

## Chapter

# A Voice Signal Filtering Methods for Speaker Biometric Identification

*Eugene Fedorov, Tetyana Utkina and Tetyana Neskorodeva*

## Abstract

The preliminary stage of the personality biometric identification on a voice is voice signal filtering. For biometric identification are considered and in number investigated the following methods of noise suppression in a voice signal. The smoothing adaptive linear time filtering (algorithm of the minimum root mean square error, an algorithm of recursive least squares, an algorithm of Kalman filtering, a Lee algorithm), the smoothing adaptive linear frequency filtering (the generalized method, the MLEE (maximum likelihood envelope estimation) method, a wavelet analysis with threshold processing (universal threshold, SURE (Stein's Unbiased Risk Estimator)-threshold, minimax threshold, FDR (False Discovery Rate)-threshold, Bayesian threshold were used), the smoothing non-adaptive linear time filtering (the arithmetic mean filter, the normalized Gauss's filter, the normalized binomial filter), the smoothing nonlinear filtering (geometric mean filter, the harmonic mean filter, the contraharmonic filter, the  $\alpha$ -trimmed mean filter, the median filter, the rank filter, the midpoint filter, the conservative filter, the morphological filter). Results of a numerical research of denoising methods for voice signals people from the TIMIT (Texas Instruments and Massachusetts Institute of Technology) database which were noise an additive Gaussian noise and multiplicative Gaussian noise were received.

**Keywords:** announcer biometric identification, voice signal filtering methods, the smoothing adaptive linear filtering, a wavelet analysis threshold processing, the smoothing non-adaptive linear filtering, the smoothing nonlinear filtering

## 1. Introduction

The preliminary stage of the personality biometric identification on a voice is voice signal filtering. Methods of a signal cleaning from noise arose and gained broad development in the twentieth century. With development a wavelet analysis joined normal time and frequency filters a wavelet filter.

Noise (interference) is the sound of an undesirable additional source added to a desired signal during its record or transfer on communication channel.

Noise can be classified by the following features: periodicity/aperiodicity; additive/multiplicative; continuity/impulsivity; to band width in a signal spectrum; color.

By continuity/impulsivity, noises are divided into: continuous; pulse (point); continuous and pulse.

Noises are divided by band width on:

- narrowband (noise with a continuous spectrum less than one octave);
- broadband (noise with a continuous spectrum more than one octave).

From color noise by the most difficult for filtering the white noise which has a uniform energy spectrum in all frequency range. The most widespread kind of a white noise is Gauss's noise.

Additive and multiplicative continuous and continuous impulse noises are removed from a signal by means of a wavelet analysis with threshold processing, the smoothing linear and many nonlinear filters, spectral subtraction. Impulse noises are removed many smoothing nonlinear filters. Additive aperiodic noise is removed low-frequency filters. Additive periodic noise is removed the bandpass and rejection filters.

## 2. The smoothing adaptive linear time filtering

*Adaptive linear time filters* call linear filters with adaptive impulse response function [1–3].

### 2.1 Algorithm of the minimum root mean square error

Algorithm of the minimum root mean square error which is applied to a signal  $x(n)$  size  $N$ , is as follows [1]:

1. Impulse response function initialization

$$\mathbf{h} = \begin{pmatrix} h_1 \\ \dots \\ h_{2M+1} \end{pmatrix} = \begin{pmatrix} h(-M) \\ \dots \\ h(M) \end{pmatrix} = \begin{pmatrix} 0 \\ \dots \\ 0 \end{pmatrix}. \quad (1)$$

2.  $n = M$ .

3. Noise vector forming from a noise signal

$$\mathbf{v} = \begin{pmatrix} v_1 \\ \dots \\ v_{2M+1} \end{pmatrix} = \begin{pmatrix} v(n - M) \\ \dots \\ v(n + M) \end{pmatrix}. \quad (2)$$

4. Signal filtering (receiving noise estimates)

$$\tilde{z}(n) = \mathbf{h}^T \mathbf{v}. \quad (3)$$

5. Error signal current value calculation

$$e(n) = x(n) - \tilde{z}(n). \quad (4)$$

6. Impulse response function adaptation

$$\mathbf{h} = \mathbf{h} + \mu \mathbf{v} e(n), \quad (5)$$

where  $0 < \mu < 1$ .

7. If  $n < N - M$  then  $n = n + 1$ , go to a step 2.

The signal is algorithm work result  $e(n)$ ,  $e(n) \approx s(n)$ .

## 2.2 Recursive least squares algorithm

Recursive least squares algorithm which is applied to a signal  $x(n)$  size  $N$ , is as follows [1]:

1. Initialization of impulse response function and adaptation matrix

$$\mathbf{h} = \begin{pmatrix} h_1 \\ \dots \\ h_{2M+1} \end{pmatrix} = \begin{pmatrix} h(-M) \\ \dots \\ h(M) \end{pmatrix} = \begin{pmatrix} 0 \\ \dots \\ 0 \end{pmatrix}, \mathbf{P} = \begin{pmatrix} p_{11} & \dots & p_{1,2M+1} \\ \dots & \dots & \dots \\ p_{2M+1,1} & \dots & p_{2M+1,2M+1} \end{pmatrix} = \lambda \mathbf{I}, \quad (6)$$

where  $\lambda$ -regularization parameter which is small at a big ratio signal/noise and is big at a small ratio signal/noise.

1.  $n = M$ .

2. Noise vector forming

$$\mathbf{v} = \begin{pmatrix} v_1 \\ \dots \\ v_{2M+1} \end{pmatrix} = \begin{pmatrix} v(n - M) \\ \dots \\ v(n + M) \end{pmatrix}. \quad (7)$$

3. Signal filtering (receiving noise estimates)

$$\tilde{z}(n) = \mathbf{h}^T \mathbf{v}. \quad (8)$$

4. Error signal current value calculation

$$e(n) = x(n) - \tilde{z}(n). \quad (9)$$

5. Adaptive gain  $\Gamma$  vector calculation

$$\Gamma = \frac{\mathbf{P}\mathbf{v}}{\mathbf{v}^T \mathbf{P}\mathbf{v} + r}, \quad (10)$$

where  $0 < r < 1$ .

6. Estimates covariance matrix  $\mathbf{P}$  calculation

$$\mathbf{P} = \frac{1}{r} \left( \mathbf{P} - \frac{\mathbf{P}\mathbf{v}\mathbf{v}^T\mathbf{P}}{\mathbf{v}^T\mathbf{P}\mathbf{v} + r} \right). \quad (11)$$

7. Impulse response function calculation

$$\mathbf{h} = \mathbf{h} + \Gamma e(n). \quad (12)$$

8. If  $n < N - M$  then  $n = n + 1$ , go to a step 2.

The signal is algorithm work result  $e(n)$ ,  $e(n) \approx s(n)$ .

**2.3 Kalman filtering algorithm**

Kalman filtering algorithm which is applied to a signal  $x(n)$  size  $N$ , is as follows [1]:

1. Estimates and white noise covariance matrixes:

$$\mathbf{h} = \begin{pmatrix} h_1 \\ \dots \\ h_{2M+1} \end{pmatrix} = \begin{pmatrix} h^{(-M)} \\ \dots \\ h^{(M)} \end{pmatrix} = \begin{pmatrix} 0 \\ \dots \\ 0 \end{pmatrix}, \mathbf{P} = \begin{pmatrix} p_{11} & \dots & p_{1,2M+1} \\ \dots & \dots & \dots \\ p_{2M+1,1} & \dots & p_{2M+1,2M+1} \end{pmatrix} = \lambda \mathbf{I}, \quad (13)$$

$$\mathbf{Q} = \begin{pmatrix} q_{11} & \dots & q_{1,2M+1} \\ \dots & \dots & \dots \\ q_{2M+1,1} & \dots & q_{2M+1,2M+1} \end{pmatrix} = \sigma_1^2 \mathbf{I},$$

where  $\lambda$ —regularization parameter which is small at a big ratio signal/noise and is big at a small ratio signal/noise,

$\sigma_1^2$ —variance of a white noise of process which has null mean value.

1.  $n = M$ .

2. Noise vector forming from a noise signal

$$\mathbf{v} = \begin{pmatrix} v_1 \\ \dots \\ v_{2M+1} \end{pmatrix} = \begin{pmatrix} v(n - M) \\ \dots \\ v(n + M) \end{pmatrix}. \quad (14)$$

3. Signal filtering (receiving noise estimates)

$$\tilde{z}(n) = \mathbf{h}^T \mathbf{v}. \quad (15)$$

4. Error signal current value calculation

$$e(n) = x(n) - \tilde{z}(n). \quad (16)$$

5. Adaptive gain  $\Gamma$  vector calculation

$$\Gamma = \frac{\mathbf{P}\mathbf{v}}{\mathbf{v}^T\mathbf{P}\mathbf{v} + \sigma_2^2}, \quad (17)$$

where  $\sigma_2^2$ —variance of a white noise of measurement which has null mean value.

6. Estimates covariance matrix  $\mathbf{P}$  calculation

$$\mathbf{P} = \mathbf{P} - \frac{\mathbf{P}\mathbf{v}\mathbf{v}^T\mathbf{P}}{\mathbf{v}^T\mathbf{P}\mathbf{v} + \sigma_2^2} + \mathbf{Q}. \quad (18)$$

7. Impulse response function calculation

$$\mathbf{h} = \mathbf{h} + \Gamma e(n). \quad (19)$$

8. If  $n < N - M$  then  $n = n + 1$ , go to a step 2.

The signal is algorithm work result  $e(n)$ ,  $e(n) \approx s(n)$ .

**2.4 Lee algorithm**

Lee algorithm [2] which is applied to a signal  $x(n)$  size  $N$ , is as follows:

1. Calculate local mean for each signal sample

$$\mu(n) = \frac{1}{2M+1} \sum_{m \in U_n} x(m), n \in \overline{M, N-M+1}, \quad (20)$$

where  $U_n$ —sample  $n$  neighborhood size  $2M + 1$ .

