UDC 004.4, 004.6=111

A.V. Fedorov, V.V. Potekhin

INTELLIGENT PROCESSING ALGORITHM OF ULTRASOUND IMAGES IN BIOMEDICINE

A software package for ultrasound imaging in the form of an elastogram has been developed to measure stiffness of human tissue. A technique for obtaining elastograms is based on the least squares method and identification of feature points in images. The method can be used for clinical purposes when ultrasound machines have no elastogram display.

ELASTOGRAPHY; ULTRASOUND IMAGE; IMAGE PROCESSING; MACHINE LEARNING.

А.В. Федоров, В.В. Потехин

ИНТЕЛЛЕКТУАЛЬНЫЙ АЛГОРИТМ ОБРАБОТКИ УЛЬТРАЗВУКОВЫХ ИЗОБРАЖЕНИЙ В БИОМЕДИЦИНЕ

Разработан программный комплекс для расчета жесткости органов человека на основе ультразвукового изображения для отображения эластограммы. Представлена методика получения эластограммы на основе метода наименьших квадратов и определения особых точек в изображении. Метод применим для проведения диагностики на аппаратах УЗИ, в которых отсутствует функция отображения эластограммы.

ЭЛАСТОГРАФИЯ; УЛЬТРАЗВУКОВОЕ ИЗОБРАЖЕНИЕ; ОБРАБОТКА ИЗОБРАЖЕНИЙ; МАШИННОЕ ОБУЧЕНИЕ.

Ultrasound elastography is a method that measures the deformation of tissue in response to an applied force to determine and display its stiffness. Tissue strain analysis gives a new dimension in ultrasound imaging, which includes such properties as quality and quantitative assessment of the tissue elasticity. Numerous applications in this field of diagnostic tools can provide more accurate diagnoses than one imaging modality alone [1]. In this paper, we describe an intelligent algorithm to acquire an elastography image, making use of conventional ultrasound equipment. We have developed a video streaming application that can help physicians to solve diagnostic problems. A multicore approach was employed to get near-real time execution of the algorithm.

With ultrasound elastography, diagnosticians may no longer need to use finger palpation to detect hard tissues. An elastogram will show if the tissue being examined is stiffer than the embedding tissue, which might suggest that this tissue can be abnormal, e. g., malignant nodules are stiffer than benign nodules. There are many systems that use built-in functions of elastography, exploiting the property of the ultrasound wave − shear-wave propagate faster in stiff tissue than in soft tissue, the tissue stiffness can be inferred by measuring shear-wave velocity [2, 3]. Unfortunately, such systems are very costly. The approach described in this paper aims to provide ultrasound elastography imaging with a standard non-modified ultrasound device not equipped with an elastography unit. All calculations were performed on a PC connected to an ultrasound machine.

All existing methods for providing additional information from an ultrasound image can be divided into three types [4].

Strain Imaging Methods:

Quasi Static Methods (Ophir et al., 1991; O'Donnell et al., 1991);

Dynamic Methods (Parker et al., 1990; Krouskop et al., 1987; Sandrin et al., 1999);

Radiation Force (Walker, 1999; Fatemi & Greenleaf, 1999; Nightingale et al., 2002; Lizzi et al., 2003).

Stress Imaging Methods:

Mechanical or Tactile Imaging (Sarvazyan et al., 1998; Wellman et al., 2001):

Computational Models;

Finite Element Modelling;

Using Surface Pressure Information.

Modulus Imaging Methods:

Iterative Modulus Reconstruction (Kallel et al., 1995);

Direct Methods (Solving PDE's) (Emelianov et al., 2000; Sumi et al., 1995);

Finite Element Inversion (Zhu et al., 2003).

In this paper, we discuss the implementation of the intelligent image processing algorithm for ultrasound elastography supported with multi-core calculations. An approach that does not allow quasi-static, shear wave or acoustic radiation force impulse (ARFI) imaging lacks precision and requires clinicians to put in some additional effort [5]. Our algorithm makes use of video streaming only and does not account for the velocity of a wave. These binding constraints imply that the system is to be used dynamically — the operator will have to apply some pressure with an ultrasound device on a periodic basis.

