

# Optimization of Short-Lag Spatial Coherence Imaging Method

Jakub DOMARADZKI<sup>(1)</sup>, Marcin LEWANDOWSKI<sup>(1)</sup>  
Norbert ŻOLEK<sup>(2)\*</sup>, Marcin LEWANDOWSKI<sup>(2)</sup>

<sup>(1)</sup> *The Faculty of Electronics and Information Technology  
Warsaw University of Technology  
Pl. Politechniki 1, 00-661 Warsaw, Poland*

<sup>(2)</sup> *Institute of Fundamental Technological Research  
Polish Academy of Sciences  
Pawińskiego 5B, 02-106 Warsaw, Poland*

\*Corresponding Author e-mail: nzolek@ippt.pan.pl

(received October 18, 2018; accepted March 11, 2019)

The computing performance optimization of the Short-Lag Spatial Coherence (SLSC) method applied to ultrasound data processing is presented. The method is based on the theory that signals from adjacent receivers are correlated, drawing on a simplified conclusion of the van Cittert-Zernike theorem. It has been proven that it can be successfully used in ultrasound data reconstruction with despeckling. Former works have shown that the SLSC method in its original form has two main drawbacks: time-consuming processing and low contrast in the area near the transceivers. In this study, we introduce a method that allows to overcome both of these drawbacks.

The presented approach removes the dependency on distance (the “lag” parameter value) between signals used to calculate correlations. The approach has been tested by comparing results obtained with the original SLSC algorithm on data acquired from tissue phantoms.

The modified method proposed here leads to constant complexity, thus execution time is independent of the lag parameter value, instead of the linear complexity. The presented approach increases computation speed over 10 times in comparison to the base SLSC algorithm for a typical lag parameter value. The approach also improves the output image quality in shallow areas and does not decrease quality in deeper areas.

**Keywords:** short lag spatial coherence; synthetic aperture; algorithm optimization; parallel processing.

## 1. Introduction

Speckle reduction in various imaging modalities has been extensively studied using various methods. It is extremely important in ultrasound imaging due to rough structures on the scale of the used wavelengths (HIREMATH *et al.*, 2013). It was shown that some reduction can be obtained by applying simple signal filtering (BENZARTI, AMIRI, 2012) or different versions of diffusion-based filtering (OVIREDDY, MUTHUSAMY, 2014). However, these attempts make it difficult to preserve edges in a filtered image (GUNGOR, KARAGOZ, 2015; VANITHAMANI, UMAMAHESWARI, 2014) and require additional pre/post-processing. Novel techniques use wavelet transform, which allows to remove the

speckles without losing much detail contained in an image (GUPTA *et al.*, 2004).

The development of synthetic aperture approaches in ultrasound research provides additional possibilities to deal with speckle reduction. One of the methods is based on the theory that the signals from adjacent receivers are correlated, which is a simplified conclusion of the van Cittert-Zernike theorem (MALLART, FINK, 1991). The theorem’s applicability to pulse-echo ultrasound imaging has been discussed and proved (LIU, WAAG, 1995) and the application of this theory was used in the introduction of the Short-Lag Spatial Coherence (SLSC) method (LEDIJU *et al.*, 2011; DAHL *et al.*, 2011). It has been shown that the method gives promising results in liver and fetal visualization despite

the fact that spatial coherence of the backscattered field is influenced by many factors (BAMBER *et al.*, 2002). The visualization quality of this method used along the Synthetic Aperture transmission-reception scheme has been analyzed in detail (BOTTENUS *et al.*, 2013) and showed that the quality of the SLSC method in its original form suffers from two main drawbacks: low contrast in the area near the transceivers and time-consuming processing.

The development of software-based ultrasound devices (Verasonics, 2019; S-sharp, 2019; WALCZAK *et al.*, 2013) and performance improvements of GPU processing enable a new approach to real-time implementation of the SLSC method. Unfortunately, the earlier proposed implementation (HYUN *et al.*, 2013) allowed authors to achieve processing with the speed not higher than 6 fps (frames per second). The subsequent attempt (HYUN *et al.*, 2014) focuses on the performance. It reduces covariance computations by shortening the covariation kernel to a single sample for I/Q data and performs uniformly sparse sampling on the receiving aperture. These techniques, when implemented on a GPU-based software beamformer, deliver real-time SLSC imaging system that generates images with frame rates ranging from 21 to 31 fps (HYUN *et al.*, 2015).

The main goal of this paper is to significantly improve the processing time of the SLSC not through data reduction, but through algorithm modification and efficient implementation.

## 2. Methods

### 2.1. SLSC algorithm and its modifications

Let us take into account the transducer with  $N$  adjacent transmit/receive elements used to insonify the medium and to receive returning echoes scattered in the medium. In the classical reconstruction attempt, for every receiving element, the values of the signals corresponding to the reconstructed pixel value of the image are summed up from values of the signals according to the delays in the transmission-reception scheme and the delays dependent on the position of the reconstructed image pixel (Delay and Sum technique) (MATRONE *et al.*, 2015).

In the SLSC technique, the value of each pixel of the resulting image is generated from spatial correlation and is calculated using pairs of signals ( $s_i, s_j$ ) from receivers located at varying positions  $i, j$  with the lag number between them  $m = j - i$ . To keep coherence in the signal for every pixel of the resulting image with coordinates  $(x, y)$ , the respective data sample from receiver  $k$  is calculated using delays from transmission reception schema, i.e. using Plane Wave insonification:

$$d_k(x, y) = \left( y + \sqrt{(x - r_k)^2 + y^2} \right) f_s / c_0, \quad (1)$$

where  $f_s$  stands for sampling frequency and  $c_0$  stands for signal propagation speed.

It is schematically shown in Fig. 1. The correlation values  $\widehat{R}(m)$  (for each pixel of the resulting image corresponding to sample number  $d_i(x, y)$  in the center of the correlation kernel of the length  $2\Delta c$ ) are averaged from all pairs at a given spacing (2) (LEDIJU *et al.*, 2011):

$$\widehat{R}_m(x, y) = \frac{1}{N - m} \sum_{i=1}^{N-m} \frac{a^*}{\sqrt{b^*}} \quad (2)$$

where

$$a^* = \sum_{n=-\Delta c}^{\Delta c} s_i(d_i(x, y) + n) s_{i+m}(d_{i+m}(x, y) + n),$$

$$b^* = \sum_{n=-\Delta c}^{\Delta c} s_i^2(d_i(x, y) + n) \sum_{n=-\Delta c}^{\Delta c} s_{i+m}^2(d_{i+m}(x, y) + n),$$

and  $m$  is the lag number,  $N$  is the number of receivers,  $n$  are sample numbers,  $s_i(n)$  is the zero-mean signal from the  $i$ -th receiver,  $2\Delta c$  is the length in samples of correlation kernel.