2. Calculate local variance for each signal sample

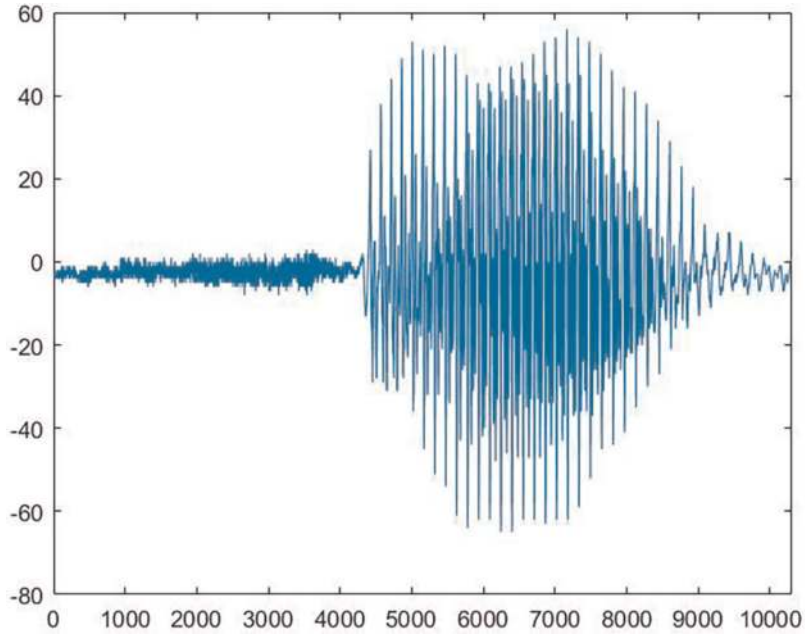
$$\sigma_x^2(n) = \frac{1}{2M+1} \sum_{m \in U_n} x^2(m) - \mu^2(n), n \in \overline{M, N-M+1}. \quad (21)$$

3. Calculate variance for each signal sample

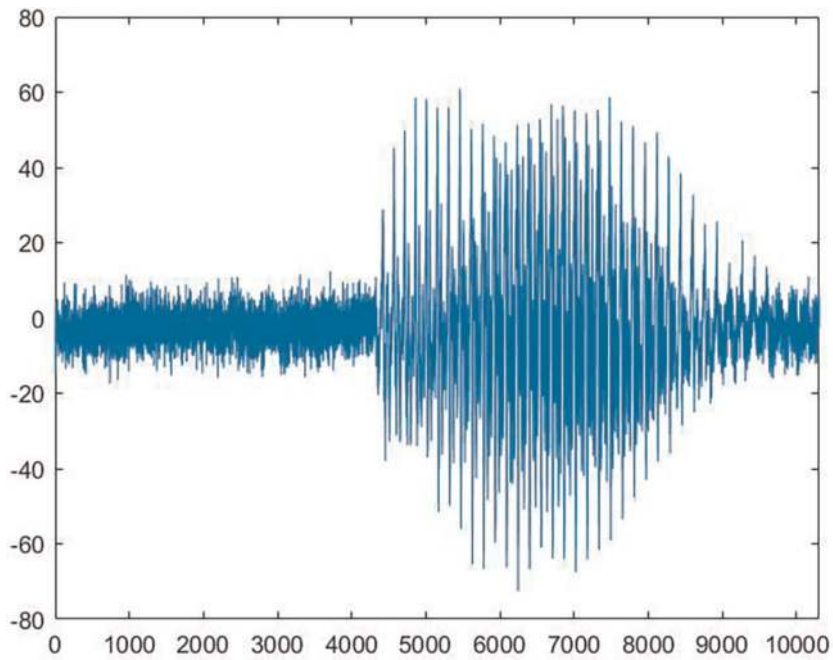
$$\sigma_v^2 = \frac{1}{N-2M} \sum_n \sigma_x^2(n), n \in \overline{M, N-M+1}. \quad (22)$$

4. Execute adaptive filtering of a signal

$$s(n) = \sum_{m=-M}^M h(m) x(n-m), n \in \overline{M, N-M+1}. \quad (23)$$

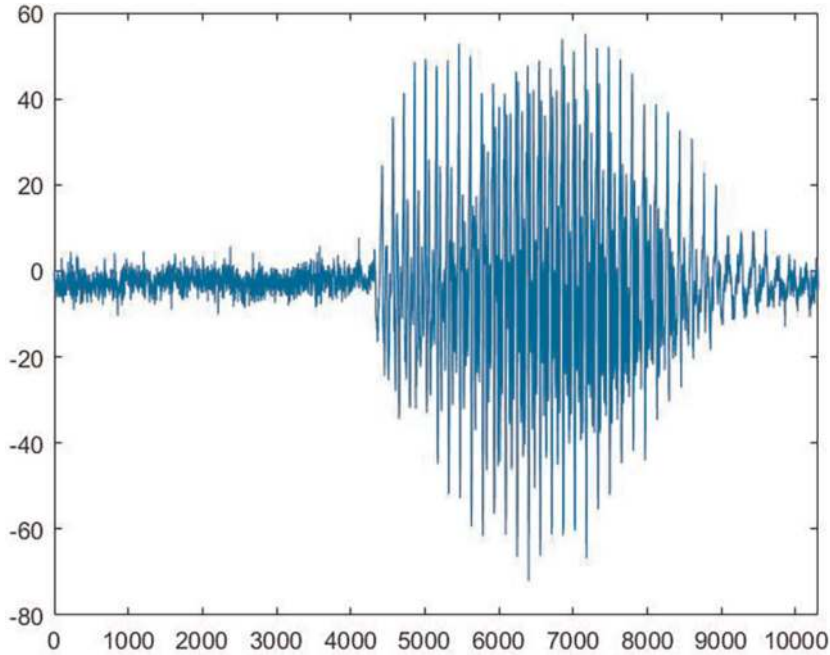


**Figure 1.**  
Source signal for smoothing adaptive linear time filtering.



**Figure 2.**  
A signal with an additive Gaussian noise for smoothing adaptive linear time filtering.

$$h(m) = \begin{cases} \frac{1}{2M+1} + \frac{\max\{0, \sigma_x^2(n) - \sigma_v^2\}}{\sigma_x^2(n)} \left(1 - \frac{1}{2M+1}\right), & m = 0 \\ \frac{1}{2M+1} - \frac{\max\{0, \sigma_x^2(n) - \sigma_v^2\}}{\sigma_x^2(n)} \cdot \frac{1}{2M+1}, & \text{otherwise} \end{cases} \quad (24)$$



**Figure 3.**  
 The signal denoised by the adaptive filter.

### Example

In **Figure 1** the source signal, is presented on **Figure 2**—noisy (additive white is added the noise with a mean 0 and variance 0.001 is Gaussian), on **Figure 3**—filtered and  $M = 1$ .

### 3. The smoothing adaptive linear frequency filtering

*Adaptive linear frequency filters* call linear filters with adaptive transfer function [4]. The smoothing adaptive linear frequency filtering is called *spectral subtraction*.

Let  $X_p(k)$ —a noisy signal spectrum of on  $p$ -th a frame,  $V(k)$ —mean noise spectrum,  $S_p(k)$ —a mean of the restored signal on  $p$ -th a frame.

Adaptive linear frequency filtering represents the inverse discrete Fourier transform of performing adaptive transfer function of the filter  $H_p(k)$  on  $p$ -th frame and signal spectrum  $X_p(k)$  on  $p$ -th a frame in a next form

$$y(n) = \frac{1}{N} \sum_{k=0}^{N-1} (X(k)H(k))e^{j\frac{2\pi nk}{N}}. \quad (25)$$

The following spectral subtraction methods are selected [1]:

1. General method (proposed Beruti, Schwartz and Makhoul)

$$S_p(k) = H_p(k)X_p(k), \quad (26)$$

$$H_p(k) = \begin{cases} G \left( \frac{|X_p(k)|^\gamma - \alpha|V(k)|^\gamma}{|X_p(k)|^\gamma} \right)^{1/\gamma}, & G \left( \frac{|X_p(k)|^\gamma - \alpha|V(k)|^\gamma}{|X_p(k)|^\gamma} \right)^{1/\gamma} > \beta|V(k)|, \\ \beta|V(k)|, & \text{otherwise} \end{cases} \quad (27)$$

where  $G, \alpha, \beta, \gamma$ —parameters.

### 2. The Ball method

$$H_p(k) = \begin{cases} \frac{|X_p(k)| - |V(k)|}{|X_p(k)|}, & |X_p(k)| - |V(k)| > 0 \\ 0, & \text{otherwise} \end{cases} \quad (28)$$

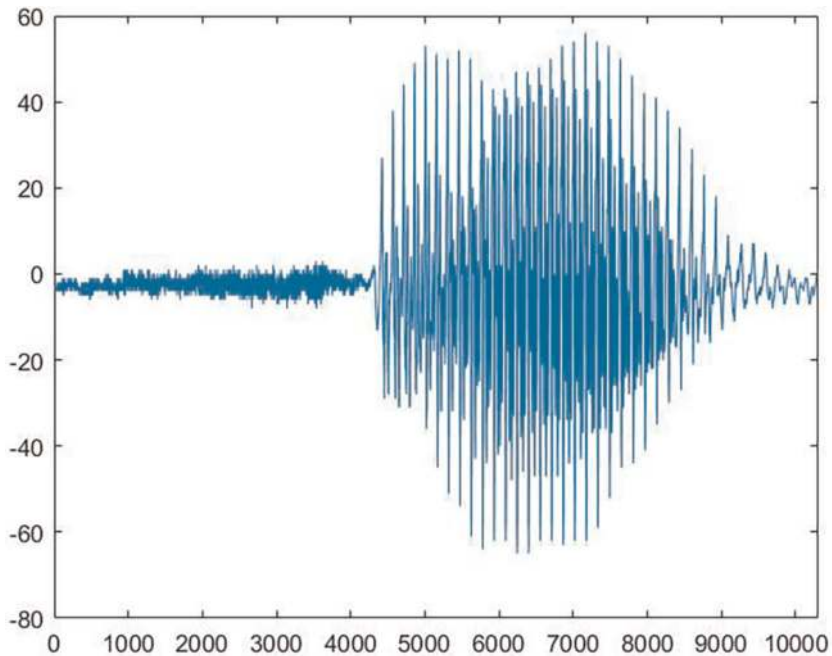
### 3. Wiener filtering

$$H_p(k) = \begin{cases} \frac{|X_p(k)|^2 - |V(k)|^2}{|X_p(k)|^2}, & |X_p(k)|^2 - |V(k)|^2 > 0 \\ 0, & \text{otherwise} \end{cases} \quad (29)$$

### 4. The MLEE method

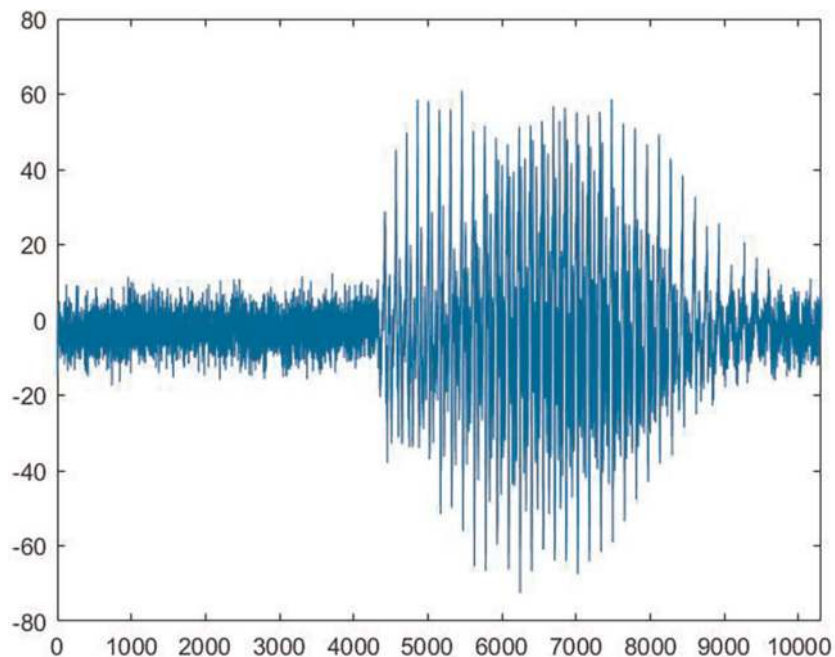
$$S_p(k) = H_p(k)X_p(k),$$

$$H_p(k) = \begin{cases} \frac{1}{2} + \frac{1}{2} \sqrt{\frac{|X_p(k)|^2 - |V(k)|^2}{|X_p(k)|^2}}, & |X_p(k)|^2 - |V(k)|^2 > 0 \\ 0, & \text{otherwise} \end{cases} \quad (30)$$



