calculations we omit division by $r(0)$, so we use autocorrelation function in form of Equation 4.

2.3. Spectral Analysis

The outcomes of spectral analysis in our multi-algorithm pitch extractor is rather use as a guiding parameter for the algorithm of pitch estimation. Spectrum is calculated also for 100 ms frame with Hanning window $w(m)$, i.e. in the following form:

$$X(i) = \sum_{k=0}^{N-1} x(kT)w(kT)\exp -ji\Omega kT \qquad (5)$$

$$i = 0, \ldots, N-1, \quad \Omega = 2\Pi/NT$$

where

$$w(m) = \frac{1}{2}(1 + \cos\frac{2\Pi m}{N}) \qquad (6)$$

and T - sampling period, equal 1/44100 s in our case.

We extract frequency of maximal amplitude f_m in the spectrum, and next we test frequencies f_m/i, $i = 1, ...12$ as candidates for fundamental.

2.4. Max-to-Min Signal Analysis

Additionally, we propose a method based on observation of signal values, intended to be used in case of very low frequencies. We start with signal downsampling by finding difference between maximal and minimal values in 1 ms range. These differences form a new, simplified signal, which is next analyzed by

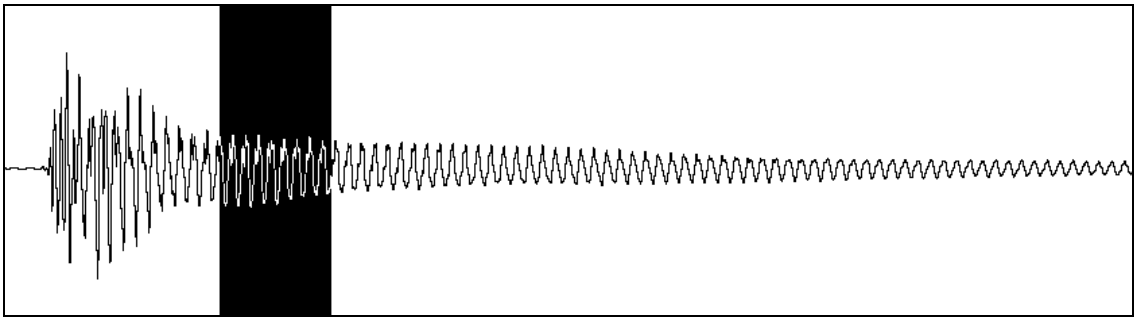

Fig. 1: Selection of a frame for pitch estimation

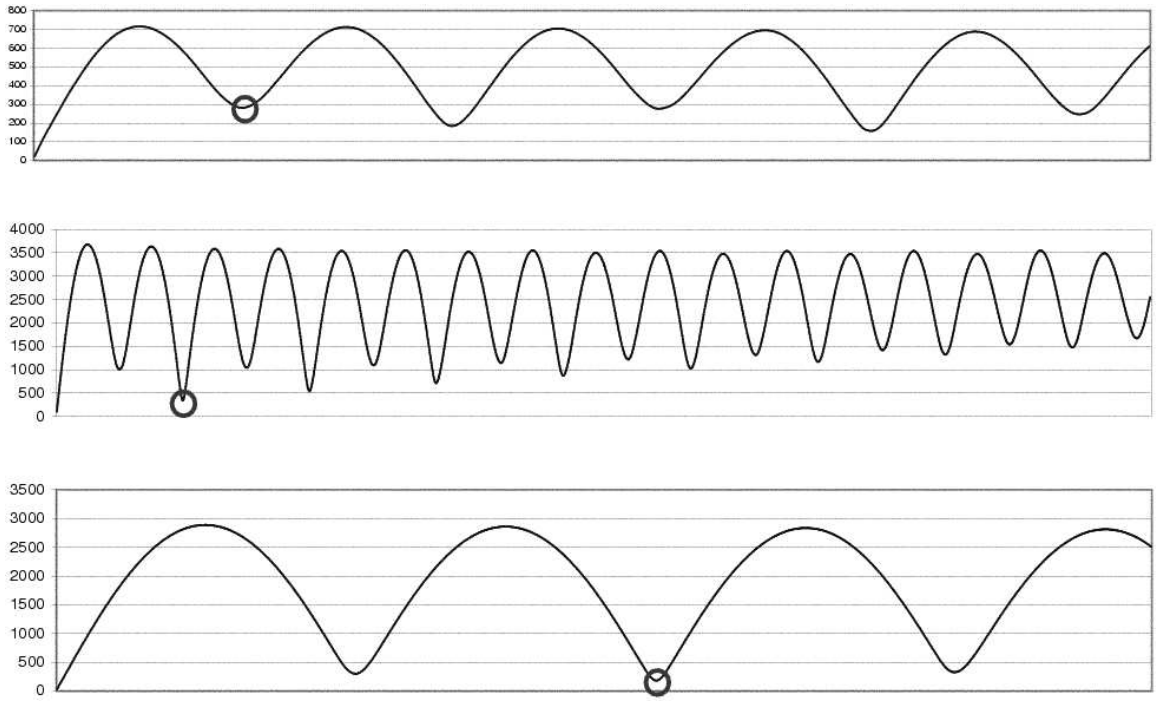

Fig. 2: Examples of AMDF results for double bass vibrato sound F1 of nominal frequency 43.65 Hz (upper figure), viola pizzicato sound F3 of nominal frequency 174.6 Hz (middle figure), and double bass pizzicato sound B1 of nominal frequency 61.73 Hz; MIDI notation applied to notes. Circles show correct values of sound period

zero-crossings. If the new signal is strong enough, i.e. its maximal value is at least twice as high as its minimal value, then its frequency obtained by zero-crossing analysis is regarded as a next candidate of a pitch. In zero-crossing analyzing frame, the number of points where signal values change from negative to positive is counted. This method is illustrated in Figure 3.