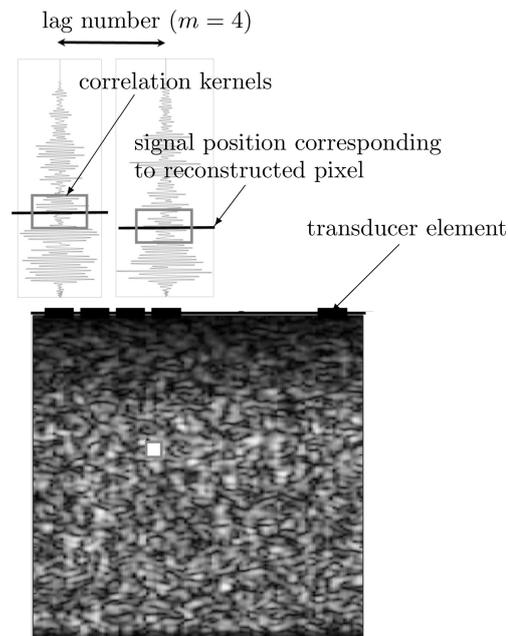


Fig. 1. Scheme of the transducer with imaging area and signals received by two elements separated using lag value equal to 4 and used to calculate correlation.

Finally, each pixel value in the resulting image is described as the sum of the spatial coherence function over the first  $M$  lags (3):

$$R_{sl} = \sum_{m=1}^M \widehat{R}(m). \quad (3)$$

The optimization presented here relies on removing the averaging factor  $\left(\frac{1}{N-m}\right)$  from the correlation calculation (2). Without loss of generality, and for further

simplicity of the following equations, it is assumed here that the signal sample positions are the same in signals  $i$  and  $i + m$  and the center of the correlation kernel is at sample 0. As a result, the original SLSC Eq. (3) can be rewritten in a new form:

$$R_{sl} = \sum_{i=1}^N \sum_{j=i+1}^{\min(i+M,N)} \frac{\sum_{n=-\Delta c}^{\Delta c} s_i(n)s_j(n)}{\sqrt{\sum_{n=-\Delta c}^{\Delta c} s_i^2(n) \sum_{n=-\Delta c}^{\Delta c} s_j^2(n)}}. \quad (4)$$

Equation (4) changes the perception of the reconstruction process. Before, the main focus was on the calculation of individual lags and collecting the results. Now, it resembles the Delay & Sum class method (MATRONE *et al.*, 2015), where we iterate over subsequent transceivers. A further transformation of Eq. (4) leads to:

$$R_{sl} = \sum_{i=1}^N \frac{\sum_{n=n_1}^{n_2} s_i(n)L(i, n, N, M)}{\sqrt{\sum_{n=n_1}^{n_2} s_i^2(n)}}, \quad (5)$$

where  $L(i, n, N, M)$  is defined by Eq. (6):

$$L(i, n, N, M) = \sum_{j=i+1}^{\min(i+M,N)} \frac{s_j(n)}{\sqrt{\sum_{n=n_1}^{n_2} s_j^2(n)}}, \quad (6)$$

and can be calculated recursively starting from the first transceiver ( $i = 1$ ):

$$L(i + 1, n, N, M) = L(i, n, N, M) - \frac{s_i(n)}{\sqrt{\sum_{n=n_1}^{n_2} s_i^2(n)}} + \frac{s_{i+1+M}(n)}{\sqrt{\sum_{n=n_1}^{n_2} s_{i+1+M}^2(n)}}. \quad (7)$$

It should be noted that Eq. (7) is the general version and is only valid for transceivers from 1 to  $N - M - 1$ . For the rest of the transceivers, the last term of Eq. (7) should be omitted.

Forms (5) and (7) are the final modification of the original SLSC method for speeding up the computation process. Instead of computing each lag separately, the method presented utilizes results from  $i$ -lag value at  $(i + i)$ -lag value. The MATLAB (Mathworks, USA) scripts illustrating the algorithm are available<sup>1</sup>.

### 2.2. Calculation speed

The presented method of removing coefficient  $1/(N - m)$  is adequate both for implementing on the CPUs<sup>2</sup> as well as on GPUs<sup>3</sup>. However, GPUs allow to

parallelize computation and get the results in a shorter time. In this case, standard NVidia’s CUDA<sup>4</sup> parallelization was used and a kernel for each pixel of output image calculation was created. When using the GPU to further improve performance, the values of  $L$  parameter should be stored in fast access Shared Memory to efficiently manage cached data.

Four different implementations of the SLSC algorithm were tested. All versions were implemented with the use of NVidia’s CUDA technology: The *Original* – original SLSC implementation rewritten to use SIMD<sup>5</sup> architectures, *Original with Shared* – original SLSC implemented with CUDA’s and rewritten to use SIMD architectures with CUDA’s Shared Memory<sup>6</sup> technology, *Speed Optimized* – as presented here, SLSC optimization with only speed optimization, and finally *SLSC-O* – as presented here, SLSC utilizing both depth-dependent lag number changing and speed optimization. It should be noted that for each signal, the average value from a signal was subtracted. The duration of this operation is included in algorithm timings.

### 2.3. Image quality verification

#### 2.3.1. Synthetic Aperture Scheme

In the MSTA (multi-element synthetic transmit aperture) algorithm (TROTS *et al.*, 2010), used here, echoes are collected in different matrices. After each plane wave insonification with the use of multiple elements (ex. 16 of 128), a single transmission aperture is shifted by one transmitter. The output image is created from complementary images each with the width of one line near the axis of symmetry of the transmission aperture. This is done to ensure that the distance from a reconstructed pixel and the transmission elements is accurately defined and not an approximation as in (TROTS *et al.*, 2010). This attempt reduces the spatial range of the transmit grating lobes and allows to focus on image quality of the presented method without external disturbances.

#### 2.3.2. Image quality quantification

The differences between the original methods of reconstruction and those proposed here may influence the resulting image quality. The quality of images reconstructed with the SLSC technique depends on two parameters: the number of samples taken into account in the correlation calculation per single reconstructed image point, and the number of lags used for spatial coherence calculation. The first parameter is commonly set to a value equal to one wavelength (POUREBRAHIMI *et al.*, 2013), while the second one ranges originally from 1 to 30% of the transmit aperture (LEDIJU *et al.*,

<sup>1</sup>SLSC-O Optimized Short Lag Spatial Coherence in Ultrasound Visualization, <https://github.com/usgold/SLSC>.

<sup>2</sup>Central Processing Unit.

<sup>3</sup>Graphics Processing Units.

<sup>4</sup>Compute Unified Device Architecture.

<sup>5</sup>Single Instruction, Multiple Data.

<sup>6</sup>See: <https://docs.nvidia.com/cuda/cuda-c-programming-guide/index.html#variable-memory-space-specifiers>.

2011). Here, the analysed lag number is extended outside this range. Lag numbers equal to 21, 41, and 61 are chosen. We choose large lag numbers in order to compare results from the SLSC and SLSC-O methods and analyze how removing the  $1/(N-m)$  coefficient in Eq. (2) influences output images. For lower lag numbers (like 21), we observe no difference in the quality of the results.