**Figure 4.** Source signal for smoothing adaptive linear frequency filtering.

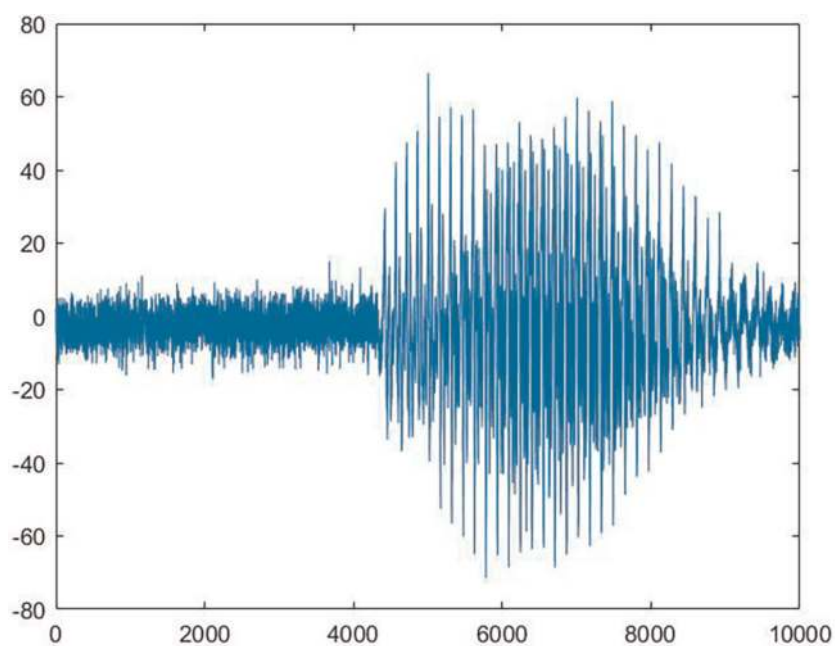




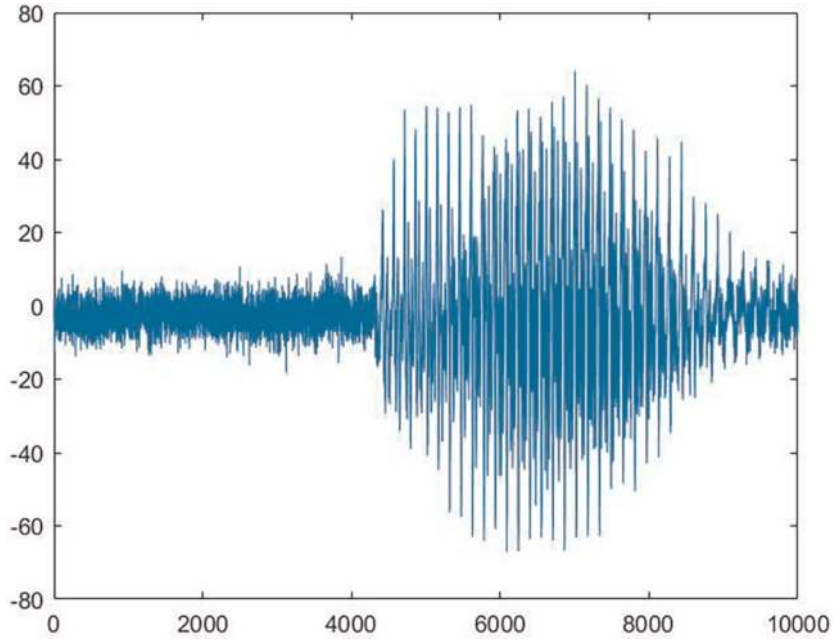
**Figure 5.**  
*A signal with additive Gaussian noise for smoothing adaptive linear frequency filtering.*

### **Example**

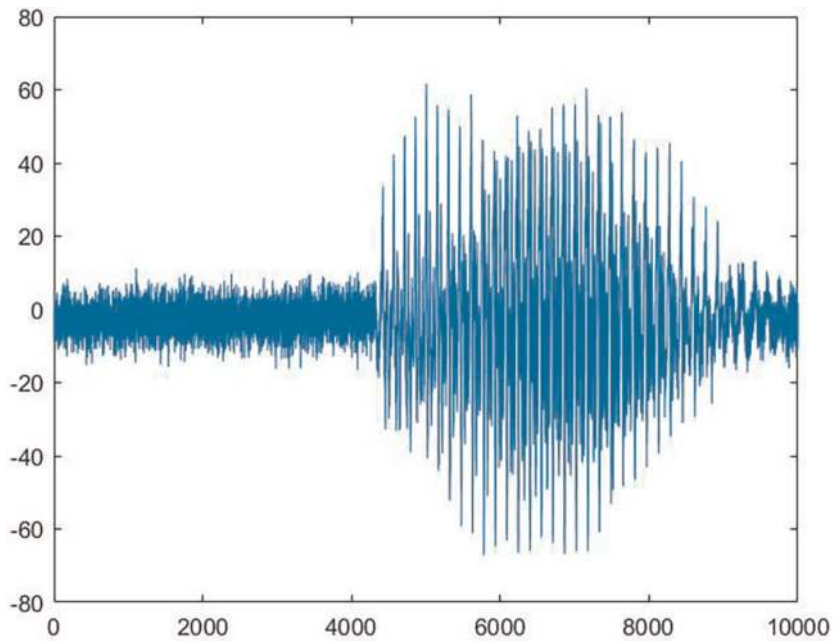
In **Figure 4** the source signal, is presented on **Figure 5**—noisy (additive white is added the noise with a mean 0 and variance 0.001 is Gaussian), the signals denoised by means of filtering according to general method ( $G = 1$ ,  $\gamma = 2$ ,  $\alpha = 6$ ,  $\beta = 0.1$ ) (**Figure 6**), Ball (**Figure 7**), Wiener (**Figure 8**), MLEE (**Figure 9**). For these methods



**Figure 6.**  
*The signal denoised by means of filtering according to general method.*



**Figure 7.**  
*The signal denoised by means of filtering according to Ball.*

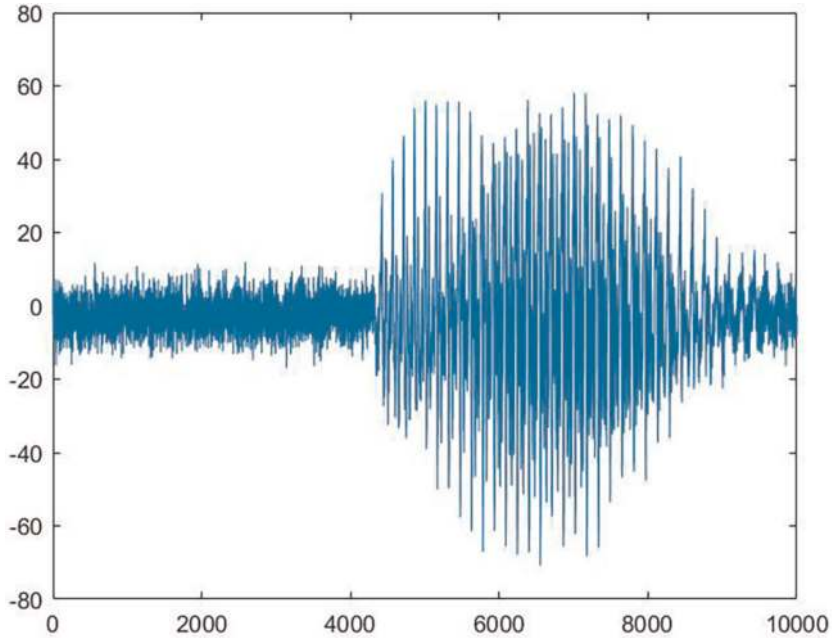


**Figure 8.**  
*The signal denoised by means of filtering according to Wiener.*

as frame length, it was selected  $\Delta N = 512$  (about 20 ms). In signal quality the word “Sasha” with a sampling rate of 22050 Hz, 8-bits, mono was selected.

#### 4. Wavelet analysis threshold processing

For a wavelet analysis the soft and rigid threshold processing is widely used [5].



**Figure 9.**  
 The signal denoised by means of filtering according to MLEE.

#### 4.1 Signal analysis

##### 1. Initialization.

Decompositions level number  $i = 1$ .

$$c_{0x} = s(x), x \in \overline{0, N/2^{i-1} - 1}, \quad (31)$$

where  $s(x)$ —original signal length  $N$ .

2. On the current  $i$ th the decomposition level signal convolution with impulse response functions of FIR-HP (Finite Impulse Response—High Pass) and FIR-LP (Finite Impulse Response—Low Pass) filter is executed  $g(k)$ ,  $h(k)$  respectively

$$d_{im} = \sqrt{2} \sum_{k=0}^{N_2/2^{i-1}-1} c_{i-1,k} g(k + 2m), m \in \overline{0, N/2^i - 1}, \quad (32)$$

$$c_{im} = \sqrt{2} \sum_{k=0}^{N_2/2^{i-1}-1} c_{i-1,k} h(k + 2m), m \in \overline{0, N/2^i - 1}. \quad (33)$$

3. If  $i < P$  then  $i = i + 1$ , go to a step 1.

#### 4.2 Decomposition coefficients conversion

1. Decompositions level number  $i = 1$ .

2. Create the vector arranged on increase

$$a_i = \left( |d_{i0}|, \dots, |d_{i,N/2^i-1}| \right), |d_{im}| < |d_{i,m+1}|. \quad (34)$$

3. Calculate noise standard deviation based on a received vector median

$$\sigma_i = \frac{\text{median}(a_i)}{0.6745}, \quad (35)$$

where  $\text{median}(x)$ —function which returns a median of a vector  $x$ .

