Learning Algorithms for the Cancellation of Old Recordings Noise

3847 (P11.7)

Andrzej Czyzewski, Cezary Suproń
Technical University of Gdansk
Electronic Faculty, Sound Engineering Dept.
Narutowicza 11/12
80-952 Gdansk, Poland

Presented at the 96th Convention 1994 February 26 - March 01 Amsterdam

AES

AN AUDIO ENGINEERING SOCIETY PREPRINT

Learning Algorithms for the Cancellation of Old Recordings Noise [1/]

Andrzej Czyżewski, Cezary Suproń

Technical Univ. of Gdańsk, Electronic Faculty, Sound Engineering Dept.
Narutowicza 11/12
80-952 Gdańsk, Poland

The elaborated software package for digital restoration of sound provides a system of digital operations applied to the registered signal of old records or magnetic tapes in order to bring the sound image back to its initial form. Particular attention was paid to the detection and elimination of scratches, hiss and similar types of impulse noise. A learning algorithm based on neural network simulation was applied to this task. Another learning algorithm implementing the Wiener adaptive filtering procedures was used to continuous noise suppression. Applied methods and obtained experimental results are presented.

0. Introduction

As results from experiments performed by the authors, elimination of impulse noise demands the use of learning algorithms, because individual cracks, clicks and hiss cannot be discerned easily from desired impulsive sounds, such as percussive ones. Consequently, many algorithms published previously [1][2] proven to be particularly effective when searching for selected cases of clicks and cracks affecting especially chosen music patterns.

Hence, the collection of methods based on the mathematical approach to the problem should be completed with learning algorithm implementations. This paper deals with the concept and with the results of an application of a neural net algorithm to the recognition of clicks in musical sound.

For the task of cancelling the additive (continuous) noise an improved Wiener filtering based algorithm was developed. The adaptive *lattice* type filter bank was used to the spectrum decomposition in order to extract from the spectrum the subbands corresponding to critical bands of hearing sense. Subsequently, the reduction of background noise using adaptive Wiener filtering in the critical bands was investigated. Correspondingly, the software was constructed in such a way that all frequency domain filtering procedures may be applied both to the full band of the signal or to each critical band individually. That gives the system the required flexibility, allowing to perform digital filtering in the selected subbands important as to the subjective perceiving of the noise. Moreover, different procedures may be used in relation to various parts of the signal spectrum improving efficiency of the elaborated sound restoration system.

2. Acquisition of the training material

The patterns used in the training of a learning algorithm should be extracted from real music or speech recordings. Synthetic impulses are not applicable to this task, because when preparing the training material it is to be ensured, that the cracks or clicks are to be cut-off together with the neighboring samples of the desired signal. Only the patterns prepared in such a way may guarantee subsequent right classification of impulses by the learning algorithm. As results from authors' experience, it is also necessary to add some patterns related to the desired musical sounds having impulsive character. Such patterns enable to teach the learning algorithm how to avoid misleading classifications in the case of spotting the transients or another musical signal events having the impulse-like character. In such a case the algorithm should be trained to not react to such impulses. Moreover, the extraction of impulse noise patterns from musical signal cannot be done only manually through the use of the computer sound editing procedures. It results from the fact that not all audible noises are to be seen in computer analyses of audio signal, remaining masked within the visual form of the time-domain characteristics. The spectral analysis of the signal reveals loud impulses presence due to the appearance of high frequency components (Fig. 1a) not seen in the neighboring fragments of the same signal (Fig. 1b). However, there is no possibility to discern the transient effect from the effect of undesired impulse presence using the simple spectrum analysis methods.

Spectral or sona-graph plots and particularly LPC sona-graph analyses proven to be far more efficient tools for spotting the impulses [3], however there was still a need to employ an automatic detection method to the task of extraction of some kind of clicks and hiss. Consequently, despite the use of the sound editor especially elaborated for this purpose, authors applied also the wavelet analysis-based procedure to automatical detection of clicks in audio material.

The wavelet transform is based on the use of mother functions g(t) providing the so called analyzing wavelet [4]:

$$g_{b,a}(t) = \frac{1}{a}\, g\left(\frac{t-b}{a}\right) \qquad (1)$$

The wavelet transform of the signal s(t) may be calculated using the following expression:

$$S(b,a) = \int_{-\infty}^{+\infty} g^*_{b,a}(t)\, s(t)\, dt = \int_{-\infty}^{+\infty} G^*_{b,a}(\omega)\, S(\omega)\, d\omega \qquad (2)$$

where the second integral is expressing the wavelet transform in terms of the Fourier transform. The wavelet transform is invariant to dilation and translation, which makes it an interesting tool to be used in signal analysis and synthesis [4].