3. KDD APPROACH TO SOUND ANALYSIS

Pitch extraction and sound analysis became the sub-

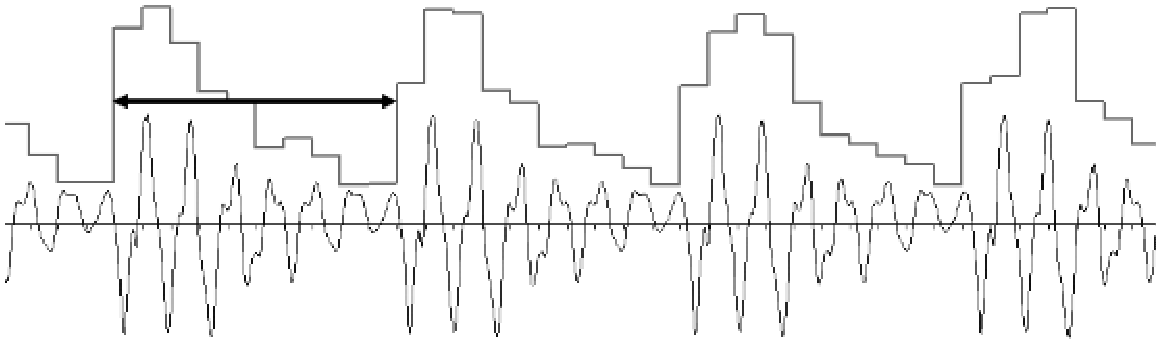

Fig. 3: Period extraction using max-to-min span algorithm

ject of interest for us because of using spectral descriptors for musical instrument sound recognition purposes. Our research focused on classification of musical instrument on the basis of its sound, independently of the pitch. Some of the timbre descriptors required calculation of fundamental frequency in a preprocessing phase. Therefore, correct estimation of pitch was crucial for correct calculation of sound features, and we realized that methods of pitch extraction need further improvement. Since our work was based on KDD methods, we decided to used this approach too for pitch estimation. Outcomes of the research on musical instrument sound classification inspired us to use sound feature to guide and aid pitch extraction. The details of this research are given in [14, 15, 18, 19]. We focused there on methodology of musical instrument sound recognition, related to KDD process of the training data analysis. Our classification was based on a set of features, calculated for particular sound samples. The feature values for the samples constituted objects in a relational database of the sound sample representations. We used features similar to descriptors from MPEG-7, but we also considered the clustering and time series framework, by taking as new descriptors temporal patterns observed for particular features. General structure and contents of the obtained database is presented on Figure 4; further details are given in [18].