4. Calculate one of the following thresholds

1. Calculate a universal threshold

$$T_i = \sigma_i \sqrt{2 \ln(N/2^i)}. \quad (36)$$

2. Calculate a SURE-threshold

1. Define a threshold based on minimal risk

$$r_{im} = 1 + \frac{-2(m-1) + \sum_{k=0}^{m-1} (a_{ik})^2 + (a_{im})^2 (N/2^i - 1 - m)}{(N/2^i)}, m \in \overline{0, N/2^i - 1}, \quad (37)$$

$$m^* = \arg \min_m r_{im}, m \in \overline{0, N/2^i - 1}, \tilde{T}_i = a_{im^*}. \quad (38)$$

2. Calculate a SURE-threshold based on the received threshold

$$T_i = \begin{cases} \sigma_i \sqrt{2 \ln(N/2^i)}, & \sum_{m=0}^{N/2^i-1} d_{im} - (N/2^i) \sigma_i^2 \leq \varepsilon_i \\ \tilde{T}_i, & \sum_{m=0}^{N/2^i-1} d_{im} - (N/2^i) \sigma_i^2 > \varepsilon_i \end{cases}, \varepsilon_i = \sigma_i^2 \sqrt{(N^f/2^i) \ln(N/2^i)^3}. \quad (39)$$

3. Calculate a minimax threshold

$$T_i = \begin{cases} \sigma_i (0.3936 + 0.1829 \ln(N/2^i)), & N/2^i > 32 \\ 0, & N/2^i \leq 32 \end{cases}. \quad (40)$$

4. Calculate a FDR-threshold

$$\mu_i = \frac{\sum_{m=1}^{N/2^i-1} a_{im}}{N/2^i - 1}, \Delta_{im} = \text{erfc} \left( \frac{1}{\sqrt{2}} \left| \frac{a_{im} - \mu_i}{\sigma_i} \right| \right) - q \frac{m}{N/2^i - 1}, m \in \overline{0, N/2^i - 1}, \quad (41)$$

$$m^* = \arg \min_m (\operatorname{sgn}(\Delta_{im}) \operatorname{sgn}(\Delta_{i,m+1})), m \in \overline{0, N/2^i - 2}, T_i = a_{im^*}, \quad (42)$$

where  $q$ —parameter,  $q \in (0, 0.5]$ , and it is normal  $q = 0.05$ ,  $\operatorname{erfc}(x)$ —additional function of errors.

5. Calculate a Bayesian threshold (using Quasi-Cauchy distribution which is the most effective)

1. Calculate function  $\beta$  (using Quasi-Cauchy distribution)

$$\beta(a_{im}) = \frac{g(a_{im})}{\varphi(a_{im})} - 1, m \in \overline{0, N/2^i - 1}, g(x) = (\gamma \cdot \varphi)(x) = \frac{1}{\sqrt{2\pi}} x^{-2} (1 - e^{-x^2/2}), \quad (43)$$

$$\varphi(x) = \frac{1}{\sqrt{2\pi}} e^{-x^2/2}, \gamma(x) = \frac{1}{\sqrt{2\pi}} \frac{\varphi(x) - |x|(1 - \Phi(x))}{\varphi(x)}, \quad (44)$$

where  $\varphi(x)$ —standard normal distribution density,  $\gamma(x)$ —Quasi-Cauchy's density of distribution.

2. Calculate the minimum parameter  $w_i$  value (using Quasi-Cauchy distribution)

$$w_i^{\min} = \frac{\frac{1}{2} (\tilde{T}_i)^2 e^{-(\tilde{T}_i)^2/2}}{1 + \Phi(\tilde{T}_i) - \tilde{T}_i \varphi(\tilde{T}_i) - \frac{1}{2}}, \tilde{T}_i = \sqrt{2 \ln(N/2^i)}. \quad (45)$$

3. Find parameter value  $w_i$  by the equation numerical solution on an interval  $[w_i^{\min}, 1]$

$$S_i(w_i) = \sum_{m=0}^{N/2^i-1} \frac{\beta(a_{im})}{1 + w_i \beta(a_{im})} = 0. \quad (46)$$

4. Find a Bayesian threshold  $T_i$  by the numerical solution of the equation on an interval  $[0, T^{\max}]$  (using Quasi-Cauchy distribution)

$$-\Phi(T_i) + T_i \varphi(T_i) + \frac{1}{2} + \frac{1}{2} (T_i)^2 e^{-(T_i)^2/2} (1/w_i - 1) = 0. \quad (47)$$

5. Execute one of the following thresholds processing

1. Execute soft threshold processing (for universal, minimax, Bayesian, a SURE-threshold)

$$\tilde{d}_{im} = \begin{cases} d_{im} - T_i, & d_{im} \geq T_i \\ d_{im} + T_i, & d_{im} \leq -T_i, m \in \overline{0, N/2^{i-1} - 1}. \\ 0, & |d_{im}| \leq T_i \end{cases} \quad (48)$$

2. Execute rigid threshold processing (for universal, minimax, Bayesian, SURE, a FDR threshold)

$$\tilde{d}_{im} = \begin{cases} d_{im}, & |d_{im}| > T_i \\ 0, & |d_{im}| \leq T_i \end{cases}, m \in \overline{0, N/2^{i-1} - 1}. \quad (49)$$

6. If  $i < P$  then  $i = i + 1$ , go to a step 1.

### 4.3 Signal design

1. Initialization.

Level number of recoveries  $i = P$ .

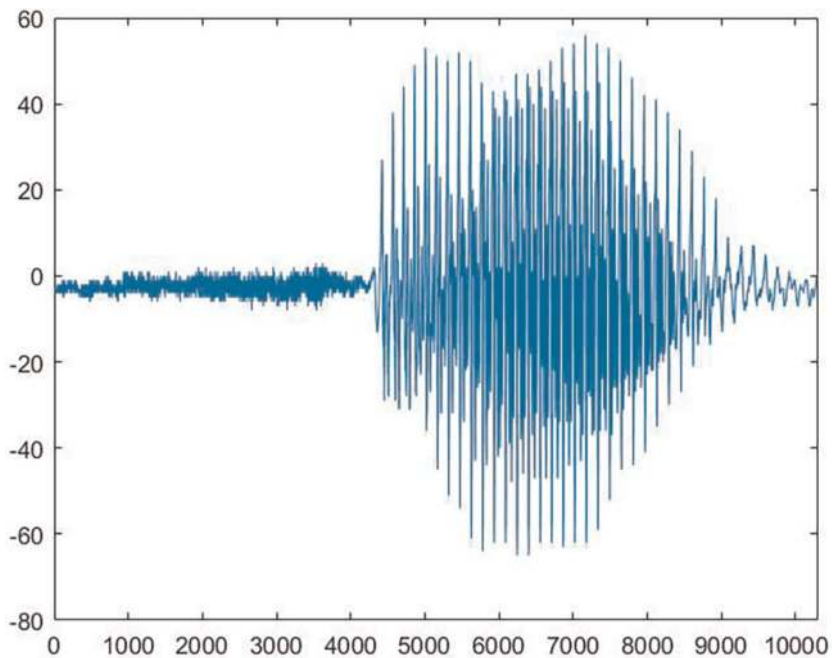
2. On the current  $i$ -th the recovery level signal convolution with impulse response functions of FIR-HP and FIR-LP filter is executed  $g(k)$ ,  $h(k)$  respectively

$$c_{i-1,n} = \sqrt{2} \sum_{m=0}^{N/2^i-1} c_{im}h(n + 2m) + \sqrt{2} \sum_{m=0}^{N/2^i-1} \tilde{d}_{im}g(n + 2m), n \in \overline{0, N/2^{i-1} - 1}. \quad (50)$$

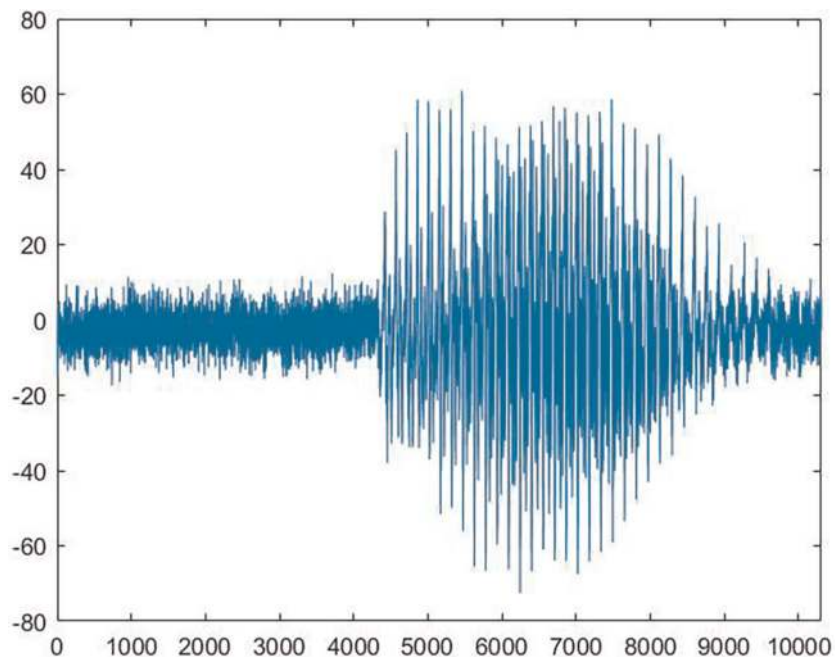
3. If  $i > 1$  then  $i = i - 1$ , go to a step 1.

#### Example

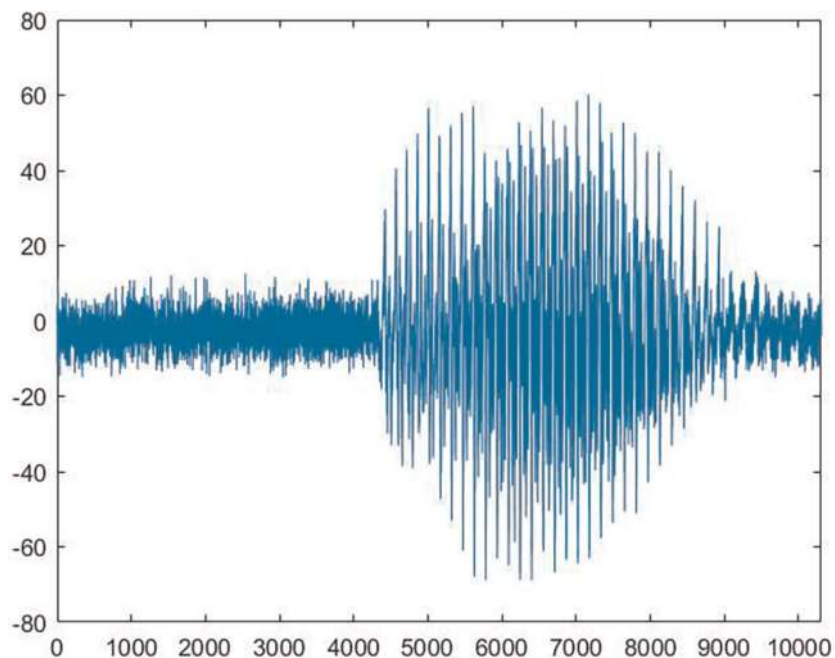
In **Figure 10** the source signal, is presented on **Figure 11**—noisy (additive white is added the noise with a mean 0 and variance 0.001 is Gaussian), in **Figure 12**—filtered. The soft SURE-threshold with Daubechies wavelet with amount of the zero moments  $L = 4$  was used (i.e., an order of FIR-HP and FIR-LP filter  $M = 8$ ).