Practical implementations of the wavelet transform were augmented owing to the work of Daubechies [5][6] and to the programming skills of Coifman, Meyer and Wickerhauser [7]. In Fig. 2a are presented two examples of signals affected by clicks. In the upper portion of Fig. 2a are plotted time-domain characteristics of a signal with audible clicks, but these clicks are not to be seen in the plot (are hidden under the signal envelope). The lower portion of the Fig. 2a presents music affected with clearly seen clicks. In Fig. 2b are plotted results of marking the clicks with the wavelet detection procedure. These markers correspond to the clicks either visible or hidden in the signals plotted in Fig. 2a. The same wavelet procedure was used to detect impulses originated from prerecorded percussive sounds (hi-fi equipment). Results are presented in Fig. 3a (signals) and in Fig. 3b (impulse markers).

Owing to the advanced implementation of the wavelet analysis to the NeXT workstation [7] it was possible to detect strong clicks in audio signal (Fig.4a) and some smaller clicks hidden under the signal envelope (Fig. 4b).

The training material for the learning algorithm was prepared using both: manual method of extraction of signal portions with the use of the sound editor especially elaborated for this purpose by the authors [3] and with the use of the wavelet extractor implemented to the NeXT workstation. The large amount (some hundreds cases) of clicks, cracks and hiss examples were acquired. Moreover, some dozens of percussive and transient sound samples were added. The rewritable optical disks available at the NeXT workstations proven to be helpful in the mentioned task.

3. Perceptron algorithm

Neural network models are capable to learn dependencies existing in the data acquired from consecutive sample packets. The neural network was applied as the basic perceptron model allowing to detect individual clicks and cracks. The program written by the authors in C language for the NeXT workstation allows for the defining optional perceptron structures. The elaborated algorithm is described in the Appendix A.

Training of the perceptron model implemented hitherto some dozens of both desired (musical) and parasite impulse patterns that were input to the neural net having 64 inputs for sound samples delivered in 64-bit segments, one 30-neuron and one 10-neuron hidden layer and a single-neuron output layer. Transfer functions in each neuron excluding the output neuron was the hyperbolic tangent. The output neuron used rectangular transfer function. The network was converging in most cases after 10.000 to 50.000 iterations, both for clicks and for musical impulse patterns. As it was proven by the obtained results, the performance of the learning algorithm during the click detection is obviously exceeding the wavelet extraction procedure features that was used to prepare the training material. In Fig. 2c are seen impulses detected by the neural net. The comparison to Fig. 2b shows that the amount of detected clicks is much higher than in the case of the wavelet procedure. The speed of processing during the recognition phase is also much bigger. Moreover, the perceptron-based recognition is rarely making mistakes in the case of percussive sounds (see Fig. 3c).

4. Adaptive filtering for the cancellation of continuous noise

This attempt to diminish noise using the Wiener filtering is based on the modified approach to the widely known method. The modification consists in the implementation of multiple Wiener filter in the critical bands of hearing. Since subjective perceiving is decisive to the assessment of sound quality, such a selection was justified by psychophysiological principles.

The principle of classic Wiener filtering was ilustrated in Fig. 5. The filter processes signal yn, producing possibly the best estimate $\hat{x}_n$ of signal xn and then removes it from the output signal en. Consequently, the output signal en is no longer correlated to any current nor any past values of yn, thus $E[e_n y_{n-i}] = 0$ for i = 0,1,2,...,M. The x_n provides an optimal estimate of this part of signal xn, that was correlated to yn. It results from the fact that if xn = x1(n) + x2(n) and $R_{x_2y} = 0$, than $R_{xy} = R_{x_1y}$. Hence, the Wiener filter can be conceived as an optimal separator between the two components of signal x1(n) and x2(n). The basic filtering operations are described by the relationships (3) and (4), and optimal weights of filter h(n); n = 0,1,...,M can be calculated on the basis of the equation (5) [9].

$$\hat{x}_n = \sum_{i=n-M}^{n} h(n-i)y_i = h(0)y_n + h(1)y_{n-1} + \ldots + h(M)y_{n-M} \tag{3}$$

$$e_n = x_n - \hat{x}_n \tag{4}$$

$$\begin{bmatrix} \mathbf{R}_{yy}(\mathbf{0}) & \mathbf{R}_{yy}(\mathbf{1}) & R_{yy}(2) & \ldots & R_{yy}(M) \\ \mathbf{R}_{yy}(\mathbf{1}) & \mathbf{R}_{yy}(\mathbf{0}) & R_{yy}(1) & \ldots & R_{yy}(M-1) \\ \mathbf{R}_{yy}(\mathbf{2}) & \mathbf{R}_{yy}(\mathbf{1}) & R_{yy}(0) & \ldots & R_{yy}(M-2) \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \mathbf{R}_{yy}(\mathbf{M}) & \mathbf{R}_{yy}(\mathbf{M-1}) & R_{yy}(M-2) & \ldots & R_{yy}(0) \end{bmatrix} = \begin{bmatrix} h(0) \\ h(1) \\ h(2) \\ \vdots \\ h(M) \end{bmatrix} = \begin{bmatrix} R_{yy}(0) \\ R_{yy}(1) \\ R_{yy}(2) \\ \ldots \\ R_{yy}(M) \end{bmatrix} \tag{5}$$

The Wiener filter was constructed on the basis of Widrow-Hoff algorithm that allows to not determine explicitely the correlation matrixes R_{xx} and R_{xy}.