In order to test how modifications applied to the original algorithm affect the resulting images, we considered the following image quality factors (BOTTENUS *et al.*, 2013; CHEN *et al.*, 2003; WANG *et al.*, 2004): Contrast (dB scale), Signal to Noise Ratio (SNR), Contrast to Noise Ratio (CNR), Peak Signal to Noise Ratio (PSNR), and Structural Similarity (SSIM).

The comparison of different versions of the SLSC method requires a peculiar normalization of colormaps of the resulting images. This is not part of the *SLSC-O* method, but rather the post-processing of resulting images of the original SLSC method. It aids in viewing and visually comparing images. It is explained as follows. Taking into account the absolute output values, the *Original* and *Original with Shared* methods give different results than the *Speed Optimized* and *SLSC-O* methods as can be seen in Eq. (6). The differences in the algorithms analyzed here influence the absolute resulting values. The coefficient  $1/(N-m)$  in Eq. (2) is removed from Eq. (4), which leads to a different way of weighing the data, particularly for higher values of lags. This results in the first two versions of the SLSC method producing some artifacts in the output data.

When trying to typically normalize resulting data, the background of images (absolute values around zero) appears to be different. The comparison is presented in Fig. 2.

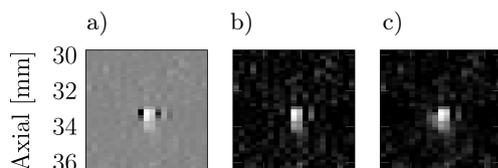


Fig. 2. Comparison of phantom W output images considering the necessity of peculiar normalization at the *Original* method. *SLSC-O* does not need peculiar normalization (data were collected using lag value of 61 in order to emphasize the effect): a) original, b) original peculiarly normalized, c) *SLSC-O*.

Artifacts (black areas around the image of the fiber) can be seen on both sides in the lateral direction of fiber (white spot) in Fig. 2a. In Fig. 2c, this effect is not present. While fiber is white in both pictures, the background of the image from the *Original* method is brighter. This situation occurs because the absolute values near the fiber are below zero. This issue gains greater influence as the value of the lag number

increases, if negative correlation in the *Original* and the *Original with Shared* occurs.

In order to overcome this issue and compare the methods properly, we introduced peculiar normalization to the resulting data. After dividing by the maximum value, we trim values below the maximum of minimum value of the images compared. After trimming, the normalization is repeated to achieve the range from 0 to 1.

#### 2.4. Improvement of quality

In order to improve contrast in the area near the transducer, the variety of the lag values dependent on the reconstructed depth of the image was analyzed. Studying individual outcomes of consecutive lags shows that, there is a noticeable gradual decrease in information about the upper part of the image (closer to transceiver) as the number of lags increases. Regarding this dependence, the prediction about lag numbers as a function of depth can be calculated with the use of the formula (8):

$$\widehat{M}(y) = \begin{cases} \forall y \leq H & 1 + \frac{y}{H} \cdot (M - 1), \\ \forall y > H & M, \end{cases} \quad (8)$$

where  $\widehat{M}(y)$  is the depth-dependent lag number,  $y$  is the current image depth,  $H$  is the width of the transmit aperture.

Equation (8) allows for applying a variable number of lags in the upper part of the image only. The value of the depth ( $H$ ) of threshold is related directly to the range of side lobes and grating lobes affecting the resulting image and should not exceed the aperture size of the transducer. Equation (8) is derived from the basic observation that close to the transducer, signals for larger lag values corresponding to distant transducer elements are independent.

The variable number of lags used for reconstruction influences the intensity at distinct depths of the image. To prevent this disparity and to equalize the outcomes at various depths, an additional weighing to individual outcomes is added. As a result, Eq. (3) takes a new form:

$$R_{sl}(y) = \sum_{m=1}^{\widehat{M}(y)} \widehat{R}(m) \cdot \frac{M}{\widehat{M}(y)}. \quad (9)$$

#### 2.5. Measurement and processing setup

The ultrasound data used for analysis of the presented method was collected using the research ultrasound acquisition system Verasonics V-1-128 (Verasonics, USA) with a 128-element linear array probe (Philips L7-4). The element pitch was equal to  $D = 0.3048$  mm, element width equaled 0.2798 mm and elevation height was 4 mm. The medium was insonified using a plane wave with 4 cycles of 5 MHz sine wave

with a triangular envelope. The RF echoes were sampled at 36 MHz. The phantoms used as a scattering medium in the ultrasound measurements consisted of cysts: model 571 (phantom C) and fibers: model 525 (phantom W), both from Dansk Fantom Service (Denmark). The C phantom contained anechoic cysts with diameters 8, 4, and 2 mm. The W phantom contained 0.1 mm diameter nylon fibers. The speed of sound assumed in all measurements was fixed at 1540 m/s.

The data processing and reconstruction algorithm was tested on a PC class computer under Microsoft Windows 7 64-bit operating system. The computer was equipped with an Intel Core 2 Duo E6400 (2.13 GHz) processor, 4 GB RAM and Graphics Processing Unit Nvidia QUADRO K5000 card (1536 CUDA cores<sup>7</sup>).

### 3. Results and discussion

#### 3.1. Performance of reconstruction

The implementation loads to the Shared Memory the data of the surrounding of the sample determined by the pixel delay for the selected receiver. This is followed by iteration over the subsequent receivers, thereby forming a larger lag in tandem with the receiver.

Moreover, the computational time of the new method is nearly independent of the number of chosen lags. The main limitation of performance is global memory access. Assuming that computation and *Shared Memory* access is negligibly cheap, and thus counting only global memory access, we get:

- *Original*:  

$$2\Delta c \cdot M \cdot (2 \cdot N - 1 - M),$$
- *Original with Shared*:  

$$2\Delta c \cdot M \cdot (N - M + (N - 1)/M - 1 + (M + 1)/2),$$
- *SLSC-O*:  

$$2 \cdot 2\Delta c \cdot (N - 1).$$

These formulas do not include the depth-dependent lag parameter. The correlation operations' count can be calculated by dividing each formula by  $2\Delta c$ .