**Figure 10.**  
Source signal for wavelet analysis threshold processing.



**Figure 11.**  
*A signal with an additive Gaussian noise for wavelet analysis threshold processing.*



**Figure 12.**  
*The signal cleaned using a wavelet analysis with threshold processing.*

## 5. The smoothing non-adaptive linear temporary filtering

The smoothing non-adaptive linear time filters are low pass filters [6].

In case of the FIR-LP filter with symmetric impulse response function  $h(-M), \dots, h(0), \dots, h(M)$ , non-adaptive linear time filtering represents convolution of non-adaptive impulse response function  $h(m)$  signal  $x(n)$  as

$$y(n) = \sum_{m=-M}^M h(m) x(n - m). \quad (51)$$

Let us give impulse response functions of the most widespread two-dimensional smoothing linear filters:

1. Impulse response function of the arithmetic mean filter

$$h(m) = \frac{1}{2M + 1}, m \in \overline{-M, M}. \quad (52)$$

2. Impulse response function of the normalized Gauss filter

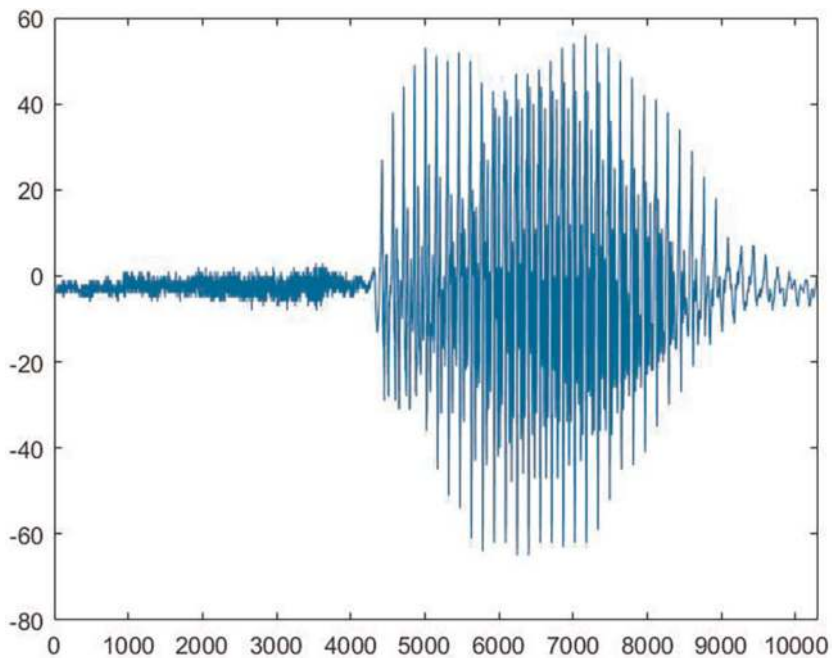
$$h(m) = \frac{\frac{1}{2\pi\sigma^2} \exp\left(-\frac{1}{2} \frac{m^2}{\sigma^2}\right)}{\sum_{l=-M}^M \frac{1}{2\pi\sigma^2} \exp\left(-\frac{1}{2} \frac{l^2}{\sigma^2}\right)}, m \in \overline{-M, M}. \quad (53)$$

3. Impulse response function of the normalized binomial filter

$$h(m) = \frac{C_{2M}^{M+m}}{\sum_{l=0}^{2M} C_{2M}^l}, C_n^m = \frac{n!}{m!(n - m)!}, m \in \overline{-M, M}. \quad (54)$$

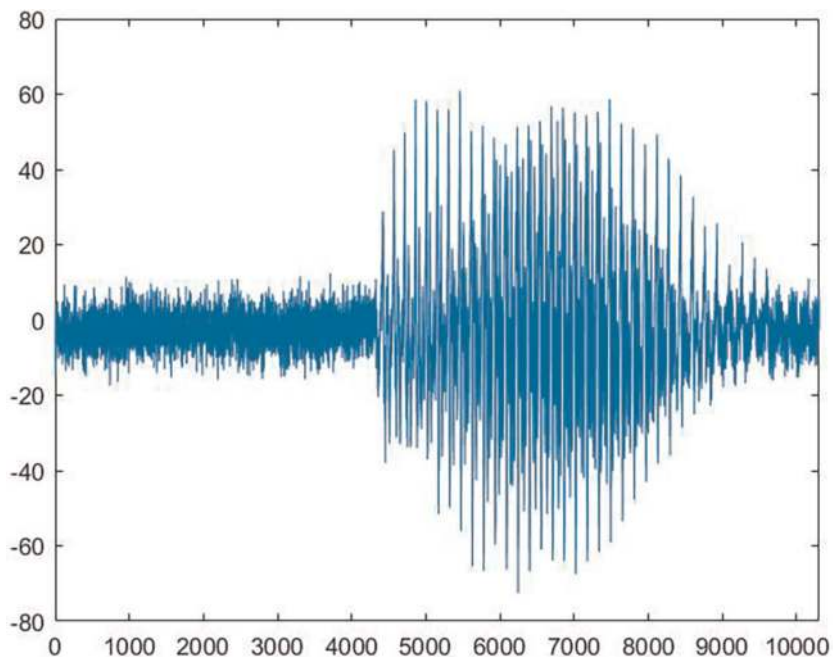
**Example**

In **Figure 13** the source signal, is presented on **Figure 14**—noisy (additive white is added the noise with a mean 0 and variance 0.001 is Gaussian), on **Figure 15**—filtered, wherein the arithmetic mean filter with  $M = 1$ .

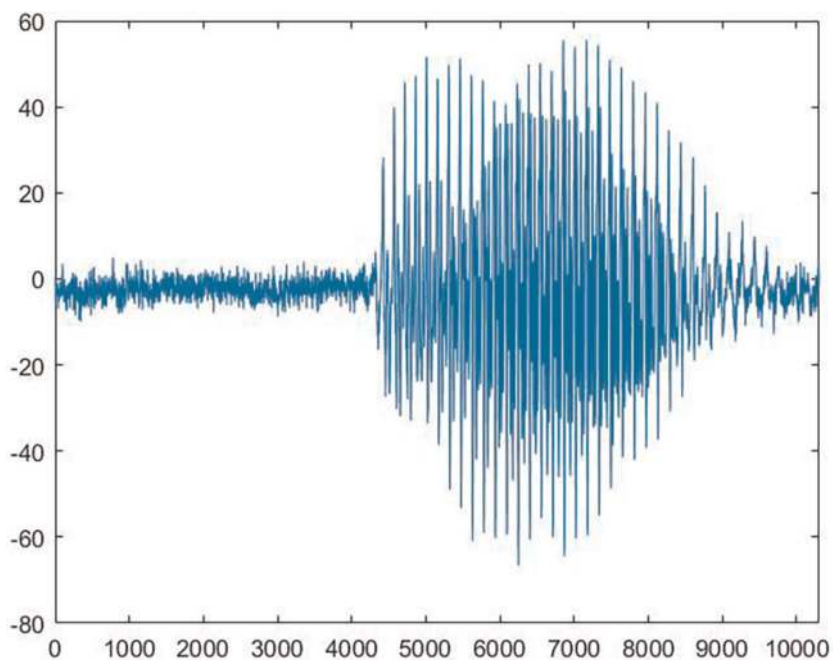


**Figure 13.** Source signal for smoothing non-adaptive linear temporary filtering.





**Figure 14.**  
*A signal with an additive Gaussian noise for smoothing non-adaptive linear temporary filtering.*



**Figure 15.**  
*The signal denoised by means of the arithmetic mean filter.*

## **6. The smoothing nonlinear filtering**

The smoothing nonlinear filters [6] are low-pass filters.

### 6.1 Geometric mean, harmonic mean, contraharmonic filters

1. Geometric mean filter

$$y(n) = \left( \prod_{m=-M}^M x(n-m) \right)^{\frac{1}{2M+1}}. \quad (55)$$

2. Harmonic mean filter

$$y(n) = \frac{2M+1}{\sum_{m=-M}^M \frac{1}{x(n-m)}}. \quad (56)$$

3. Contraharmonic filter

$$y(n) = \frac{\sum_{m=-M}^M x^{Q+1}(n-m)}{\sum_{m=-M}^M x^Q(n-m)}. \quad (57)$$

Geometric mean, harmonic mean, contraharmonic filters delete additive and multiplicative continuous and continuous impulse noises.

The harmonic mean filter in case of an impulse noise suppresses only white points.

The contraharmonic filter in case of an impulse noise at  $Q > 0$  suppresses only black points (at  $Q = -1$  receive the harmonic mean filter), and at  $Q < 0$  suppresses only white points. At  $Q = 0$  receive the arithmetic mean filter.

Therefore, for suppression of an impulse noise it is better to use  $\alpha$ -trimmed mean, median or rank, conservative and morphological filters.

### 6.2 $\alpha$ -trimmed mean filter

Algorithm  $\alpha$ -trimmed mean filtering applied to a signal  $x(n)$  size  $N$ , is as follows:

1. Create for each sample of a signal a vector from elements of its neighborhood  $U_n$  size  $2M+1$  as

$$a_n = (x(n-M), \dots, x(n+M)), n \in \overline{M, N-M+1}. \quad (58)$$

2. Sort for each sample of a signal element of its vector by increase

$$\tilde{a}_n = \text{sort}(a_n), n \in \overline{M, N-M+1}. \quad (59)$$

3. Execute  $\alpha$ -trimmed mean filtering of a signal in a form

$$y(n) = \frac{\sum_{m=1+\alpha/2}^{2M+1+\alpha/2} \tilde{a}_n(m)}{2M+1-\alpha}, n \in \overline{M, N-M+1}, \quad (60)$$

where  $\alpha$ —parameter, which multiple 2,  $0 \leq \alpha \leq 2M$ .

At  $\alpha = 0$  we receive the arithmetic mean filter, and at  $\alpha = 2M$  receive the median filter.  $\alpha$ -trimmed mean filter deletes additive and multiplicative continuous both continuous impulse noises and impulse noises.

### 6.3 Median and rank filters

Median filtering is defined in a form.

$$y(n) = \text{median}_{m \in U_n} \{x(m)\}, n \in \overline{M, N - M + 1}. \quad (61)$$

where  $U_n$ —neighborhood of sample  $n$  size  $2M + 1$ .