Especially two of spectral-based sound descriptors were important for our current research – the brightness of the sound:

$$Br = \frac{\sum_{n=1}^{M} n\, A_n}{\sum_{n=1}^{M} A_n} \tag{7}$$

and the even harmonics content in spectrum:

$$Ev = \frac{\sqrt{\sum_{k=1}^{P} A_{2k}^2}}{\sqrt{\sum_{n=1}^{M} A_n^2}} \tag{8}$$

where A_i - amplitude of i-th partial, M - number of available partials in the spectrum, and P - number of available even partials in the spectrum.

All spectral parameters were calculated on the basis of fundamental frequency, so we had to determine such frequency at the beginning of preprocessing stage. A spectral-based algorithm (described in the next sections as C_1 candidate for fundamental frequency) was used to achieve it, but this method had a limited accuracy. Thus, a new method for more accurate fundamental frequency estimation was needed. Since the database was used to automatic recognition of musical instruments, the new fundamental frequency estimator could not rely on information about a kind of instrument (but this information can be inferred from the other sound features, as temporal or spectral ones).

4. MULTI-ALGORITHM APPROACH

Our pitch detection algorithm consists of several steps, leading to extraction of four f_0 candidates,

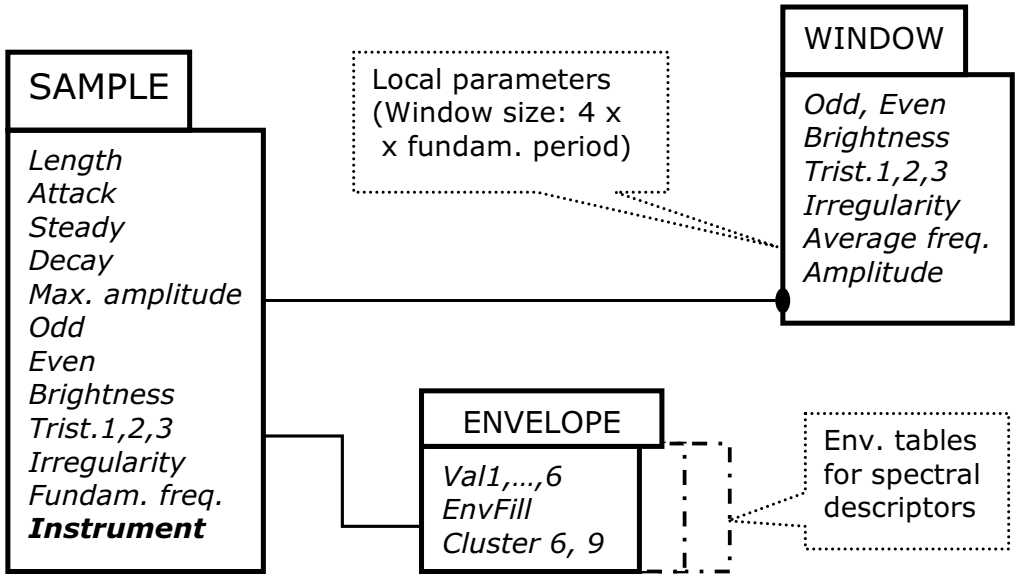

Fig. 4: Relational database used in our experiments on musical instrument sound recognition [18]

denoted C_i, $i = 1, ..., 4$. The main candidate is C_1 is extracted by the following procedure. First of all, the maximal amplitude of the spectrum is calculated and corresponding minimum of AMDF function is found. This value is regarded as the candidate C_2 for fundamental period, then it is doubled to obtain candidate C_3. Then, up to 12 multitudes of the period are found and local minima of AMDF function are calculated. The best candidate C_1 for fundamental frequency should have significantly lower value of AMDF function comparing to the value for the first candidate (at least 50% decrease), or at least both AMDF function and corresponding spectrum amplitude should be better (20% change – increase in case of spectrum, or decrease in case of AMDF). The fourth candidate C_4 is calculated via min-to-max procedure, described in Section 2.4.