The principle of working of adaptive algorithm is ilustrated in Fig. 5. The output error signal e_n is taken from the output and is used to the updating of filter weights. Provided x_n a y_n are correlated somehow, the filter will accommodate its weights until the mentioned correlation is removed from the output error signal e_n. Provided x_n contains both: desired signal and parasite noise and if y_n is the noise correlated somehow to the noise being part of x_n, then y_n is correlated only to the noisy components of x_n. The adaptive filter accommodates to the removing of such a correlation. Specifically, the filter produces a copy of the noisy part of x_n and subsequently removes it. The main problem related to the filter design is the selection of the right length of FIR structure. The complexity of this problem growths proportionally to the processed bandwidth. Processing of full audio band would be improved by the splitting the spectrum into many subbands in order to allow the selection of filter order separately in each part of the frequency band. As the effect of filtering is to be assessed subjectively on the basis of hearing mechanisms, thus it is reasonable to select the subbands boundaries conforming the critical bands parameters. Practically, it might be sufficient to use 15 similar type subbands [10].

The elaborated software package consists of 3 main modules: spectrum decomposing block, Wiener filtering block and signal composing block as in Fig. 6.

The filter bank used to decompose signal spectrum was constructed with split-to-halves filters similar to those used by Spille and Schroeder for the subband audio coding [10]. The quadrature mirror filters in the lattice form were applied. The construction of the individual filters is described in Appendix B.

5. Conclusions

Hitherto obtained results allow to appreciate the neural network algorithm as a tool for the detection of clicks affecting old recordings. The neural network approach brings the "human touch" to the recognition of clicks, because the training process is based on the human decision as to the meaning of individual impulse patterns. The discerning capability in the domain of music and non-music impulses is leading to obvious supremacy of the learning algorithms over other methods that may be applied to the detection of impulse noise in old recordings. The speed of processing is also much more satisfying.

Application of adaptive algorithms to the cancellation of continuous noise may be more effective, when applied to many frequency subbands, separately. The "musical noise" is less audible in such a case.

6. References

[1] Montresor S., Valiere J.C., Allard J.F., Baudry M.: The Restoration of Old Recordings by means of Digital Techniques. AES Preprint 2915 (G4), Montreux 1990.

[2] Montresor S., Valiere J.C., Allard J.F., Baudry M.: Evaluation of two interpolation methods applied to old recordings restoration. AES Preprint 3022 (C-7), Paris 1991.

[3] Czyzewski A., Kostek B., Roland-Mieszkowski M., Multi-task method for the digital restoration of old recordings. AES Preprint 3436, San Francisco 1992.

[4] Guillemain Ph., Kronland-Martinet R., Parameters estimation through continuous wavelet transform of synthesis of audio-sounds. AES Preprint 3009 (A-2), Paris 1991

[5] Daubechies I., Orthonormal bases of compactly supported wavelets, Comm. Pure & Appl. Math. 41 (1988)

[6] Meyer Y. ed., Wavelets and Applications, Springer Verlag, Berlin 1992

[7] Coifman R,. Meyer V., Wickenhauser M., Wavelet Packet Laboratory Application, Software packet elaborated in the Yale University, 1989.

[8] Czyzewski A., Kaczmarek A.: Multilayer knowledge base system for the speaker independent recognition of isolated words. RSKD'93, Banff, Can., 1993

[9] Orfandis S.J.: Optimum signal processing: An Introduction. Macmillan Publishing Company New York 1985.

[10] Spille J., Schröder E.F.: Design of an Optimum Filterbank for High Qality Subband Audio Coding. AES Preprint No. 2749, Hamburg, March 1989.

APPENDIX A NEURAL NETWORK ALGORITHM

The program written in C language for the NeXT workstation allows for the defining optional perceptron structures. The program was elaborated in the first version for the purpose of speech recognition [8]. As it was proven by experimental results, it was possible to modify this algorithm and to implement it successfully to the detection of impulses in audio signal. The number of layers and neurons in each layer can be selected by the operator. The hyperbolic tangent was used as a transfer function for each neuron. Hence, neuron output states are determined by the values of the following function:

$$y = \mathrm{th}\,(\sum_i w_i\, x_i + b_j) \tag{AA1}$$

where: x_i - input signals,
w_i - i-th synapse weights,
b_j - neuron biases.