To start with, a video-stream from an ultrasound device is transmitted to a PC via the VGA-port. The program captures frames from this stream and displays them in the main window. The user selects a region of interest (ROI), and the program accumulates the images in the image storage. The images uploaded to the image storage are grayscale. To create a visual map, or elastogram, the program renders frames in real-time and, applying a special algorithm, calculates the relative stiffness of one tissue section in comparison with the other tissues in ROI. The algorithm is akin to the schema in neural networks. First, we compute the weights for each pixel or zone of pixels of the image stored in the image storage. Then, having matched up the actual image to the estimated weights and having made use of a threshold function, we can see the difference in density, as it is indicated by the color of every dissimilar pixel in the zone of pixels. A threshold function supports four color spaces — blue (soft tissues), green (soft tissue of medium density), yellow (stiff tissue of medium density), and red

(the most rigid tissue). Before getting to the image storage, images are put through a filter. It is aimed at ensuring accuracy and helps to avoid saving artifacts. Another algorithm allows the image storage to get updated to the latest frame, which means getting the target image right after pressure was applied to the tissue.

The program can extract visual features from ultrasound images. When the user selects a region of interest, the program collects black and white images $I_1, I_2, I_3, ..., I_m$ compressed to an optimal size (to accelerate the program and image delivery) and saves them in an intermediate image stack of a given size:

$$S = \{I_1, I_2, I_3, ..., I_m\},$$

where n is a whole number.

The following actions can be performed with the selected area:

inverting colours;

equalizing the histogram;

displaying the elastogram;

identifying feature points and showing their displacement vectors.

When the user chooses an overlay of the elastogram, each current image C is compared with a stack of images, using the following formula:

$$E_{xy} = \frac{\sum_{n=0}^{m}(C_{xy} - I_{xy_n})^2}{n},$$

where E_{xy} is a part of the selection (may consist of one pixel or of the square area with sides of 10 pixels). This, with the addition of some RGB schemes, determines the color of the pixels on the elastogram. There is a choice of four colors: red (if $6 \geq E_{xy} \geq 0$), yellow (if $10 \geq E_{xy} > 6$), green (when $20 \geq E_{xy} > 10$), and blue (if $E_{xy} > 20$). These colors were selected in accordance with established practice. The thresholds were chosen on the basis of the study of the boundary correspondence between certain types of tissues that have different stiffness. After every fifth iteration (frame updating or a certain period of time), the oldest image is deleted and a new one is added to the stack. This methodology relies on image dynamics — after the tissue was subjected to external pressure, its deformed areas are highlighted blue, and its rather rigid areas are highlighted red.
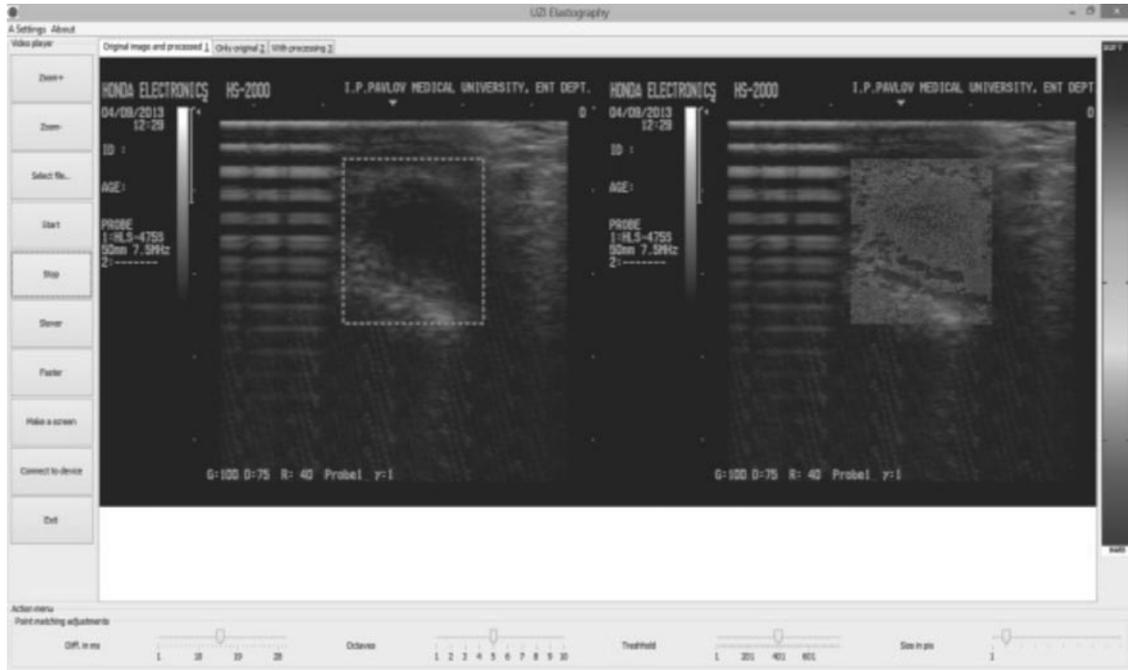
Fig. 1. Ultrasound and elastography images used to identify tissue stiffness