For the performance test, an image of  $128 \times 2048$  pixels from echo signals of the size 4480 was reconstructed. Figure 3 shows the execution time of each of the two versions of the SLSC algorithm implemented on a CUDA platform with the correlation kernel size set to the number of samples corresponding to the typical length of one wavelength (LEDIJU *et al.*, 2011). The performance results match theoretical considerations. The *Original* approach represents the base algorithm without our modifications. The method *SLSC-O* utilizes both optimizations described in this paper. The execution time, which is stable for different lag values, is equal to 65 ms, resulting in 15 fps. For lag value

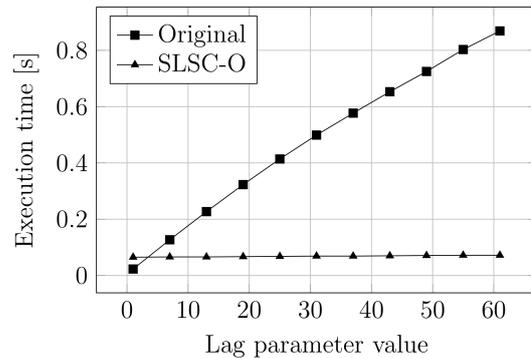


Fig. 3. Comparison of execution time of two versions of the SLSC algorithm. Tests were performed on the GPU. However, the speed-up of *SLSC-O* over *Original* should be similar for the CPU. The tests were performed for  $128 \times 2048$  resulting output image size.

equal to 61, the presented approach is about 12 times faster than the base algorithm, while for lag value equal to 31, this becomes over 7 times faster. The proposed techniques appear to be most effective specifically for large lag numbers. For the typical lag parameter value (10–30), we still observe significant speed-up. Only for lag values lower than about 4, the *SLSC-O* performs slower than *Original*. Both modifications of the original algorithm allow to achieve better results. The increasing differences in performance are more visible for higher values of lag parameters. Moreover, the computational time of the new method is almost independent of the number of chosen lags and leads to constant complexity execution time.

Higher frame rate can be obtained by decreasing the size of the reconstructed image. For example, when reconstructing  $128 \times 512$  pixels image, which still gives similar accuracy and image quality, the execution time of the algorithm is 39 ms. This timing results in over 25 fps. It should be noted that at lag value not higher than 4, the *SLSC-O* method has worse performance, but in practical applications such values are not used.

The SLSC images of the reconstructed phantoms used in this study are shown in Figs 4 and 5. A rectangular region ranging about  $40 \times 84$  mm has been reconstructed for both the W and C phantoms. The regions of interest chosen for SNR, CNR, contrast and

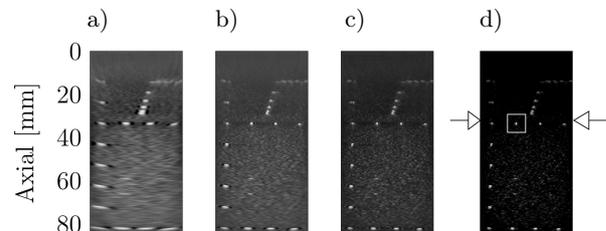


Fig. 4. Comparison of reconstruction of images of phantom W shows difference when manipulating the lag parameter number (the *Original* method; 25 dB scale): a) Lag 11, b) Lag 31, c) Lag 51, d) Lag 71.

<sup>7</sup>Quadro K5000 Specifications, <http://www.nvidia.com/object/quadro-k5000.html>

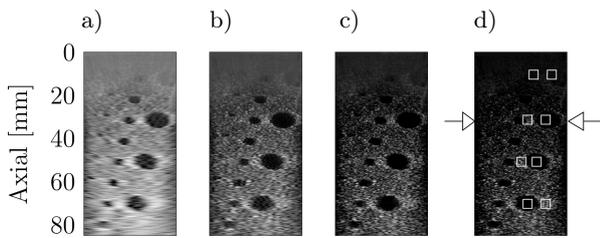


Fig. 5. Comparison of reconstruction of images of phantom C shows difference when manipulating the lag parameter number (the *Original* method; 25 dB scale): a) Lag 11, b) Lag 31, c) Lag 51, d) Lag 71.

PSNR calculations are presented as white rectangles. The selected regions for phantom C were located inside and outside of cysts and collected in pairs at the same depth. The selected region for phantom W contains one of the fibers. SSIM parameter calculations (WANG *et al.*, 2003) were applied to the whole resulting images.

Additionally, the section lines of the resulting image at depth of 32 mm (phantom C) and 33 mm (phantom W) were analyzed. The chosen section goes through the middle of the two cysts (phantom C, Fig. 5d) and four fibers (phantom W, Fig. 4d). In both figures, the position of the section line is marked with the arrows on both sides of each image.

### 3.2. Quality of the resulting images

Image quality can be compared in Figs 4 and 5, where the reconstruction of both phantoms was made using different lag values.

It can be observed that the *Original* method results in larger negative correlation values in the area near the fiber in Fig. 6.

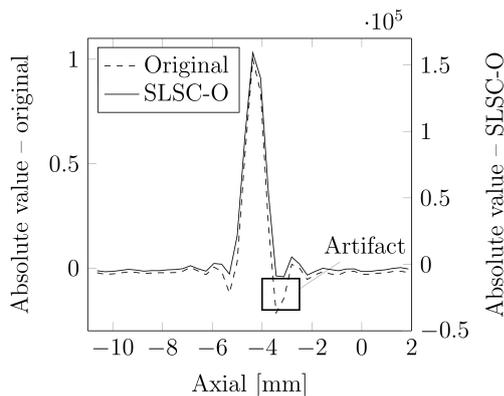


Fig. 6. 33 mm depth line of phantom W result data for *Original* and *SLSC-O* versions of SLSC method. Both Y-axes were adjusted to show the marked artifact which is not desirable. All data were collected using the lag value of 31.

The comparison of the reconstructed section line of phantom W at the depth of 33 mm using lag values of 41 and 81, respectively, presented in linear, normalized

scale is shown in Fig. 7. It can be seen that absolute values of reconstructed images are similar. Peaks of data representing the fibers have almost the same shape. The baseline is flat in each case. The main differences between the *Original* and *SLSC-O* methods are the areas near the fibers. Data obtained using the *Original* method have higher variability than the *SLSC-O*, which stems from the fact that for higher lag values, correlation is less stable. The *SLSC-O* method for higher lag values results in the preservation of more detail. The reason why we observe such artifact are negative values caused by different weights, especially when using a higher lag value parameter. Using *Original*, when the coherence is strong negative for higher lag numbers (for example, when there is a small strong scatterer surrounded by no scatterers), the output values of coherence calculation can be also strong negative. However, when using *SLSC-O*, strong negative coherence for higher lag numbers is reduced, while for lower lag numbers (where coherence is typically still positive) calculations are much the same. It should be noted that when no negative or positive coherence occurs, both *Original* and *SLSC-O* method give the same values (baseline). It can be observed in Figs 6 and 7.