Median filtering is a special case of rank filtering at a rank  $r = M + 1$ . In case of rank filtering not the central sample, but sample which number corresponds to a rank undertakes  $r$ , and  $1 \leq r \leq 2M + 1$ .

Median and rank filters delete additive and multiplicative continuous both continuous impulse noises and impulse noises.

### 6.4 Midpoint filter

Algorithm of the midpoint filtering applied to a signal  $x(n)$  size  $N$ , is as follows:

1. Calculate for each sample of a signal the minimum and maximum value in its neighborhood in a form

$$\alpha(n) = \min_{m \in U_n} \{x(m)\}, \beta(n) = \max_{m \in U_n} \{x(m)\}, n \in \overline{M, N - M + 1}, \quad (62)$$

where  $U_n$ —neighborhood of sample  $n$  size  $2M + 1$ .

2. Signal midpoint filtering execute in a form

$$y(n) = \frac{1}{2} (\alpha(n) + \beta(n)), n \in \overline{M, N - M + 1}. \quad (63)$$

The Midpoint filter deletes additive and multiplicative continuous and continuous impulse noises.

### 6.5 Conservative filter algorithm

Conservative filtering algorithm applied to a signal  $x(n)$  size  $N$ , is as follows:

1. Calculate for each sample of a signal the minimum and maximum value in its neighborhood in a form

$$\alpha(n) = \min_{m \in U_n \setminus \{n\}} \{x(m)\}, \beta(n) = \max_{m \in U_n \setminus \{n\}} \{x(m)\}, n \in \overline{M, N - M + 1}. \quad (64)$$

where  $U_n$ —sample neighborhood  $n$  size  $2M + 1$ .

2. Signal conservative filtering execute in a form

$$y(n) = \begin{cases} x(n), & \alpha(n) < x(n) < \beta(n) \\ \alpha(n), & x(n) \leq \alpha(n) \\ \beta(n), & x(n) \geq \beta(n) \end{cases}, n \in \overline{M, N - M + 1}. \quad (65)$$

The conservative filter deletes additive and multiplicative continuous both continuous impulse noises and impulse noises.

### 6.6 Morphological filter

Morphological filtering is carried out by consecutive performing operations of open and close or close and open. At open, operations dilatation and an erosion are consistently executed, and at close—an erosion and dilatation.

Dilatation can be defined in a form.

$$z(n) = \max_{m \in U_n} \{x(m)\}, n \in \overline{M, N - M + 1}. \quad (66)$$

Erosion can be defined in a form.

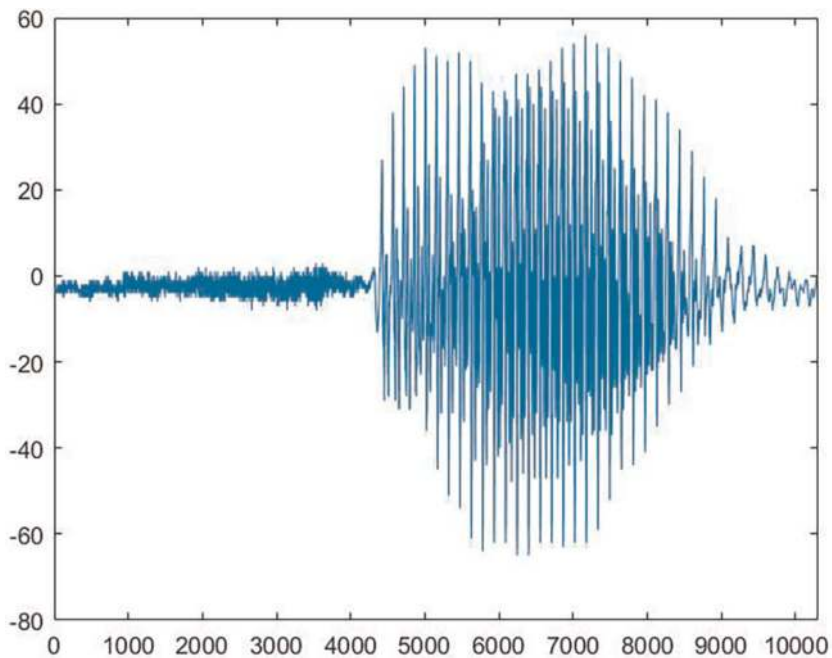
$$z(n) = \min_{m \in U_n} \{x(m)\}, n \in \overline{M, N - M + 1}, \quad (67)$$

where  $U_n$ —sample neighborhood  $n$ .

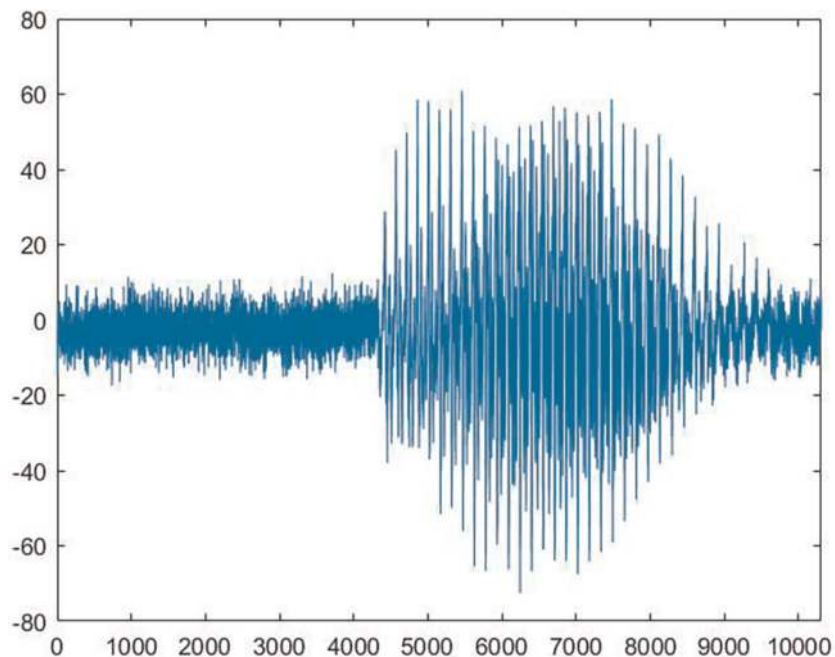
The morphological filter deletes impulse noises.

#### Example

In **Figure 16** the source signal, is presented on **Figure 17**—noisy (additive white is added the noise with an mean 0 and variance 0.001 is Gaussian), the signals denoised

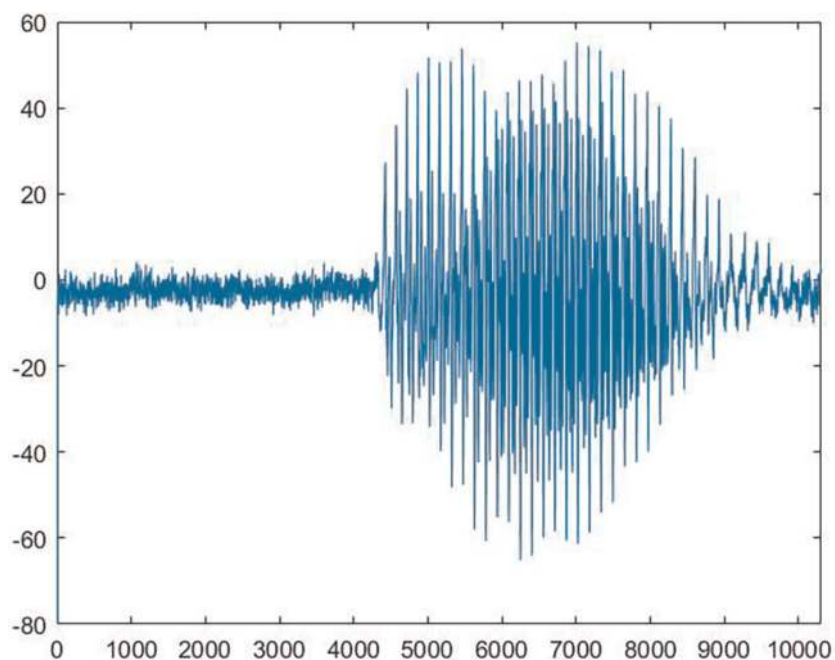


**Figure 16.** Source signal for smoothing nonlinear filtering of additive Gaussian noise.

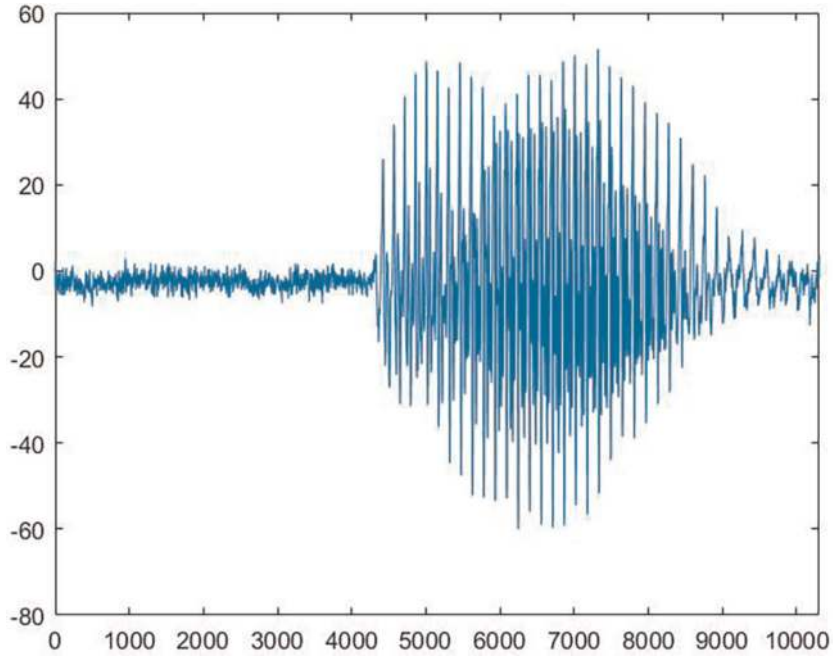


**Figure 17.**  
*A signal with an additive Gaussian noise for smoothing nonlinear filtering.*

by means of the geometric mean filter ( $M = 1$ ) (**Figure 18**),  $\alpha$ -trimmed mean filter ( $M = 2, \alpha = M = 2$ ) (**Figure 19**), median filter ( $M = 2$ ) (**Figure 20**), midpoint filter ( $M = 1$ ) (**Figure 21**), conservative filter ( $M = 1$ ) (**Figure 22**). In signal quality the syllable “sa” with a sampling rate of 22050 Hz, 8-bits, mono was selected.