The modified by the authors back propagation algorithm was used in the perceptron learning procedure. The neural network state is determined by the status vector:

$$\overline{S} = [w_1, w_2, ..., w_p, b_1, b_2, ...,b_r] \tag{AA2}$$

where: p - number of synaptical connections, r - number of neurons,
w_i - weight of the i-th connection, b_j - bias of the j-th neuron.

Denoting Y_{ijk} as the state of i-th output while the j-th element of k-th set is present at the perceptron input one can obtain:

$$Y_{ijk} = f\,(\,\overline{S}\,) \tag{AA3}$$

The total output error value may be calculated using the next relationship:

$$E = \sum_{k=1}^{K} \sum_{j=1}^{j_k} \sum_{i=1}^{n} (Y_{ijk} - Z_{ik})^2 \tag{AA4}$$

where: K - number of sets,
j_k - number of k-th set elements,
n - number of outputs,
Z_{ik} - the desired state of i-th output for k-th set.

In order to minimize the error rate the error gradient is calculated with regard to all elements of the vector $\overline{S}$, where $E=E(\overline{S})$:

$$\overline{\text{grad}}\ E = \left[\frac{\partial E}{\partial w_1}, \frac{\partial E}{\partial w_2}, ..., \frac{\partial E}{\partial w_p}, \frac{\partial E}{\partial b_1}, \frac{\partial E}{\partial b_2}, ..., \frac{\partial E}{\partial b_r}\right] \qquad \text{(AA5)}$$

Practically, instead of computing partial derivatives the following quotients are to be found:

$$\frac{\Delta E}{\Delta w_i}; \frac{\Delta E}{\Delta b_j} \qquad i = 1, ..., p; \quad j = 1, ..., r \qquad \text{(AA6)}$$

At first, increments $\Delta w_1,..,\Delta w_p,\Delta b_1,...,\Delta b_r$ are all set equal to a small number Δ, for example to $\Delta = 10^{-4}$. Further, these increments are multiplied by the coefficient equal to 0.9 in each iteration step. Correspondingly, the need is fulfilled to estimate the gradient more precisely during the approaching the end of this value computations. Subsequently, the new state of the network is found, such that the following term is fulfilled:

$$E (\overline{S}_n^j) < E (\overline{S}_o^j) \qquad \text{(AA7)}$$

where: $\overline{S}_o^j$ - an output status vector,
$\overline{S}_n^j$ - the consecutive output status vector.

It is also assumed that:

$$\overline{S}_n^j = \overline{S}_n(j) = \overline{S}_p - \overline{\text{grad}}\ E \ \ (\text{step}) \ \ j \qquad \text{(AA8)}$$

where: step - certain small value equal to Δ at the training beginning,
j - variable to be described below.

The coefficient j in the relationship (AA8) is an integer number varying accordingly to the following rule:

$$\bigvee_{j>1} \bigwedge_{i<j} \{E [\overline{S}_n(i)] > E [\overline{S}_n(i+1)]\} \cap \{E [\overline{S}_n(j)] < E [\overline{S}_n(j+1)]\} \qquad \text{(AA9)}$$

The smallest number j fulfilling the above relationship (AA9) is used to the calculation of the final status vector $\overline{S}_n$ for a given iteration step. The result $j = 1$ is equivalent to the reaching the local error minimum, thus it is necessary to increase the Δ value, so that the next iteration step based on the revised gradient calculation result is allowing to escape from the minimum zone. Hence, it is possible to resume error minimization procedure starting far from the minimum zone.

APPENDIX B PRINCIPLES OF LATTICE -TYPE WIENER FIR FILTER

The adaptive filtration as in Fig. 6 was based on FIR-type Wiener filter in the lattice form shown in Fig. 8. The principle of working of this type of the filter is as follows:

A. Initialisation (AB1) using $e_0^+(n)=e_0^-(n)=y_n$ and subsequently iteration (AB2) for p=0,1,...,M-1 with $\gamma_{p+1}(n)$ updated using (AB3).

$$\begin{aligned} e_{p+1}^+(n) &= e_p^+(n)-\gamma_{p+1}(n)e_p^-(n-1) \\ e_{p+1}^-(n) &= e_p^-(n-1)-\gamma_{p+1}(n)e_p^+(n) \end{aligned} \qquad \text{(AB1)}$$