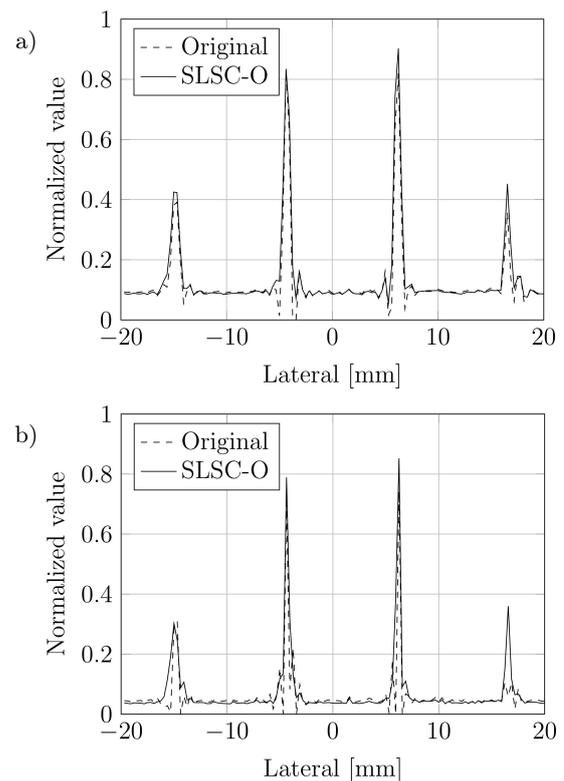


Fig. 7. Result values at 33 mm depth line for phantom W (values are normalized and in linear scale): a) Lag 41, b) Lag 81.

In Fig. 7, one can see all four fibers from *SLSC-O*, while the *Original* did not reconstruct the fiber at 16–17 mm axial position. Note that we do not observe such differences for smaller lag values.

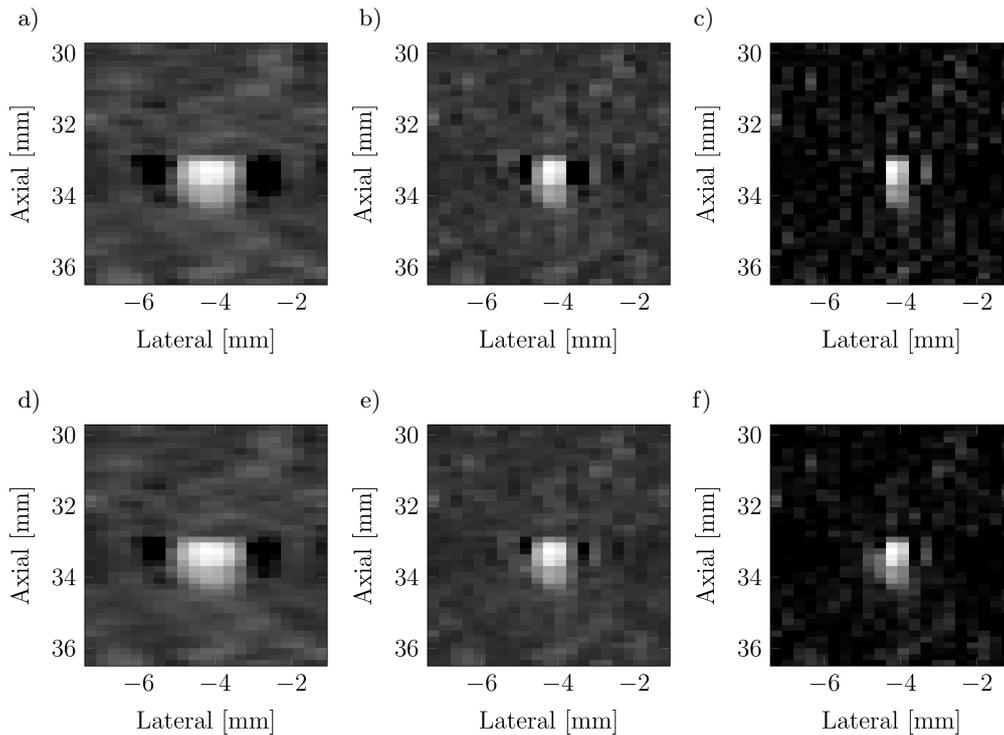


Fig. 8. Comparison of reconstructed images of chosen region of phantom W (25 dB scale): a) Original Lag 21, b) Original Lag 41, c) Original Lag 61, d) SLSC-O Lag 21, e) SLSC-O Lag 41, f) SLSC-O Lag 61.

Images of the phantom W obtained from the *Original* and *SLSC-O* algorithms are shown in Fig. 8. The first row shows the reconstruction with the *Original* method, while the lower row used the *SLSC-O* algorithm. The lag values used in the presented cases were set to: 21, 41, and 61. The resulting images show a magnification of one of the fibers located at 33 mm depth. All images in Fig. 8 are displayed in a 25 dB logarithmic scale (calculated for the whole imaging area). A strong similarity can be observed between images obtained using both the *SLSC-O* and *Original* methods. The lag value of 21 allows us to obtain images with no visual difference. For higher lag values, that is 41 and 61, the differences appear in areas close to the sides of the fiber. Other areas of the resulting images are comparable for all lag values.

Figure 9 shows the comparison of the reconstructed line of the phantom C at the depth of 32 mm using lag number values equal to 21 and 41, respectively. Both the *Original* and *SLSC-O* methods' results are shown in each figure. The data are presented in linear, normalized scale. The main difference can be seen in the data representing cysts (from -2 to 2 and from 8 to 17 mm of axial position). Inside these cysts, the *Original* method gave higher data values. However, the overall shape of the plot of data is similar.

Taking into account all points at the 32 mm depth line, the average difference between absolute values of the result is 0.73% and the maximum is 2.02%.

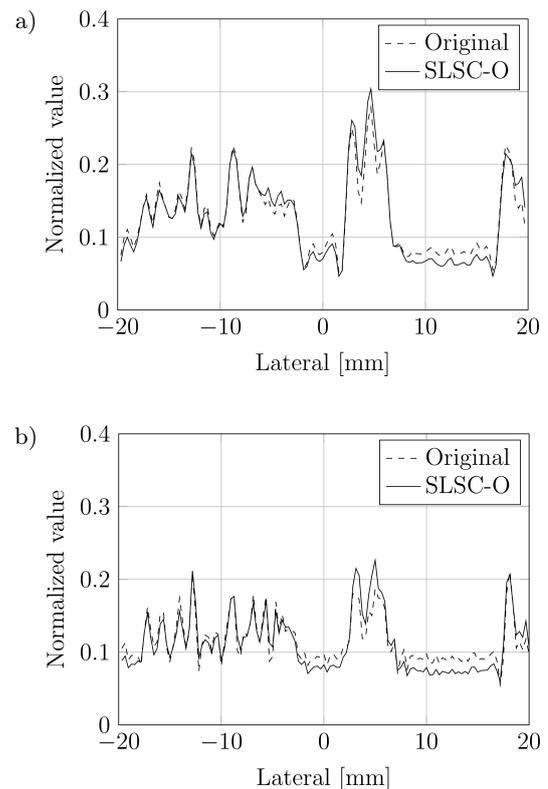


Fig. 9. Result values at 32 mm depth line for phantom C (values are normalized and in linear scale): a) Lag 21, b) Lag 41.