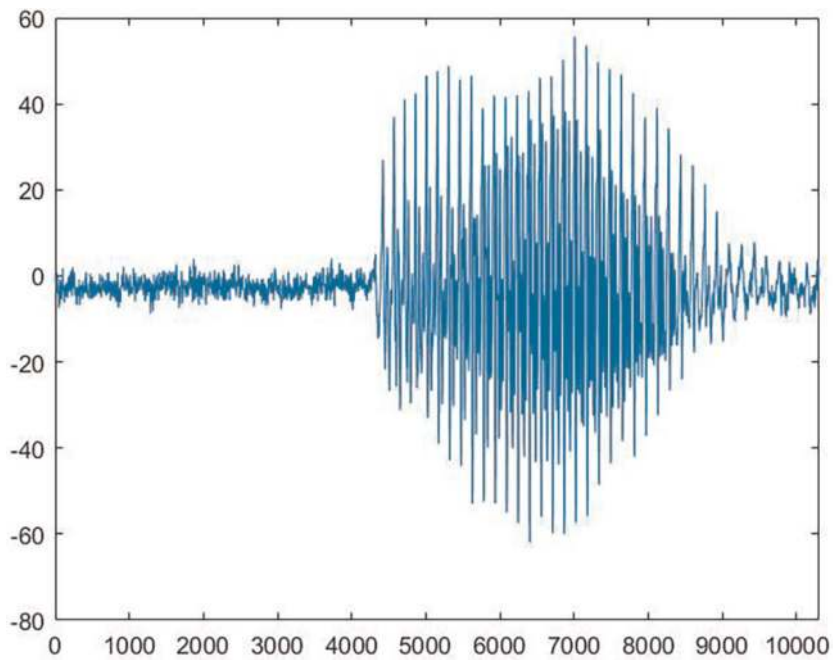
**Figure 18.**  
*The signal denoised by means of the geometric mean filter.*



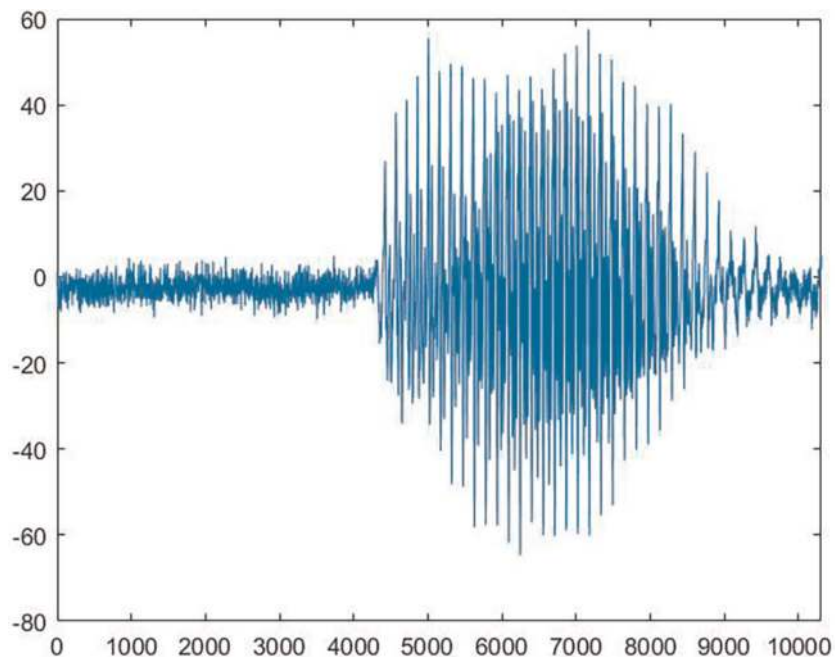
**Figure 19.**  
*The signal denoised by means of the  $\alpha$ -trimmed mean filter.*

**Example**

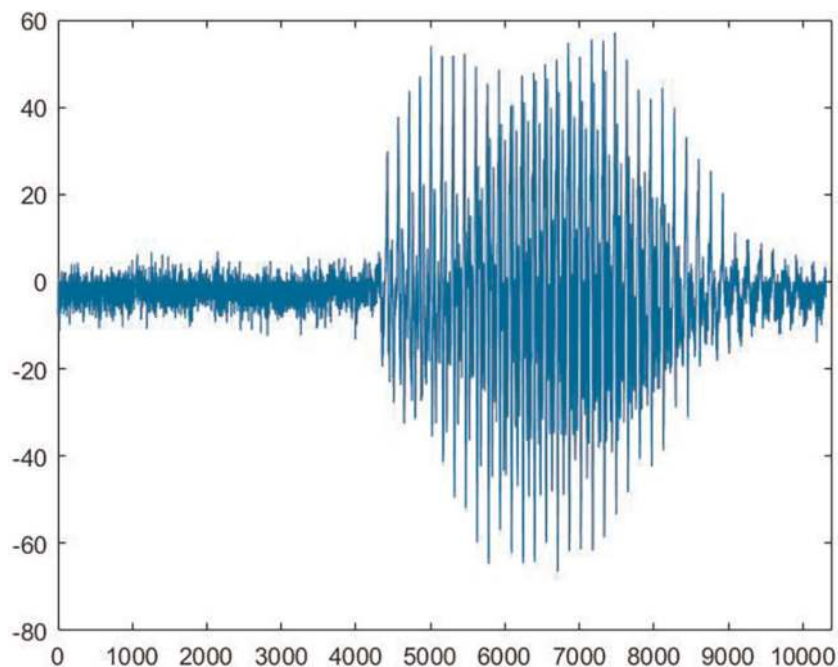
In **Figure 23** the source signal, on **Figure 24**—noisy (the impulse noise “salt and pepper” with a noisiness of 1% of sample of a signal is added), the signals denoised by means of the  $\alpha$ -trimmed mean filter ( $M = 2$ ,  $\alpha = M = 2$ ) (**Figure 25**), the median



**Figure 20.**  
*The signal denoised by means of the median filter.*

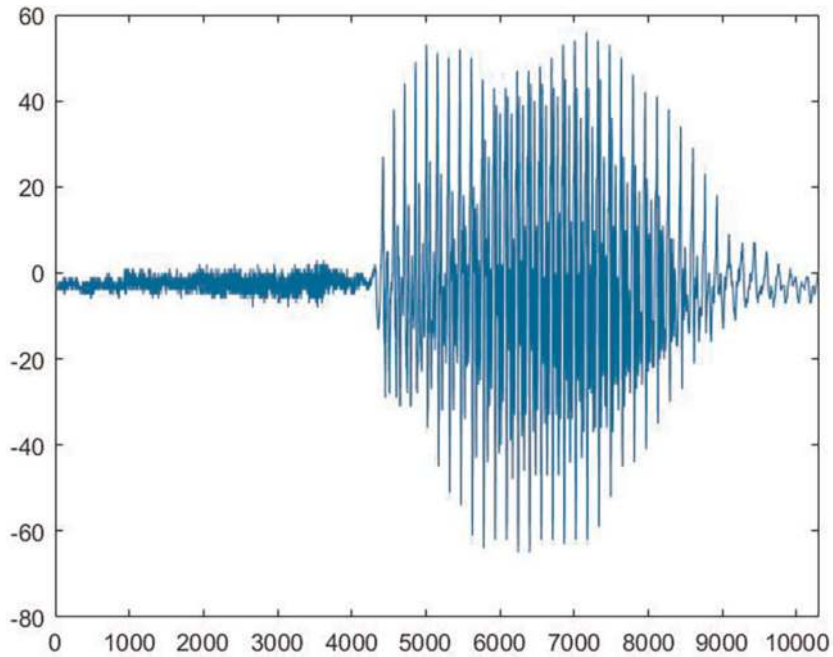


**Figure 21.**  
*The signal denoised by means of the midpoint filter.*

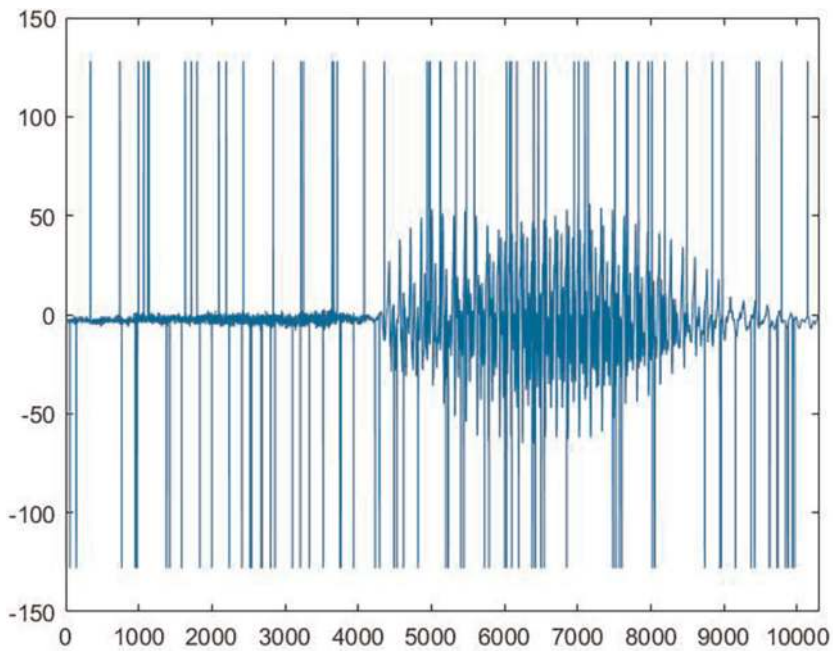


**Figure 22.**  
*The signal denoised by means of the conservative filter.*

filter ( $M = 2$ ) (**Figure 26**), the conservative filter ( $M = 1$ ) (**Figure 27**), the morphological filter (consistently executed by open and close with  $M = 3$ ) (**Figure 28**). In signal quality the syllable “sa” a sampling rate of 22050 Hz, 8-bits, mono was selected.