The elastography image gets displayed next to the corresponding original image. To remove noise, the user can increase the size of the E_{xy} by utilizing a special slider, and each pixel will be replaced with a cluster of them.

The image stack holds up to 30 images. If, after 5 iterations, the number of the images stored in the stack equals or exceeds its maximum capacity, the first 15 images are removed, and the program retains only the latest 15 images, followed by another 30 ones to fill the stack. These limitations are empirical in nature and stem from the acquired level of accuracy and the obtained results.

To further increase accuracy, there is another stack which is updated alongside with the original one and holds 50 large-scale images. These sample images are periodically com-
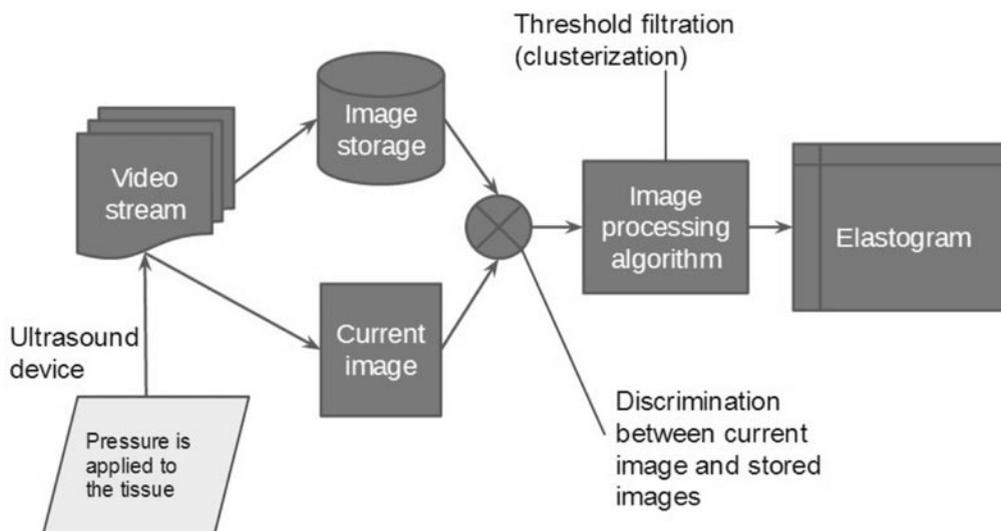

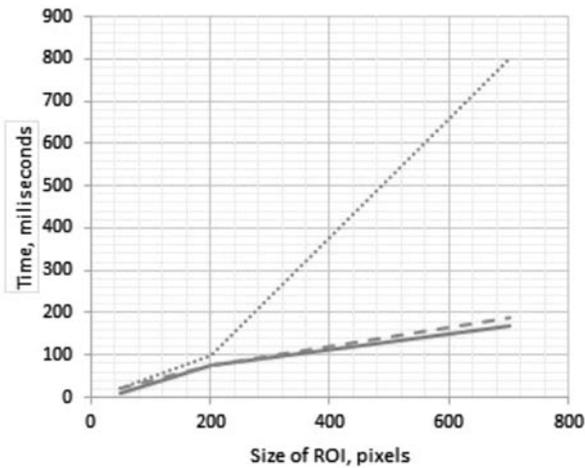Fig. 2. Schematic algorithm for computing an elastogram

Fig. 3. Correspondence between the computation
rate and the window size in combination with
various parameters of parallelization
(•••••) without parallel computing;
(————) with parallel computing X axis;
(– – –) with parallel computing Y axis

pared with those from a smaller stack, so that severe errors in the calculations for an elastogram would not be missed. If any discrepancy is detected, both stacks are utilized to form the next frame to compute the elastogram.