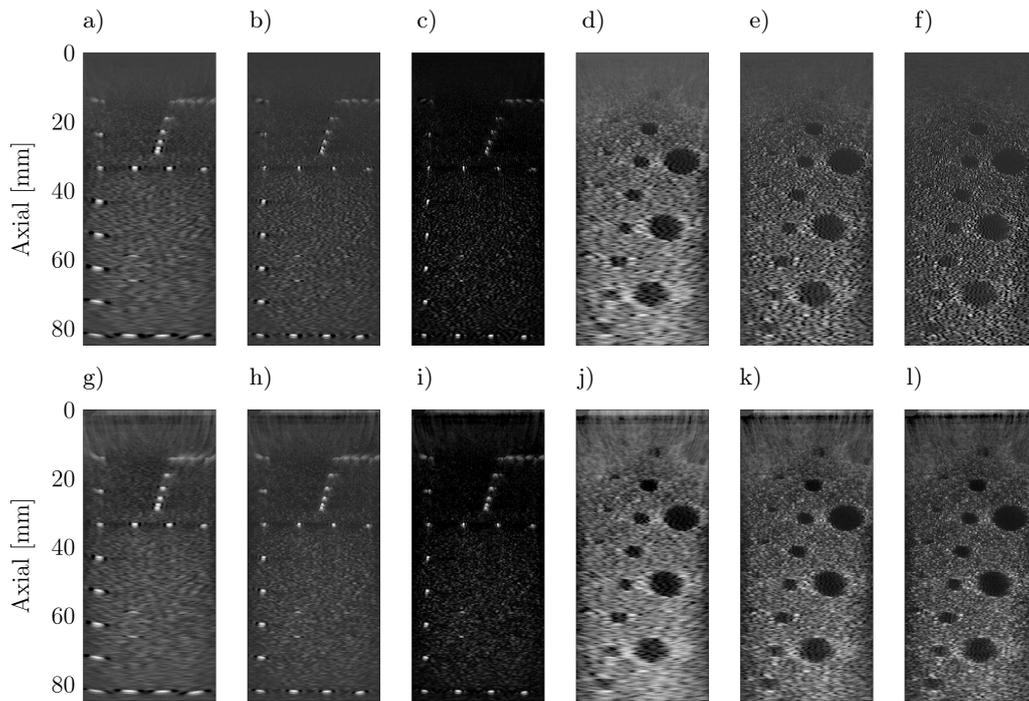


Fig. 10. Comparison of *Original* (top row) with *SLSC-O* (bottom row) for both phantoms C and W for lag numbers of 21, 41, and 61 (25 dB scale): a) Lag 21, b) Lag 41, c) Lag 61, d) Lag 21, e) Lag 41, f) Lag 61, g) Lag 21, h) Lag 41, i) Lag 61, j) Lag 21, k) Lag 41, l) Lag 61.

The comparison of the whole resulting images of phantoms C and W are presented in Fig. 10. The first row shows reconstruction using the *Original* method, while the second row uses the *SLSC-O* method for lags set to: 21, 41, and 61. All images are shown in a 25 dB logarithmic scale.

Figure 10 shows the comparison of the *Original* with the *SLSC-O* method for both phantoms C and W

and different lag numbers. The images differ fundamentally only in the upper part where the lag parameter was manipulated. For phantom C, the shallow cysts at the depth of 16 mm can be observed with the *SLSC-O* method, mainly at higher lag values.

Several image quality parameters have been calculated: Contrast, CNR, SNR, and PSNR for selected pairs of regions and are reported in Table 1. The

Table 1. Comparison of phantom C resulting images quality factors. In tests there were included 4 selected regions.

	Original				SLSC-O			
	Contrast [dB]	CNR	SNR	PSNR	Contrast [dB]	CNR	SNR	PSNR
Lag 21								
Region 1	-0.67	0.77	11.90	23.2	-2.40	1.62	7.63	17.6
Region 2	-9.60	1.75	2.62	13.7	-12.2	2.08	2.76	12.6
Region 3	-12.0	1.49	2.00	12.9	-14.1	1.80	2.25	12.1
Region 4	-12.0	2.47	3.38	11.8	-13.5	2.90	3.75	11.1
Lag 41								
Region 1	-0.28	0.38	13.58	25.9	-1.84	1.33	8.19	18.1
Region 2	-5.53	0.97	2.08	18.2	-8.81	1.48	2.32	15.5
Region 3	-7.49	0.90	1.55	16.0	-10.5	1.37	1.96	14.3
Region 4	-7.83	1.29	2.20	15.1	-10.5	1.94	2.78	13.0
Lag 61								
Region 1	-0.02	0.03	14.80	26.7	-1.61	1.18	8.15	19.0
Region 2	-3.65	0.70	2.02	18.6	-7.44	1.34	2.32	17.8
Region 3	-5.49	0.61	1.30	17.0	-9.12	1.20	1.84	15.5
Region 4	-5.80	0.73	1.50	17.0	-9.22	1.51	2.31	14.3

SLSC-O images have a better contrast and SNR for regions 2–4, however, other factors are slightly inferior.

Additionally, calculations of the SSIM factor have been performed. Comparing the *Original* with *Speed Optimized* or *SLSC-O* methods results in SSIM value close to 1. For phantom W and lag numbers from 11 to 91, the calculated SSIM was always 1. For phantom C, however, the value was 1 for lag numbers below 21, and 0.9999 for lag numbers higher or equal to 21.

#### 4. Conclusion

The article presents a modification of the base SLSC (LEDIJU *et al.*, 2011) algorithm, which enables real-time implementation without a loss in quality of the reconstructed images.

The proposed method renders the execution time independent of the value of the lag parameter. The utilization of the GPU allowed to achieve 25 fps for a resulting image with a resolution of  $128 \times 512$  pixels, and 15 fps for  $128 \times 2048$  pixels. Moreover, the presented optimizations are not dedicated to the GPU only. The same speed-up results should be obtained while utilizing different platforms (ex. FPGA, CPU).

The higher values of lags carry less important but still valuable information. In such a case removing the averaging factor does not invalidate the data. It simply assigns a new weight to the correlation outcome, which decreases with the increasing lag value. These new weights match the importance of the parts of a sum. Changes in the results of the algorithm are at most at the level of  $1/(N - m)$  (see Fig. 2). When considering a large aperture size ( $N > 100$ ), the differences can be negligible. However, when SLSC-O is used for a small aperture, the difference can not be negligible.

The presented results show an improved quality in the upper part of the resulting images in comparison to classical SLSC. For the remaining parts of the images, the optimization results influence the differences in quality at a negligible level. We do not observe artifacts like in the *Original* approach at higher lag values. Unfortunately, our method does not reduce the general limitations of SLSC imaging in its inability to detect point-like targets in speckle-based background and signals away from the transmit focal depth show decreased spatial coherence (THOMPSON *et al.*, 2017; LEDIJU *et al.*, 2011; MALLART, FINK, 1991).