B. Calculation of $d_{p+1}(n)$ for p=0,1,...,M-1 using (AB2).

$$d_{p+1}(n)=\sum_{k=0}^{n}\lambda^{n-k}\left[e_p^+(k)^2+e_p^-(k-1)^2\right]=\lambda d_{p+1}(n-1)+\left[e_p^+(n)^2+e_p^-(n-1)^2\right] \qquad \text{(AB2)}$$

C. Calculation of $\gamma_{p+1}(n+1)$ for p=0,1,...,M-1 using (AB3).

$$\gamma_{p+1}(n+1)=\gamma_{p+1}(n)+\frac{\beta}{d_{p+1}(n)}\left[e_{p+1}^+(n)e_p^-(n-1)+e_{p+1}^-(n)e_p^+(n)\right] \qquad \text{(AB3)}$$

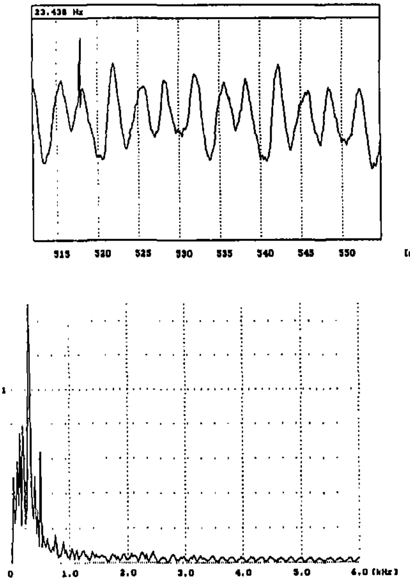
D. Calculation of $d_p^-(n)$ for p=0,1,...,M using (AB4)

$$d_p^-(n)=\sum_{k=0}^{n}\lambda^{n-k}e_p^-(k)^2=\lambda d_p^-(n-1)+e_p^-(n)^2 \qquad \text{(AB4)}$$

E. Calculation of $g_p(n+1)$ for p=0,1,...,M using (AB5)

$$g_p(n+1)=g_p(n)+\frac{\beta}{d_p^-(n)}e_n e_p^-(n) \qquad \text{(AB5)}$$

(a)



(b)

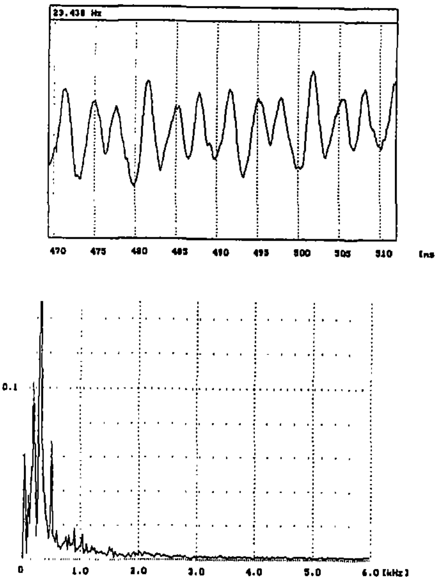


Fig. 1a Spectral analysis of the signal containing a click (a) and spectrum of neighboring fragment of the same signal without any clicks (b).

(a)

(b)

(c)