All the data coming from pitch estimation procedures, together with the actual fundamental frequency for each sample, form the training set, enabling our algorithm to learn the ultimate fundamental frequency estimator. Additional criteria come from sound features calculated during preprocessing for sound description purposes [18]. Since we work on isolated sounds, we also included parameters of sound envelope.

Altogether, the following features are calculated as starting point of our algorithm:

- candidate frequencies C_1, C_2, C_3, C_4 extracted by the procedures described above,
- autocorrelation and density of zero-crossings, i.e. number of points where signal values changes from negative to positive in the analyzed 100 ms frame,
- length of the attack, quasi-steady state and decay, normalized through dividing by the length of the whole sound,
- signal length,
- type of signal envelope – one of 9 classes found by an unsupervised learning algorithm based on clustering (see [18]); the envelopes are presented in Figure 5,
- spectral parameters: sound brightness and level of even harmonics (see Section 3), calculated for the candidate C_1,
- difference between candidate C_1 and C_2, also octave number for candidate C_2.

These features may be used as constraints (attributes in a decision table) in choosing the correct value of pitch.

5. EXPERIMENTS AND RESULTS

The data we analyzed comprise 667 objects, representing isolated monophonic sounds. 11 musical instruments were chosen, and sounds were recorded with various articulation. Our samples represent flute, oboe, clarinet, violin, viola, cello, double bass, trumpet, trombone, French horn, and tuba; articulation techniques include vibrato, pizzicato, and muting (in case of brass). Frequency range covers full 5 octaves, from C2 of nominal frequency 65.41 Hz to C7 of nominal frequency 2093 Hz (notes in MIDI notation). Sounds were digitally stereo recorded with 44.1 kHz sampling frequency and 16 bit resolution.

We started our experiments with classic time-domain methods, i.e. AMDF and autocorrelation, and also with analysis of spectral peaks. Exemplary results for AMDF and autocorrelation, compared with spectral peak, are presented in Figure 6. In this case, results from all methods coincide, yielding correct period value, equal about 525 samples for sampling frequency 44.1 kHz.

For a total number of 667 sound samples, only 35% were recognized correctly using zero-crossing analysis and only 47% were recognized properly using autocorrelation. On the other hand, recognition ratio for simple spectrum-AMDF method (described above as C_2), where spectrum amplitude maximum was used as a guide for AMDF algorithm, gave the recognition ratio 59%.

The best result was obtained by more complex algorithm, involving analysis of 12 candidate frequencies (denoted above as C_1). As much as 96.25% samples was recognized properly. On the other hand, many of these errors may be corrected by using the max-to-min span technique described in Section 2.4. Finally, we used four pitch estimation algorithms: simple spectrum-AMDF method (C_2), the same method with the result lowered by one octave (C_3), max-to-min span (C_4), and finally the complex method (optimal value C_1 among 12 candidates). If we were able to choose the proper method among these 4, classification ratio would rise to 99.55%.

Final decision on the choice of the pitch in the described research is made using AI methods. The whole algorithm of the pitch estimation is presented in Figure 7.

Our experiments involve use of AI methods (data mining techniques) to determine the best among these four pitch detection techniques. Our decision table contains 9 conditional attributes (mainly time-related) as described in the previous sections. The number of the proper pitch detection algorithm is used as a decision. We have used several data mining tools, including decision trees, neural networks, Bayesian classifiers and rule induction algorithms. Unfortunately we were able to increase results only slightly – the best method was k-NN classifier (k nearest neighbors) which yielded up to 96.70% of correct answers (only 3 more samples were classified correctly comparing with the best non-AI method C_1).