The *SLSC-O* method depends only on the number of elements in the aperture used, which can lead to different results with different apertures. The generic drawbacks of the SLSC algorithm, i.e. the need to choose the correct lag parameter number, still occur.

#### Appendix A. Detailed derivations of equations

Let us concentrate on Eqs (2) and (4). One can see that within these formulas a little different notations

are used (without losing generality and for further simplicity). The simplification can be expressed as:

$$s_i(n) = s_i(d_i(x, y) + n). \tag{10}$$

For more simplification let us introduce a formula that is a single component of sum in Eq. (4):

$$f(i, i + m) = \frac{\sum_{n=-\Delta c}^{\Delta c} s_i(n) s_{i+m}(n)}{\sqrt{\sum_{n=-\Delta c}^{\Delta c} s_i^2(n) \sum_{n=-\Delta c}^{\Delta c} s_{i+m}^2(n)}}. \tag{11}$$

Going further, the combination of Eqs (2), (3), and the newly introduced Eq. (11), without the averaging factor  $1/(N - m)$  can be rewritten as:

$$R_{sl} = \sum_{m=1}^M \sum_{i=1}^{N-m} f(i, i + m), \tag{12}$$

which corresponds to the original method. Our proposed method uses notation (see Eq. (4)) that can be simplified using Eq. (11) and rewritten as:

$$R_{sl} = \sum_{i=1}^N \sum_{j=i+1}^{\min(i+M, N)} f(i, j). \tag{13}$$

The formulas (12) and (13) may seem not equal, but they are. To explain that fact let us assume that  $N = 5$  and  $M = 3$  and consider the following example. Table 2 enumerates each single component of the sum of the sum for Eq. (12) and Table 3 enumerates the same components for Eq. (13).

Table 2. Single components for *Original* (see Eq. (12)).

$m = 1$ $i = 1$	$m = 2$ $i = 1$	$m = 3$ $i = 1$
$m = 1$ $i = 2$	$m = 2$ $i = 2$	$m = 3$ $i = 2$
$m = 1$ $i = 3$	$m = 2$ $i = 3$	
$m = 1$ $i = 4$		

Table 3. Single components for proposed *SLSC-O* (see Eq. (13)).

$i = 1$ $j = 2$	$i = 2$ $j = 3$	$i = 3$ $j = 4$
$i = 1$ $j = 3$	$i = 2$ $j = 4$	$j = 5$
$i = 1$ $j = 4$	$i = 2$ $j = 5$	
$i = 1$ $j = 5$		

Now let us concentrate on derivation Eq. (5). Let us rewrite Eq. (4) to following formula with some trivial moves:

$$R_{sl} = \sum_{i=1}^N \left[ \frac{\min(i+M,N) \sum_{j=i+1}^{\Delta c} \frac{\sum_{n=-\Delta c}^{\Delta c} s_i(n)s_j(n)}{\sqrt{\sum_{n=-\Delta c}^{\Delta c} s_j^2(n)}}}{\sqrt{\sum_{n=-\Delta c}^{\Delta c} s_i^2(n)}} \right]. \quad (14)$$

Now focus on the bottom divider. In our considerations here it depends only on value  $i$ . Because it does not depend on value  $j$ , one can separate this divider and the sum with  $j$ . As a result, the equation can take a new form:

$$R_{sl} = \sum_{i=1}^N \left[ \frac{\min(i+M,N) \sum_{j=i+1}^{\Delta c} \frac{\sum_{n=-\Delta c}^{\Delta c} s_i(n)s_j(n)}{\sqrt{\sum_{n=-\Delta c}^{\Delta c} s_j^2(n)}}}{\sqrt{\sum_{n=-\Delta c}^{\Delta c} s_i^2(n)}} \right]. \quad (15)$$

The next step is similar. Because  $s_i(n)$  does not depend on value  $j$ , it can be moved before the sum. This gives the following form:

$$R_{sl} = \sum_{i=1}^N \left[ \frac{\sum_{n=-\Delta c}^{\Delta c} s_i(n) \min(i+M,N) \sum_{j=i+1}^{\Delta c} \frac{s_j(n)}{\sqrt{\sum_{n=-\Delta c}^{\Delta c} s_j^2(n)}}}{\sqrt{\sum_{n=-\Delta c}^{\Delta c} s_i^2(n)}} \right], \quad (16)$$

which gives Eq. (5) with considering Eq. (6).

## Appendix B. Simulated data

For the validation of the experimental results, Field II software was used to simulate a PSF using 1D uniform linear array as a transducer with 128 elements and a center frequency of 5 MHz. The data were acquired from all 128 channels at a 50 MHz sampling rate. The speed of sound was set to the value of 1540 m/s. The element pitch was equal to 0.328 mm. A synthetic transmit aperture was generated using single element transmits to achieve full dynamic focusing on both transmit and receive parts. For the simulated data, uniform white noise was added to simulate acoustic noise received by the transducer. The noise was

added at the level such that the ratio of channel Signal to Noise (SNR) was 10 dB. This noise was added to simulate reverberation and the coherence of low-amplitude echoes.

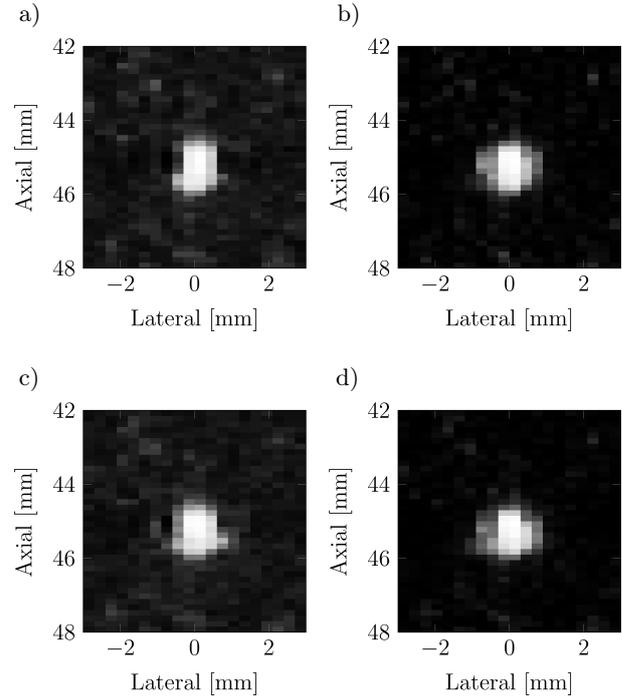


Fig. 11. Comparison of reconstructed images of a fragment for the simulated fiber (30 dB scale): a) Original Lag 15, b) Original Lag 50, c) SLSC-O Lag 15, d) SLSC-O Lag 50.

In Fig. 11, one can observe results for such simulation data for both the *Original* and the *SLSC-O* methods. Both methods give the same visual results on simulated data.

## Acknowledgements

This work has been partially supported by Project POIG.01.03.01-14-012/08-00, co-financed by the European Regional Development Fund under the Innovative Economy Operational Programme.