The results of the computed analysis that involved examining a human lymphoma are shown in Fig. 1. The lymphoma can be seen within the selected region of interest in the left image. The post-processed image on the right side is an elastogram. As expected, the program indicates the rigidity of the lymphoma tissue with a red highlight, while the area around it is tinted green and blue, which implies that it is composed of fat or muscle.

Fig. 2 presents the scheme for computing an elastogram.

The multi-core approach was executed by doing parallel weight computation of the image storage in each frame. Parallel computing in the ROI was down by rows [6]. Such an approach allows for almost real-time elastography mapping. When the program is used on a PC with a Core 2 processor, the FPS is only 15, but with a CPU Core i7 processor, the FPS amounts to 45, which makes it possible for the radiologist to view a real-time video feed of the ultrasound exam and conduct an immediate analysis of the images. The correspondence between the computation rate and the window size, coupled

with various parameters of parallelization, is shown in Fig. 3. With no parallelization, the computation time grows exponentially with the increasing size of the ROI. The computational procedure of creating an elastogram was tested not only without parallelization, but also with parallelization in the X and Y axes. The program produces similar results for parallelization both in the X axis and in the Y axis.

Having studied different approaches, we decided on such methods for detecting feature points as SIFT and SURF. Feature points are used to spot some contrast differences between images and determine how far these points are from each other. As soon as feature points have been detected, the program makes use of the K-means clustering algorithm to find the nearest points in the previous image and defines the displacement vector for each point [7]. But the application of these methods is complicated in large part due to the specific ultrasound imagining characteristics — ultrasound images feature unclear contours and there is no way to find any certain point. If we lower the threshold for finding feature points, computation time will increase exponentially [8]. The results of computing the SURF algorithm with different thresholds are presented in Fig. 4.
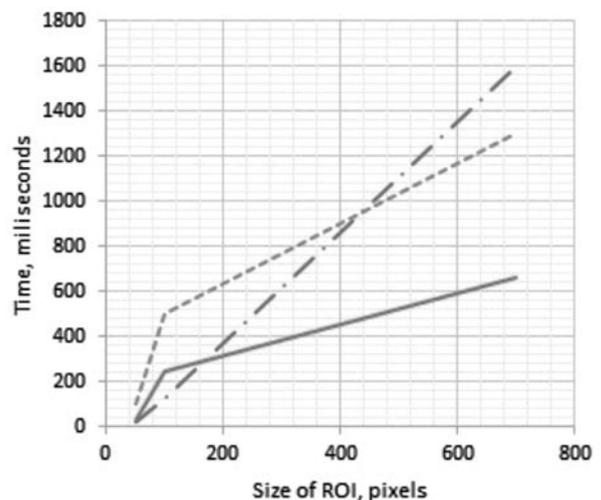

Fig. 4. Correspondence between computation time
and the size of a ROI in relation to different values
of the threshold in the SURF algorithm
(— · —) treshold equals 400;
(————) treshold equals 100;
(– – –) treshold equals 50

Compared with the SURF algorithm, the SIFT method turned out to take much longer to yield results and produced fewer feature points than the application requires in order to work properly. After applying one iteration, the user can see the displacement vectors of the feature points detected earlier.

The direct comparison method has proven to be the most effective of all.

We are planning to add functionality to the program by implementing the following features.

• Categorization of tissues according to their stiffness to enable anomaly detection.

• Machine learning aspect [9]. The learning aspect will involve learning on test cases featuring malignant and benign nodules [10].

• The function for selecting two ROIs and defining their correlation coefficient.

• The function for determining the Young's modulus for any point in the image.

• Utilizing finite element analysis of images.

• Transducing the contact force with which an ultrasound device is pressed against the skin, correlating the resulting data with the registered strain, and thereby determining more accurate measurements of tissue stiffness [11].

• Clustering selected points or pixels of any color to visually identify the contour lines.

• In the future, to classify diagnostic findings, the operator will have to save research results in a database.

In our laboratory, we used a General Electric 200 PRO ultrasound machine and a PC with a Core-i7 CPU that has 4 cores. The application has proven successful in laboratory testing, and now it is being tested by medical experts who are to estimate its precision and evaluate the utilization of the statistical parameter.

The major findings of the current study can be summarized as follows.

The given method suffers from several limitations: some outer tissues are of the same color as the tissues inside of them, and as the program works only with images, it simply cannot distinguish this hidden layer.

In 17 out of 20 video streams showing different malignant nodules, the program determined that the tissue surrounding these lumps were much stiffer than the other tissues and highlighted them red, which establishes the precision of the algorithm at 0.85. In contrast, the accuracy level of the elastography method based on the quasi-static approach is 0.95 [12].