The main reason of relatively poor efficiency of the approach described above is concerned mainly with the fact, that in our case different decision values may be good at the same time. Classical data mining tools are not ready for such additional feature of decision problems. Thus, a hierarchical system was implemented and tested on our data set. The final algorithm is constructed as follows:

- Calculate all parameters described in Section 4.
- Create a subset of database containing only sound samples for which max-to-min signal analysis (see Section 2.4) is calculated (i.e. the signal is strong enough). In our data set, the table contains 542 samples out of 667 total.
- Solve the first decision problem: whether one should use span analysis (C_4 candidate) or spectral one (C_1 candidate). We have tested several data mining tools, and k-nearest neighbors method turned out to be the most successful. As much as 531 samples (97.97%) were classified correctly.
- Analyze the rest of data set (in our case – 125 samples). Simple rule induction method yields surprising result – one decision rule can eliminate half of erroneous objects: if brightness parameter is more than 5, then use candidate C_2, else use candidate C_1.

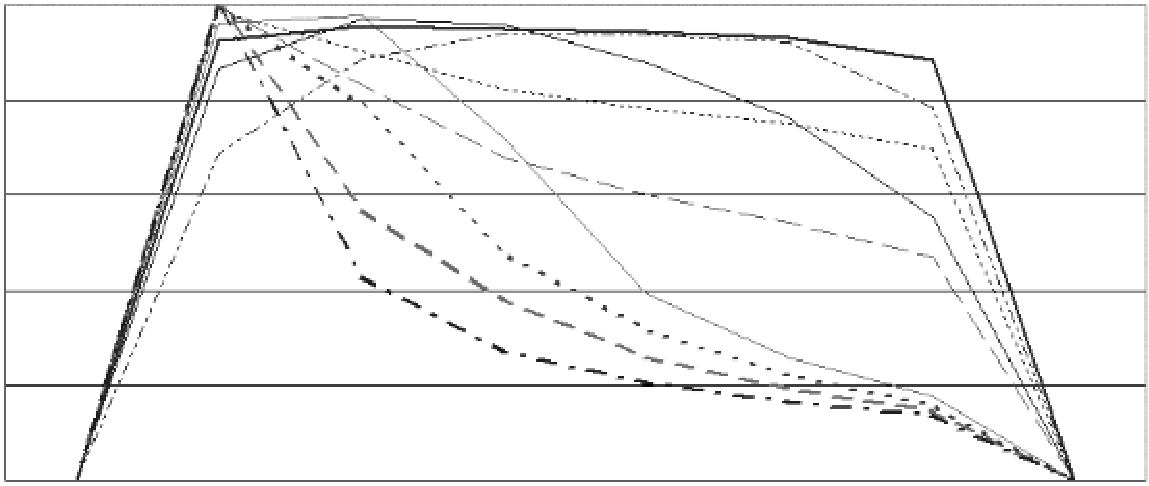

Fig. 5: Types of sound envelopes found via clustering algorithm

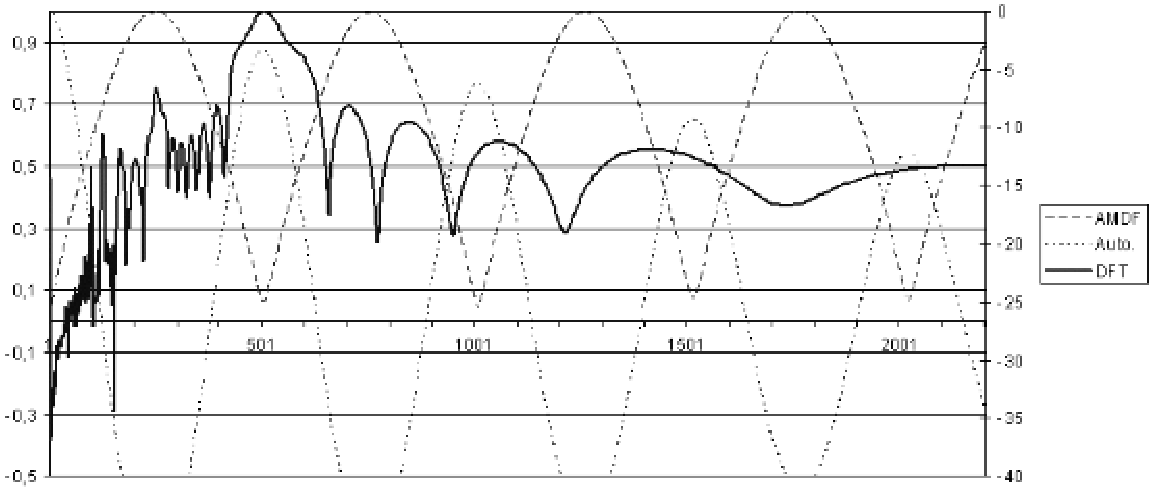