## References

1. BAMBER J.C., MUCCI R.A., OROFINO D.P. (2002), *Spatial coherence and beamformer gain*, [in:] *Acoustical Imaging*, Vol. 24, Springer Nature, pp. 43–48, doi: 10.1007/0-306-47108-6\_6.
2. BENZARTI F., AMIRI H. (2012), *Speckle noise reduction in medical ultrasound images*, *International Journal of Computer Science Issues*, **9**, 2, 3, 8 pages, <https://arxiv.org/ftp/arxiv/papers/1305/1305.1344.pdf>.
3. BOTTENUS N., BYRAM B.C., DAHL J.J., TRAHAY G.E. (2013), *Synthetic aperture focusing for short-lag spatial coherence imaging*, *IEEE Transactions on*

- Ultrasonics, Ferroelectrics, and Frequency Control, **60**, 9, 1816–1826, doi: 10.1109/tuffc.2013.2768.
4. CHEN T.J., CHUANG K.S., WU J., CHEN S.C., HWANG I.M., JAN M.L. (2003), *A novel image quality index using Moran I statistics*, Physics in Medicine and Biology, **48**, 8, N131–N137, doi: 10.1088/0031-9155/48/8/402.
  5. DAHL J.J., LEDIJU M.A., TRAHEY G.E. (2011), *Methods, systems and apparatuses for van-Cittert Zernike imaging*, US Patent 2013/0109971.
  6. GUNGOR M.A., KARAGOZ I. (2015), *The homogeneity map method for speckle reduction in diagnostic ultrasound images*, Measurement, **68**, 100–110, doi: 10.1016/j.measurement.2015.02.047.
  7. GUPTA S., CHAUHAN R.C., SEXANA S.C. (2004), *Wavelet-based statistical approach for speckle reduction in medical ultrasound images*, Medical & Biological Engineering & Computing, **42**, 2, 189–192, doi: 10.1007/bf02344630.
  8. HIREMATH P.S., PREMA T., BADIGER S. (2013), *Speckle noise reduction in medical ultrasound images*, [in:] Gunarathne G.P.P [Ed.], *Advancements and breakthroughs in ultrasound imaging*, InTech, Chapter 8, pp. 201–241, doi: 10.5772/56519.
  9. HYUN D., TRAHEY G.E., DAHL J.J. (2013), *A GPU-based real-time spatial coherence imaging system*, [in:] Proceedings of SPIE – Medical Imaging 2013: Ultrasonic Imaging, Tomography, and Therapy, J.G. Bosch, M.M. Doyley [Eds], Vol. 8675, doi: 10.1117/12.2008686.
  10. HYUN D., TRAHEY G.E., DAHL J.J. (2014), *Sparse sampling methods for efficient spatial coherence estimation*, [in:] 2014 IEEE International Ultrasonics Symposium, pp. 535–538, Chicago, IL, doi: 10.1109/ultsym.2014.0132.
  11. HYUN D., TRAHEY G.E., DAHL J.J. (2015), *Real-time high-framerate in vivo cardiac SLSC imaging with GPU-based beamformer*, [in:] 2015 IEEE International Ultrasonics Symposium (IUS), pp. 1–4, doi: 10.1109/ULTSYM.2015.0077.
  12. LEDIJU M.A., TRAHEY G.E., BYRAM B.C., DAHL J.J. (2011), *Short-lag spatial coherence of backscattered echoes: imaging characteristics*, IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control, **58**, 7, 1377–1388, doi: 10.1109/tuffc.2011.1957.
  13. LIU D.L., WAAG R.C. (1995), *About the application of the van Cittert-Zernike theorem in ultrasonic imaging*, IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control, **42**, 4, pp. 590–601, doi: 10.1109/58.393102.
  14. MALLART R., FINK M. (1991), *The van Cittert-Zernike theorem in pulse echo measurements*, The Journal of the Acoustical Society of America, **90**, 5, 2718–2727, doi: 10.1121/1.401867.
  15. MATRONE G., SAVOIA A.S., CALIANO G., MAGENES G. (2015), *The delay multiply and sum beamforming algorithm in ultrasound B-mode medical imaging*, IEEE Transactions on Medical Imaging, **34**, 4, 940–949, doi: 10.1109/tmi.2014.2371235.
  16. OVIREDDY S., MUTHUSAMY E. (2014), *Speckle suppressing anisotropic diffusion filter for medical ultrasound images*, Ultrasonic Imaging, **36**, 2, 112–132. doi: 10.1177/0161734613512200.
  17. POUREBRAHIMI B., YOON S., DOPSA D., KOLIOS M.C. (2013), *Improving the quality of photoacoustic images using the short-lag spatial coherence imaging technique*, [in:] A.A. Oraevsky, L.V.Wang [Eds], Proceedings of the Photons Plus Ultrasound: Imaging and Sensing 2013, Vol. 8581, San Francisco, CA, doi: 10.1117/12.2005061.
  18. S-Sharp (2019), *Prodigy – Features*, [http://www.s-sharp.com/web/products/products.jsp?dm\\_id=DM1412758766647](http://www.s-sharp.com/web/products/products.jsp?dm_id=DM1412758766647) (accessed on March 7, 2019).
  19. THOMPSON, A.R., MORAN J.M., SWENSON G.W. (2017), *Van Cittert-Zernike theorem, spatial coherence, and scattering*, [in:] *Interferometry and synthesis in radio astronomy*, Springer International Publishing, Cham, pp. 767–786, doi: 10.1007/978-3-319-44431-4\_15.
  20. TROTS I., NOWICKI A., LEWANDOWSKI M., TASINKEVYVH Y. (2010), *Multi-element synthetic transmit aperture in medical ultrasound imaging*, Archives of Acoustics, **35**, 4, 687–699, doi: 10.2478/v10168-010-0052-y.
  21. VANITHAMANI R., UMAMAHESWARI G. (2014), *Speckle reduction in ultrasound images using neighshrink and bilateral filtering*, Journal of Computer Science, **10**, 4, 623–631, doi: 10.3844/jcssp.2014.623.631.
  22. Verasonics (2019), *Vantage – Research ultrasound*, url: <http://verasonics.com> (accessed on March 7, 2019).
  23. WALCZAK M., LEWANDOWSKI M., ZOLEK N. (2013), *Optimization of real-time ultrasound PCIe data streaming and OpenCL processing for SAFT imaging*, [in:] 2013 IEEE International Ultrasonics Symposium (IUS), Institute of Electrical and Electronics Engineers (IEEE), pp. 2064–2067, doi: 10.1109/ultsym.2013.0527.
  24. WANG Z., BOVIK A.C., SHEIKH H.R., SIMONCELLI E.P. (2004), *Image quality assessment: from error visibility to structural similarity*, IEEE Transactions on Image Processing, **13**, 4, 600–612, doi: 10.1109/tip.2003.819861.