Feature points algorithms are applicable only in cases with small-sized ROIs.

When used on a multicore PC, the application provides near real-time video stream processing.

The method is noise-resistant and not subject to perspective distortion.

The method is conditional on pressure being applied periodically to the area meant to be examined with elastography.

The program was written in the Java and Scala programming languages, thus it is possible to run it on any OS where the JDK v.7 is installed.

To obtain more accurate results, such features as the histogram equation, color inversion and binary comparisons between images were utilized in the program [13]. The user can choose which function to make use of and compare results. In addition, the user of the program can stop a video stream to examine the image. Alternatively, the user can slow a video down or speed it up when needed.

REFERENCES / СПИСОК ЛИТЕРАТУРЫ

1. **Benson J., Fan L.** A Complete Ultrasound Solution for Elastography, Tissue Strain Analytics, Siemens Medical Solutions USA, Inc., 2012.

2. **Si Luo, Cheoljin Lee, Kerem Karadayi, Yongmin Kim** Real-Time Ultrasound Elastography on a Multi-core DSP, Texas Instruments Incorporated, 2011.

3. **Kolesnikov M.P., Potekhin V.V., Tchernykh T.Y.** Active vibration control using modal analysis and rejection of resonance. *15th Internat. Congress on Sound and Vibration,* 2008.

4. **Saftoiu A., Vilmann P., Gorunescu F., Gheonea D.-I., Gorunescu M., Belciug S.** Endoscopic Ultrasound Elastography in the Diagnosis of Pancreatic Cancer. *Annals of gastroenterology*, 2010, Vol. 23(3), pp. 200-201.

5. **Varghese Tomy.** Elastography: Imaging the elastic properties of soft tissues with ultrasound. *Journal of Medical Ultrasonics*, 2002, Vol. 29, Iss. 4, pp. 155−171.

6. **Carlsen J.F., Ewertsen C., Lunn L., Nielsen M.B.** Strain Elastography Ultrasound: An Overview with Emphasis on Breast Cancer Diagnosis. *Diagnostics,* 2013, No. 3.

7. **Bishop C.M.** Pattern recognition and machine learning, 2006.

8. **Luo Juan, Oubong Gwun** A Comparison of SIFT, PCA-SIFT and SURF. *Internat. Journal of Image Processing*, 2009, Vol. 3, Iss. 4.

9. **Varghese T., Zagzebski J.A., Frank G., Madsen E.L.** Elastographic imaging using a handheld compressor, *Ultrasonic Imaging*, 2002, Vol. 24, pp. 25−35.

10. **Garra B.S.** Tissue elasticity imaging using ultrasound, *Applied Radiology*, Apr. 2011.

11. **Chen L., Housden R.J., Treece G.M., Gee A.H., Prager R.W.** A hybrid displacement estimation method for ultrasonic elasticity imaging, CUED/F-INFENG/TR 615, Cambridge University Department of Engineering, Nov. 2008.

12. **Kapoor Atul, Kapoor Aprajita, Mahajan Goldaa** Real-time Elastography in Acute Appendicitis. *Ultrasound Med*, 2010, No. 29, pp. 871−877.

13. **Rangayyan Rangaraj M.** Biomedical Image Analysis, 2008.

FEDOROV, Aleksandr V. *St. Petersburg State Polytechnical University.*
195251, Polytekhnicheskaya Str. 21, St. Petersburg, Russia.
E-mail: aleksandr.v.fedorov@gmail.com

ФЕДОРОВ Александр Викторович − *студент кафедры систем и технологий управления Санкт-Петербургского государственного политехнического университета.*
195251, Россия, Санкт-Петербург, ул. Политехническая, д. 21.
E-mail: aleksandr.v.fedorov@gmail.com

POTEKHIN, Vyacheslav V. *St. Petersburg State Polytechnical University.*
195251, Polytekhnicheskaya Str. 21, St. Petersburg, Russia.
E-mail: slava.potekhin@gmail.com

ПОТЕХИН Вячеслав Витальевич − *доцент кафедры систем и технологий управления Санкт-Петербургского государственного политехнического университета, кандидат технических наук.*
195251, Россия, Санкт-Петербург, ул. Политехническая, д. 21.
E-mail: slava.potekhin@gmail.com