Fig. 6: Exemplary result for classical methods: AMDF, autocorrelation, and maximal spectral peak for cello pizzicato sound F2 (MIDI notation) of nominal frequency 87.31 Hz. The peak values for all three diagrams are around 525 samples (for 44.1 kHz sampling frequency), so the results are correct. Diagram for Fourier analysis (DFT) is plotted in dB scale

The hierarchical algorithm presented above was able to reduce as much as a half of pitch detection errors, comparing to the best simple method (candidate C_1). The final recognition ratio is 98.05% of correct answers (13 errors).

Results of our experiments are presented in the table 1.

A few sounds, belonging to various families of instruments, were especially difficult for pitch estimation. These sounds represented muted trumpet (A4, 440 Hz and D5, 587.3 Hz), and viola vibrato (D♯5, 622.3 Hz). A fragment of time domain of one of them is presented in Figure 8. As we can see, two local maxima of almost the same value can be mislead-

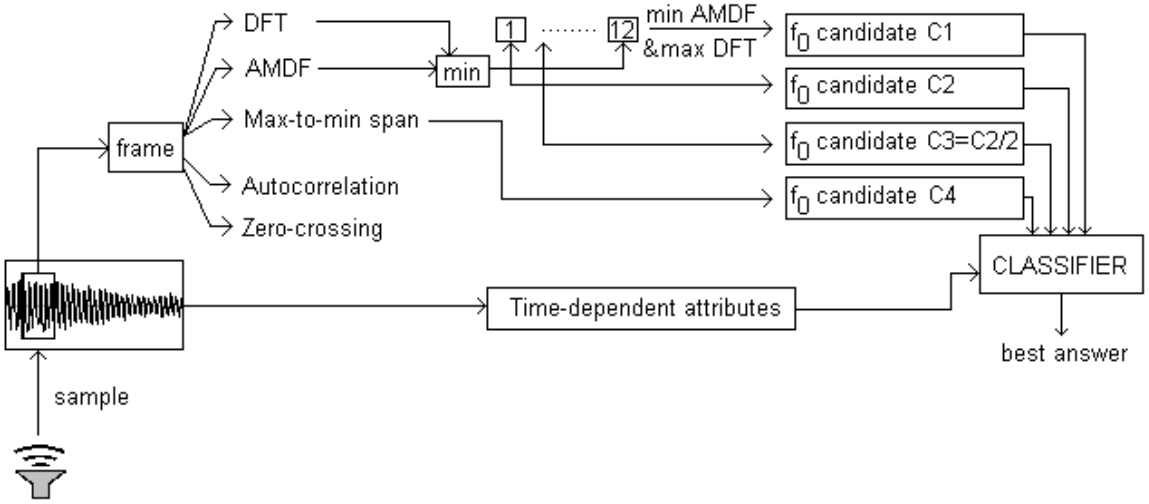

Fig. 7: Algorithm of fundamental frequency extraction

Method	Correct answers
Simple pitch detectors:	
Spectral (C_2)	58.92%
Spectral optimized (C_1)	96.25%
Autocorrelation	47.68%
Zero-crossing	35.23%
Max-to-min span (C_4)	38.08%
The best choice of above (upper boundary)	99.55%
Data mining algorithms on four candidates:	
k-NN	96.70%
Decision tree	96.40%
Neural network	95.95%
Hierarchical approach	**98.05%**