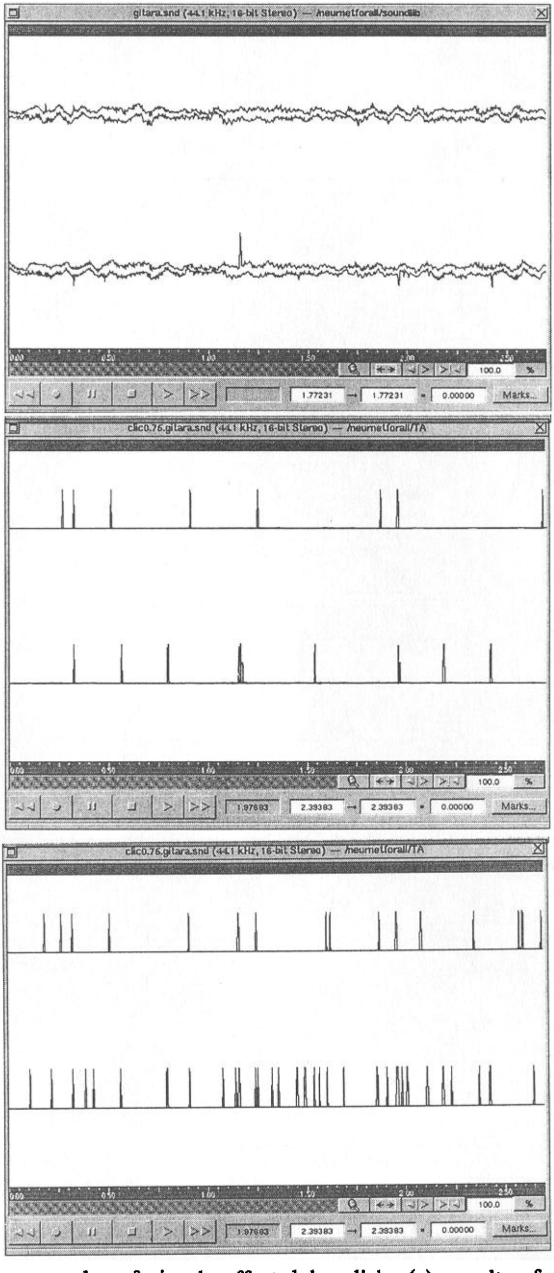


Fig. 2 Two examples of signals affected by clicks (a); results of marking the clicks with the wavelet detection procedure (b) and results of click detection by the perceptron algorithm (c).

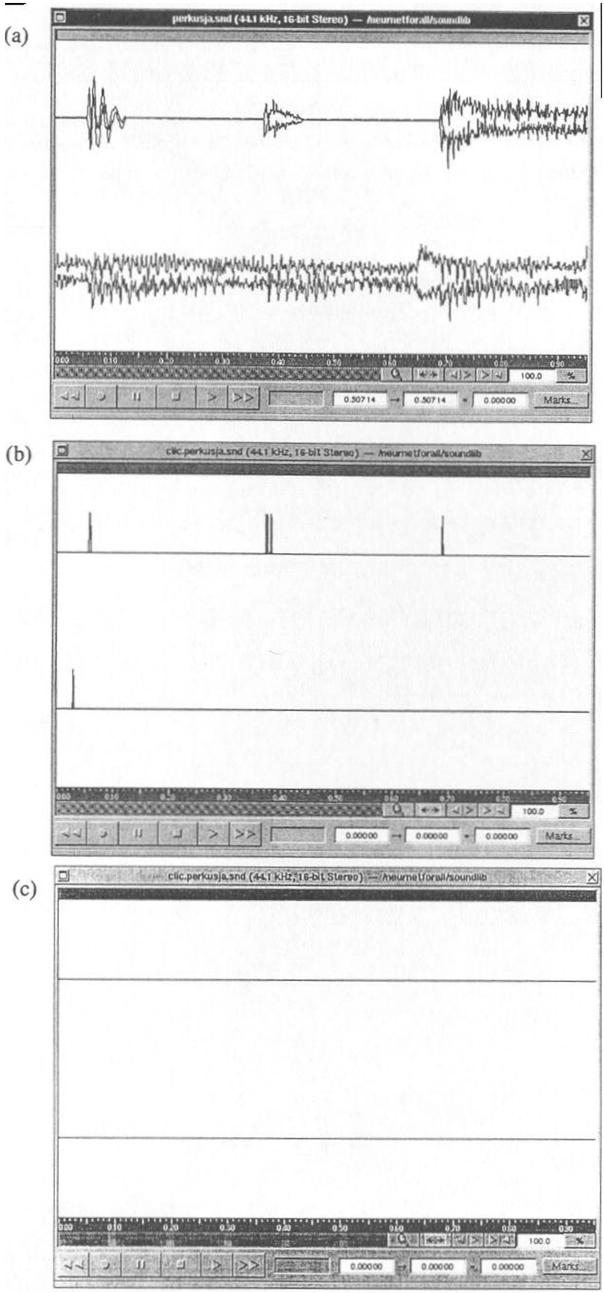
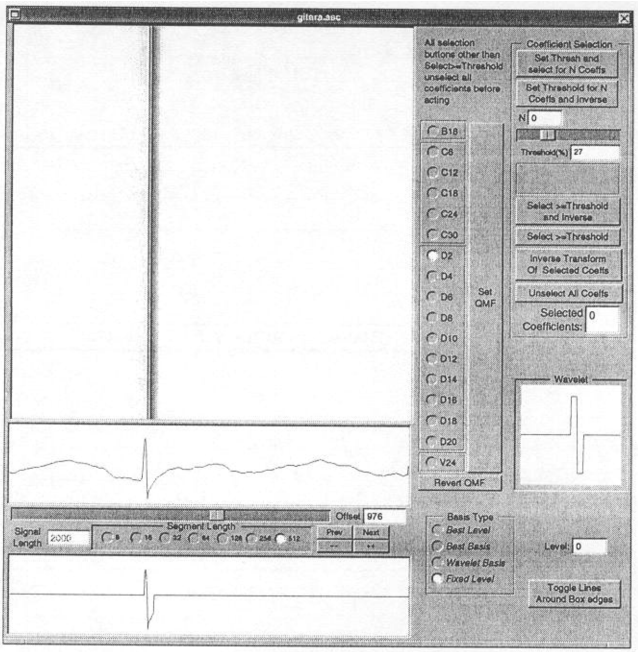
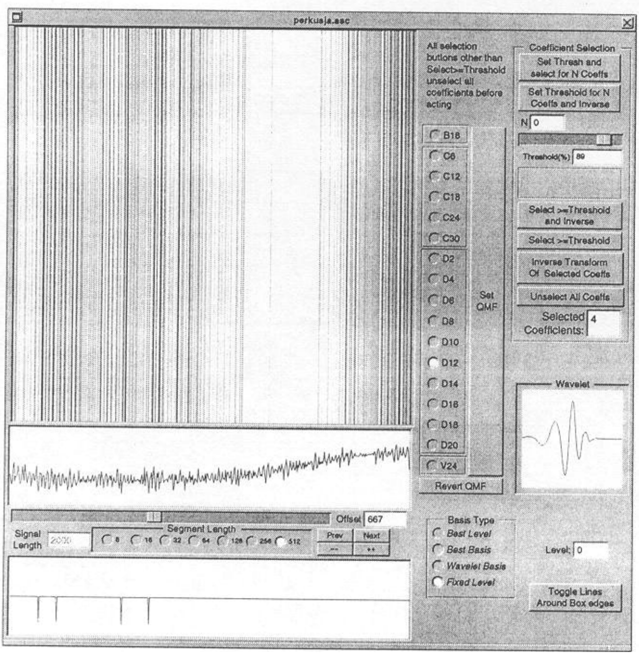