**Figure 23.**  
*Source signal for smoothing nonlinear filtering of impulse noise.*



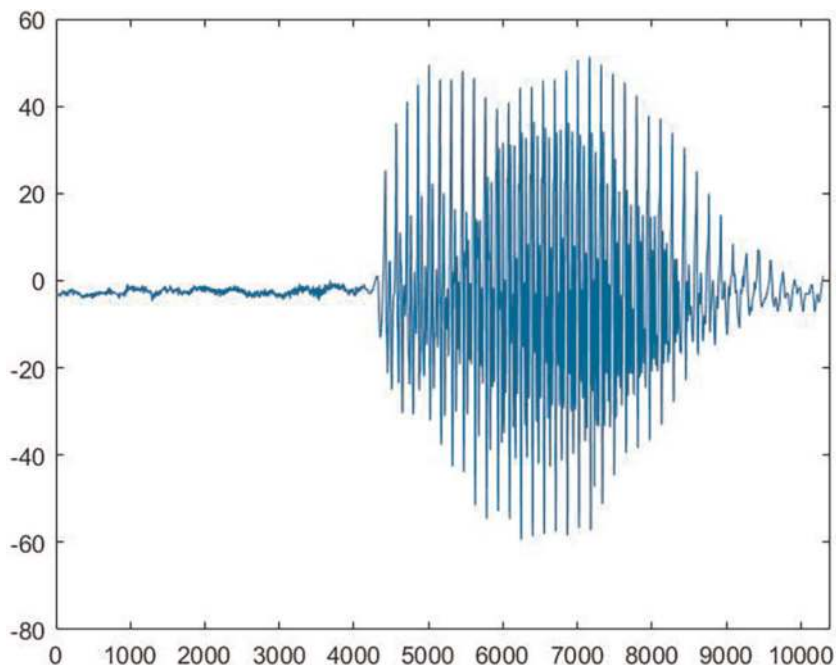
**Figure 24.**  
*A signal with an impulse noise "salt and pepper" for smoothing nonlinear filtering.*

## 7. Numerical research of denoising methods noise

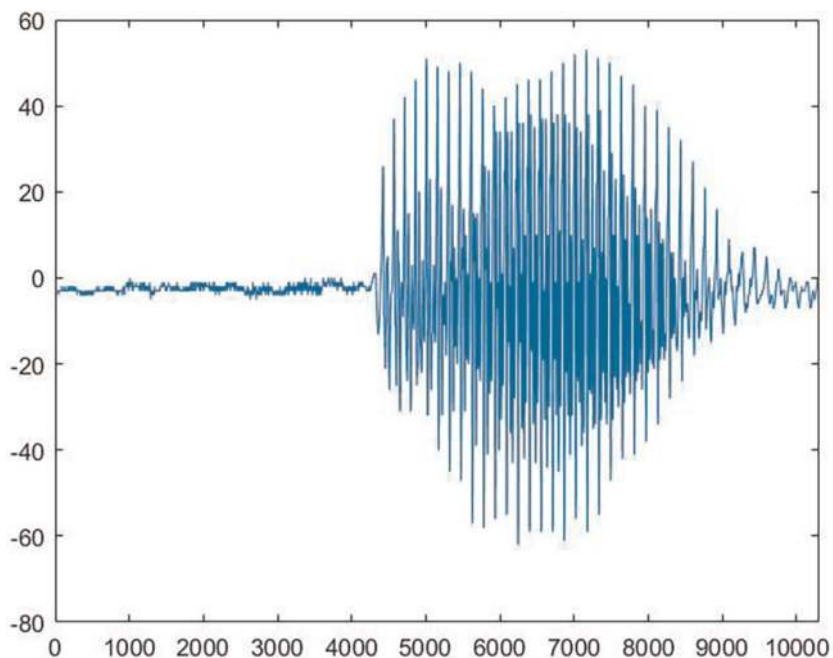
For the voice signals containing vocal sounds the sampling rate of 8 kHz and quantity of quantizing levels 256 was set.

Numerical research results of denoising methods on a basis a wavelet analysis with threshold processing in case of Daubechies wavelet about 8 with soft threshold



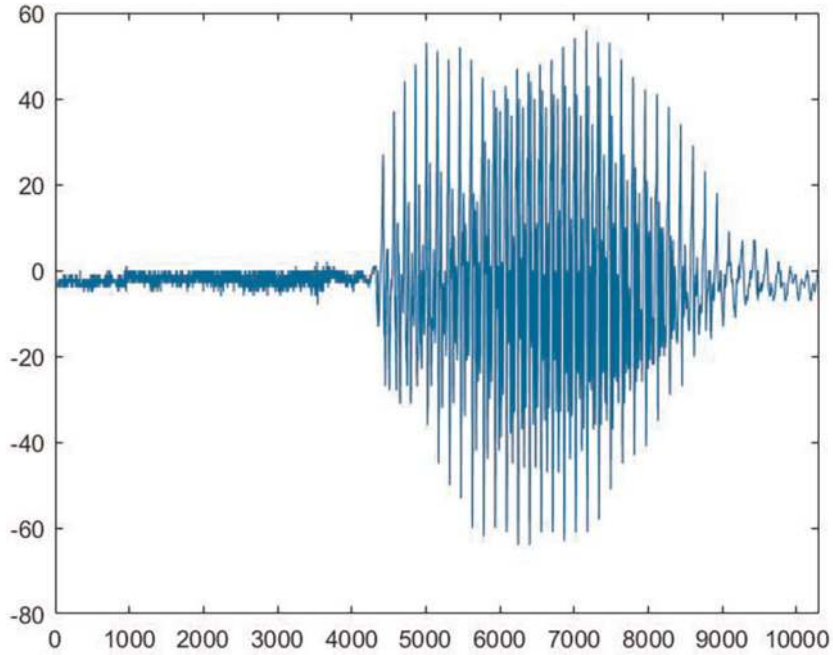


**Figure 25.**  
*The signal denoised by means of the  $\alpha$ -trimmed mean filter.*

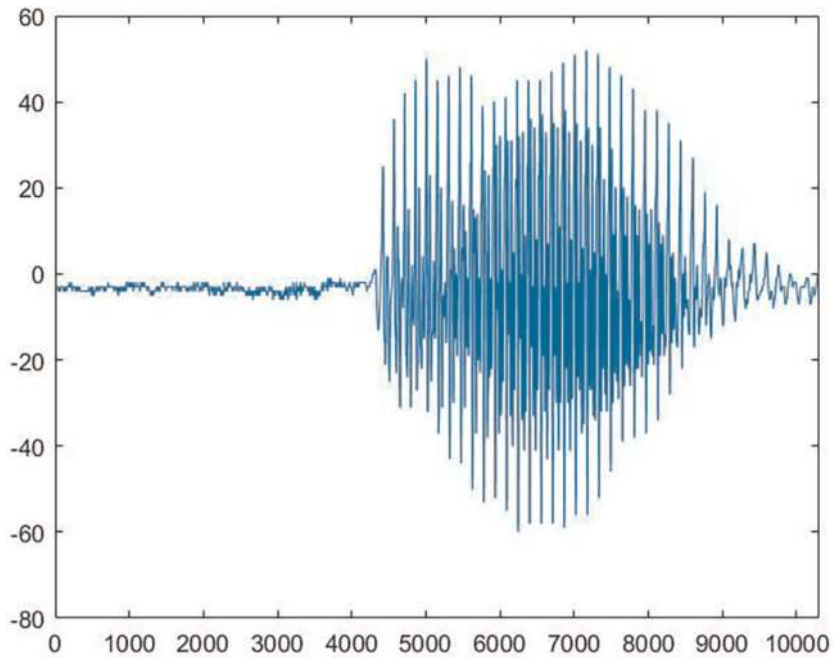


**Figure 26.**  
*The signal denoised by means of the median filter.*

processing with a SURE-threshold, the adaptive filter about 1, it is Gaussian the filter about 1 with parameter  $\sigma = 0.7$ , the arithmetic mean filter about 1, geometric mean filters about 1, harmonic mean filters about 1, contraharmonic filters about 1 with parameter  $Q = 1$ , median filter about 2,  $\alpha$ -trimmed mean filter of about 2 with parameter  $\alpha = 2$ , the midpoint filter about 1, conservative filters about 1 for voice



**Figure 27.**  
*The signal denoised by means of the conservative filter.*



**Figure 28.**  
*The signal denoised by means of the morphological filter.*

signals people from the TIMIT database which were noise an additive Gaussian noise with mean 0 and variance 0.001 (a signal-to-noise ratio about 11 dB) and multiplicative Gaussian noise with mean 1 and variance 0.07 (a signal-to-noise ratio about 23 dB), are presented to **Table 1**, where MSE—Mean Square Error.

Denoising method on a basis	MSE	
	Additive Gaussian noise	Multiplicative Gaussian noise
Wavelet analysis with threshold processing	11.2809	14.7860
Adaptive filter	9.9072	13.3914
Gaussian filter	14.3395	14.7662
Arithmetic mean filter	13.4738	14.0495
Geometric mean filter	15.5846	15.3252
Harmonic mean filter	20.4298	19.8623
Contraharmonic filter	13.3552	13.4845
Median filter	6.8697	6.5193
$\alpha$ -Trimmed mean filter	5.0843	4.9043
Midpoint filter	6.7667	6.4873
Conservative filter	9.0294	9.6437

**Table 1.** Results of a numerical research of denoising methods from additive Gaussian noise and multiplicative Gaussian noise.

The result is provided in **Table 1** shows that the smallest MSE is provided  $\alpha$ -trimmed mean filter.

## 8. Conclusion

For biometric identification are considered and in number investigated the following methods of noise suppression in a voice signal. The smoothing adaptive linear time filtering (the minimum root mean square error algorithm, the recursive least squares algorithm, the Kalman filtering algorithm, the Lee algorithm), the smoothing adaptive linear frequency filtering (the generalized method, the MLEE method, a wavelet analysis with threshold processing (universal threshold, SURE-threshold, minimax threshold, FDR-threshold, Bayesian threshold were used), the smoothing non-adaptive linear time filtering (the arithmetic mean filter, the normalized Gauss's filter, the normalized binomial filter), the smoothing nonlinear filtering (geometric mean filter, the harmonic mean filter, the contraharmonic filter, the  $\alpha$ -trimmed mean filter, the median filter, the rank filter, the midpoint filter, the conservative filter, the morphological filter). Numerical research results of denoising methods for voice signals people from the TIMIT database which were noise an additive Gaussian noise and multiplicative Gaussian noise were received. The  $\alpha$ -trimmed mean filter proved to be the most effective for both noise types.

## **Author details**

Eugene Fedorov<sup>1\*</sup>, Tetyana Utkina<sup>1</sup> and Tetyana Neskorocheva<sup>2</sup>


1 Cherkasy State Technological University, Cherkasy, Ukraine

2 Vasyl' Stus Donetsk National University, Vinnytsia, Ukraine

\*Address all correspondence to: fedorovee75@ukr.net

## **IntechOpen**

---

© 2022 The Author(s). Licensee IntechOpen. This chapter is distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/3.0>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited. 

## References

- [1] Diniz PSR. Adaptive Filtering: Algorithms and Practical Implementation. Berlin: Springer; 2020. 505 p. DOI: 10.1007/978-1-4614-4106-9
- [2] Lim JS. Two-Dimensional Signal and Image Processing. Englewood Cliffs, NJ: Prentice Hall; 1990. p. 694
- [3] Rabiner LR, Schafer RW. Theory and Applications of Digital Speech Processing. Upper Saddle River, NJ: Pearson Higher Education, Inc.; 2011. p. 1042
- [4] Yektaeian M, Amirfattahi R. Comparison of spectral subtraction methods used in noise suppression algorithms. In: Proceedings of 6th International Conference on Information, Communications and Signal Processing (ICICS 2007). 2007. pp. 1-4
- [5] Mallat S. A Wavelet Tour of Signal Processing: Sparse Way. 3rd ed. Burlington, MA: Academic Press; 2008. p. 832
- [6] Gonzalez R, Woods R. Digital Image Processing. Hoboken, NJ: Pearson Education, Inc.; 2018. p. 1306