Table 1: Comparison of results of different pitch detection algorithms

ing for the pitch extraction algorithm, which must be tolerant to small differences between maxima in consequent periods of the signal. Also vibration of sound, as in case of viola vibrato D♯5 sound, makes correct pitch extraction difficult because of signal fluctuations. Altogether, these sounds lowered accuracy of our algorithm, but the final correctness is quite high.

6. SUMMARY AND CONCLUSION

We have proposed a multi-algorithm approach to pitch detection, and also a method based on signal value observation that traces extrema and zero-crossings, which is especially useful for lower frequencies. We have tested our algorithm on a library of 667 samples, played by 18 different instruments or articulation styles and covering full 5 octaves. Our method is relatively efficient as it achieves more than 96% of correct answers.

The next step is to determine automatically the best method of pitch detection among four proposed. Using a hybrid (hierarchical) solution, employing data mining techniques, we were able to significantly decrease the number of errors by 12 samples (out of 25 incorrect ones) to the value of 98.05%. However, it is still far from upper boundary of these four methods working together in optimal way (99.55%), and from 100% correct estimation. The reason of such

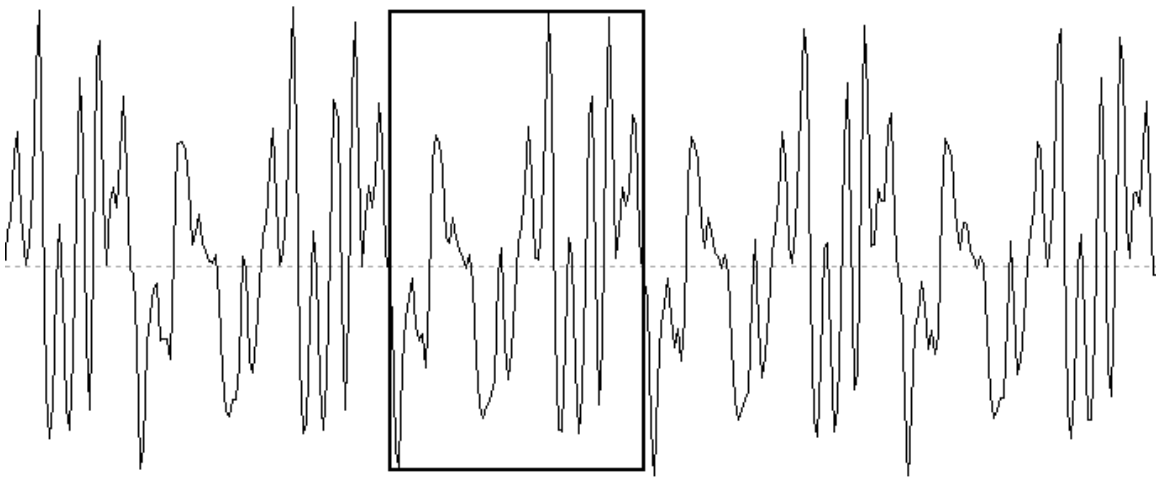

Fig. 8: Sound of muted trumpet, D5 (in MIDI notation), of nominal frequency 587.3 Hz. Black frame shows the sound period

a result may be concerned with the special structure of the decision: all typical data mining methods are focused on strict discernibility of decision values, whereas in our case different decision values may be good at the same time, since different methods may yield the same pitch. Our future works will be focused on employing the special structure of decision values in data mining algorithms.

7. ACKNOWLEDGEMENT

This research was sponsored by the Research Center of PJIIT, supported by the Polish National Committee for Scientific Research (KBN).

8. REFERENCES

[1] J. W. Beauchamp, R. Maher, R. Brown, *Detection of Musical Pitch from Recorded Solo Performances*, 94th AES Convention, preprint 3541, Berlin, 1993.