Fig 3 Percussive sounds (a); corresponding wavelet threshold analyses (b) and detection of impulses by the perceptron algorithm (c).

(a)



(b)



Analyzing tool: "Wavelet Packet Laboratory Application" elaborated in Yale University for the NeXT workstation [7]. Analyses with Quadrature Mirror Filtering. Upper part of the screen: wavelet phase plane, below: original signal, lower part of the screen: threshold resynthesis of the signal (extracted click pattern).

Fig.4 Wavelet analyses of signal containing strong click (a) and some smaller clicks hidden under the signal envelope (b).

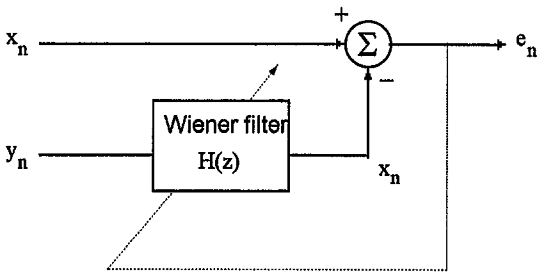


Fig. 5 Wiener filter as an optimal separator of signal components $\hat{x}_n = \sum_{i=0}^{M} h(i) y_{n-i}$

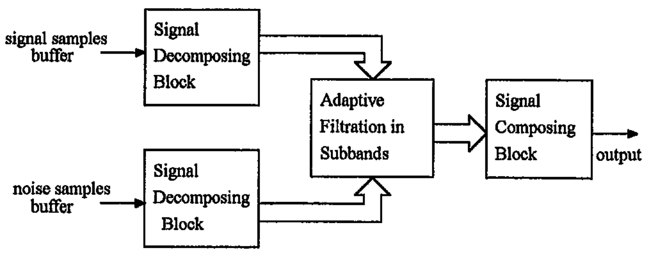


Fig. 6 Lay-out of the noise reduction system.

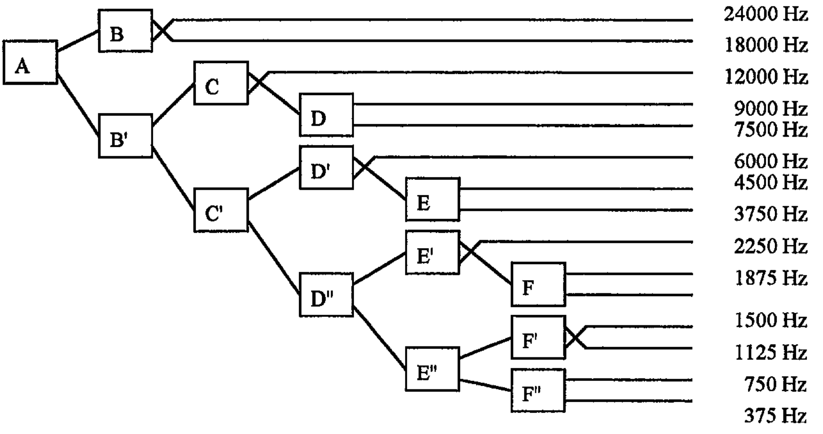


Fig. 7 Filter band used to spectrum splitting. Selection of frequency bands as in the publication [10].