[2] J. C. Brown, *Calculation of a Constant Q Spectral Transform*, J. Acoust. Soc. Am. 89, 425-434, 1991.

[3] J. C. Brown, B. Zhang, *Musical Frequency Tracking using the Methods of Conventional and "Narrowed" Autocorrelation*, J. Acoust. Soc. Am., 89, 2346–2354, 1991.

[4] G. C. Burnett, *The Physiological Basis of Glottal Electromagnetic Micropower Sensors (GEMS) and Their Use in Defining an Excitation Function for the Human Vocal Tract*, PhD dissertation, University of California, Davis, 1999.

[5] P. R. Cook, D. Morrill, J. O. Smith, *An Automatic Pitch Detection and MIDI Control System for Brass Instruments*, invited for special session on Automatic Pitch Detection, Acoustical Society of America, New Orleans, 1992.

[6] D. Cooper, F. C. Ng, *A monophonic pitch tracking algorithm*, Report 94.15 (May 94), University of Leeds School of Computer Studies, 1994.

[7] P. de la Cuadra, A. Master, C. Sapp, *Efficient Pitch Detection Techniques for Interactive Music*, Proceedings of ICMC 2001, International Computer Music Conference, La Habana, Cuba, September 2001.

[8] B. Doval, X. Rodet, *Estimation of Fundamental Frequency of Musical Sound Signals*, IEEE, A2.11, 3657–3660, 1991.

[9] D. Gerhard, *Pitch Extraction and Fundamental Frequency: History and Current Techniques*, Technical Report TR-CS 2003-06, Department of Computer Science, University of

Regina Regina, Saskatchewan, Canada, November 2003.

[10] T. Jehan, *Musical Signal Parameter Estimation*, MS Thesis in Electrical Engineering and Computer Sciences from IFSIC, University of Rennes 1, France. Center for New Music and Audio Technologies, Berkeley, 1997.

[11] R. C. Maher, *Evaluation of a Method for Separating Digitized Duet Signals*, J. Audio Eng. Soc., Vol. 38, No. 12, 956-979, December 1990.

[12] K. Marasek, *EGG and voice quality*, tutorial, available at http://www.ims.uni-stuttgart.de/phonetik/EGG/, 1997.

[13] F. Opolko, J. Wapnick, *MUMS – McGill University Master Samples*, CD's, 1987.

[14] D. Slezak, P. Synak, A. Wieczorkowska, J. Wroblewski. *KDD-based approach to musical instrument sound recognition*, M.-S. Hacid, Z.W. Ras, D.A. Zighed, Y. Kodratoff (eds.): Foundations of Intelligent Systems. Proc. of 13th Symposium ISMIS 2002, Lyon, France. Springer-Verlag (LNAI 2366), Berlin, Heidelberg 2002, pp. 28-36.

[15] P. Synak, *Temporal Aspects of Data Analysis: A Rough Set Approach*, Ph.D. Thesis, The Institute of Computer Science of the Polish Academy of Sciences, Warsaw, Poland, 2003.

[16] E. Terhardt, G. Stoll, M. Seewann, *Algorithm for extraction of pitch and pitch salience from complex tonal signals*, J. Acoust. Soc. Am., Vol. 71, No. 3, 679-688, 1982.

[17] P. J. Walmsley, S. J. Godsill, P. J. W. Rayner, *Bayesian Graphical Models for Polyphonic Pitch Tracking*, Diderot Forum, Vienna, December 1999.

[18] A. Wieczorkowska, J. Wroblewski, D. Slezak, P. Synak. *Application of temporal descriptors to musical instrument sound recognition*, Journal of Intelligent Information Systems 21(1), Kluwer 2003, pp. 71-93.

[19] A. Wieczorkowska, J. Wroblewski, D. Slezak, P. Synak. *Problems with Automatic Classification of Musical Sounds*, Proc. of the Intelligent Information Processing and Web Mining Conference IIS: IIPWM'2003, Zakopane, Poland. Springer 2003, pp. 423-430.