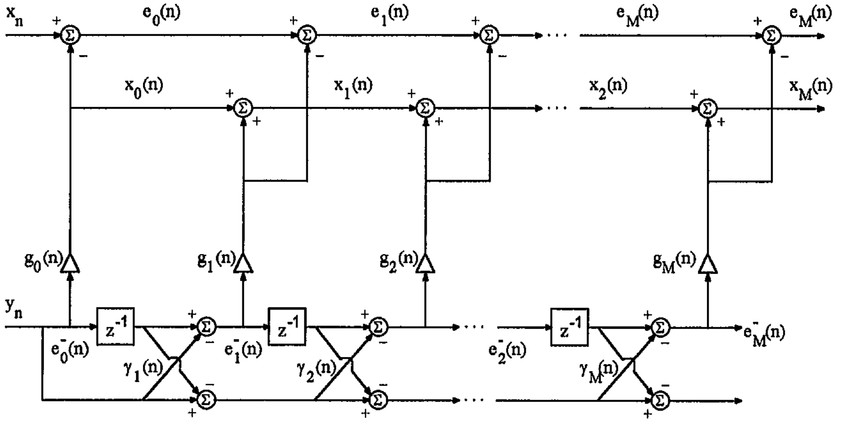


Fig. 8 Lattice Wiener filter lay-out [9].

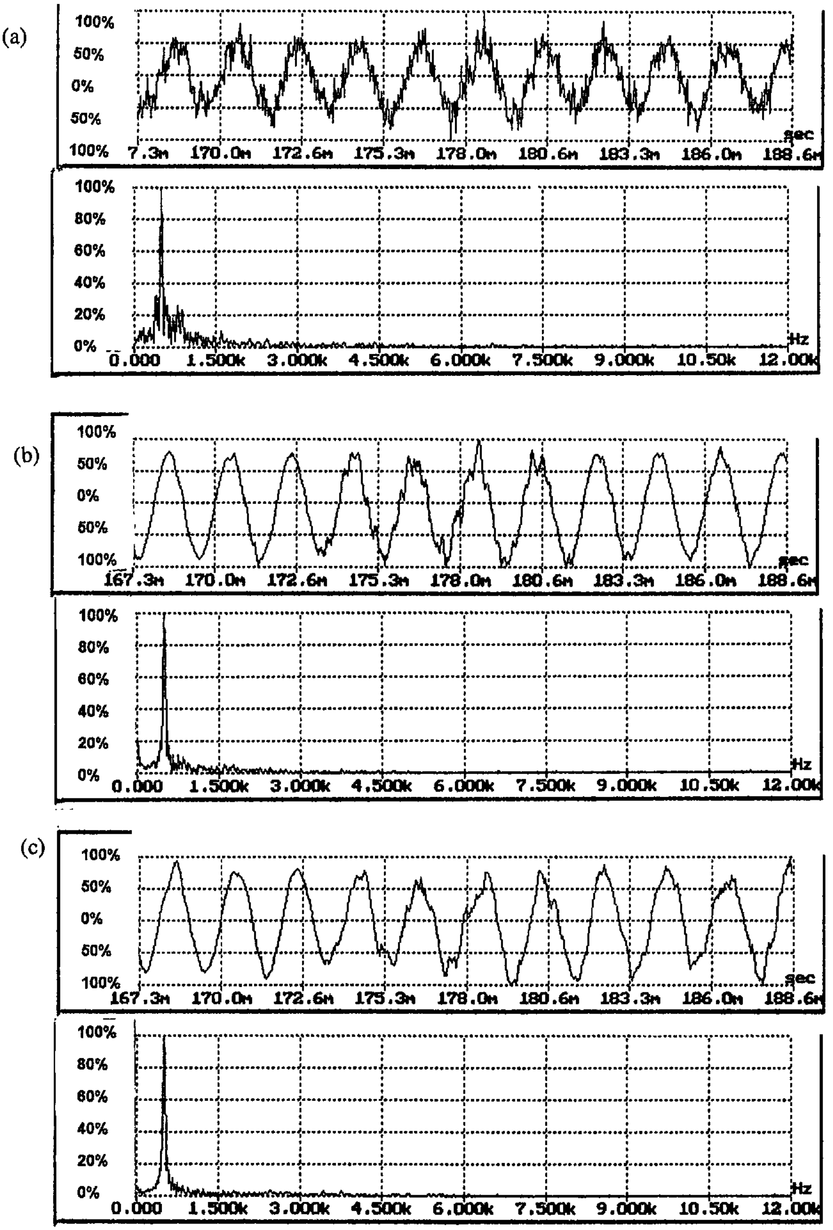


Fig. 9 Effects of adaptive filtering. Original harmonic signal affected by noise (a); results of filtering with 8th-order QMF Wiener filter (b); results of filtering with 32-order QMF Wiener filter (c)