Software Tools for Manual Segmentation of Tomography Images Supporting Radiologist's Personal Context

Vitaly Boitsov, Alexandra Vatian, Nikolay Egorov,
Anton Klochkov, Artem Lobantsev, Ekaterina Markova,
Natalia Gusarova, Anatoly Shalyto
ITMO University
Saint Petersburg, Russia
alexvatyan@gmail.com

Alexey Zubanenko
"Teleradiomedicine" LLC
Saint Petersburg, Russia
zubdocmri@gmail.com

Roman Soldatov
"Ramsay Diagnostics Rus" LLC
Saint Petersburg, Russia
alterroma@yandex.ru

Rajdeep Niyogi
Indian Institute of Technology (IIT)
email: rajdpfec@iitr.ac.in

Abstract—**The article addresses the issue of developing of software tools for manual segmentation of tomography images supporting radiologist's personal content. Through an expert survey we have identified the requirements for such software tools from the doctors' point of view as end users, as well as to the nomenclature and functionality of tools that implement these requirements. In order to meet the identified requirements, we have developed a solution based on a client-server architecture with a cloud access point. The nomenclature of tools for marking tomographic images implemented in the solution, as well as the methodology for working with them, fully complies with the identified requirements. The following functions have been developed and implemented: calculation of the volume of the region of interest, as well as three options for the semiautomatic segmentation of the image based on threshold, extreme points and neuron networks. All functions have customizable parameters and (or) implementation options, which provides flexibility in solving specific markup problems. Experimental studies have shown that the constructed service meets all the requirements put forward by radiologists and corresponds to the global level in terms of accuracy and speed (performance) of segmentation.**

I. Introduction

The transition to personalized medicine highlights the diagnostic techniques that most accurately characterize a particular patient and best reflect the doctor's opinion of that patient. Such techniques, unconditionally, include tomographic images. By tomography in medicine is meant any method that produces images of single tissue planes [1]. In reconstruction tomography, all images are processed by special computer program, and as a result, a three-dimensional image of the organ is modeled.

One of the key stages of processing tomographic images is their segmentation. The quality of segmentation largely determines the final result of image analysis and the result of diagnostics in general.

Segmentation is intended to highlight regions with certain properties on images, such regions usually corresponding to objects of interest or their fragments. Authors [2], [3] define image segmentation as the process dividing an image into regions with similar properties such as gray level, color, texture, brightness, and contrast.

According to [4–6], the segmentation of medical images pursues the following goals:

- Study anatomical structures of different organs
- Identify region of interest, in particular, the separation of healthy and pathological areas (locating tumor, lesion and other abnormalities)
- Measure tissue volume to assess the dynamics of tumor development (increase or decrease in size of tumor with treatment)
- Help in treatment planning prior to radiation therapy; in radiation dose calculation

In recent years, due to the widespread use of machine learning methods in medical images processing, another goal has been added to this list: creating a sample dataset for training neural networks designed to select objects of interest in automatic mode [7].

The problem of medical images segmentation has been known for a long time, so far various classification approaches to its solution have been proposed. For example, in [4] segmentation techniques are divided into two groups, namely gray level based and textural feature based techniques. [8] divides segmentation methods into three groups according to the increase in the algorithmic complexity of the applied algorithms. [5] classify segmentation methods in four categories: region-based methods, clustering methods, classifier methods, and hybrid methods. In [9] it was proposed to separate segmentation techniques into manual and computer-aided, the latter being, in turn, divided into automatic (unsupervised), interactive (semi-supervised), and supervised. A similar classification was proposed in [6]: supervised, unsupervised and interactive techniques stand out here, but manual segmentation is not even mentioned. The authors [10], [11] also take manual segmentation into account in their classification, namely: manual segmentation, intensity-based methods, atlas-based methods, surface-based methods and hybrid segmentation methods.

ISSN 2305-7254

Summarizing the aforementioned and other works, we can conclude that as the basis for the classification of medical image segmentation methods, individual image features (e.g., contrast, texture, etc.), or types of segmentation algorithms (e.g., clustering, active contour, etc.) are mainly used. At the same time, manual segmentation is based on a different principle. In this case, the expert physician first makes an integral assessment of the 3D image built in the viewer, and based on his own context (knowledge and personal experience) decides that a particular organ fragment to be segmented is an area of interest (for example, a tumor), and the image of this fragment is localized on certain slices. On each of the mentioned slices, the expert draws with the help of a computer drawing tool ("brush" or "pencil") around the contours of this zone of interest, and then saves the entire group of slices under the same name.

The fundamental reliance of manual segmentation on personal context provides this method with undeniable advantages: it is believed to be the most accurate and reliable and thus used as “ground truth” and for quantitative evaluation of automated segmentation methods [11]. At the same time, this circumstance can entail different interpretations of the same image. For example, in [12] two independent experts carried out purely manual segmentation of 3D images of the complete human mandible. Only images with clear bone contours and anatomical structures without artifacts were used. Even in these idealized conditions, the discrepancy between both experts, measured through the Dice score coefficient, was up to 9%, and both segmentation results for each image were included as ground trooth in the generated dataset.

For areas of interest with low contrast boundaries and, especially, for neoplasms and tumors, an even greater discrepancy is observed [13]. In these cases cases it is mostly important to identify and preserve the results of segmentation, reflecting the views of individual experts, especially for assessing the dynamics of the development of the area of interest, i.e. to detect small changes in the size or configuration of the neoplasm.

Manual segmentation is not error free. An important source of errors is the presence of artifacts in the analyzed image [4, 14]. A variety of algorithms have been proposed to combat artifacts. However, according to radiologists’ opinion, excessive suppression of artifacts can interfere with an adequate image assessment [15].

A significant source of errors is also the tools present in the workspace of the radiologist. Insufficient, inadequate or excessive number of tools and options leads to a cognitive overload of the radiologist, especially in the situation of highly repetitive tasks having stringent requirements of accuracy and speed [16].

Besides, manual segmentation is extremely slow and tedious, and an expert physician can make mistakes due to fatigue. To solve this problem, interactive segmentation methods [9] can be used. In this case, the rough segmentation results obtained using automatic algorithms are iteratively refined by the intervention of human experts. Obviously, the effectiveness of these methods depends not only on the automatic segmentation algorithm used, but also on the organization of the iterative procedure, i.e. on the method of taking into account the expert opinion.

Although so far a lot of software tools for manual segmentation have been proposed (their brief review is given in the next section), the urgent problem is to take into account the needs of the doctor as much as possible, i.e. provide him with a tool for adequate implementing and maintaining his own context and at the same time for preventing segmentation errors. This is especially important in conditions of personalized medicine and in the rapid assessment of the dynamics of diseases.

II. Background and Related Works

In modern clinical practice, three main technologies for obtaining tomographic images dominate – computer tomography (CT), positron emission tomography (PET) and magnetic resonance imaging (MRI).

With CT, using X-ray radiation, a sequence of two-dimensional slices of the examined area of the body is formed, from which, using a specialized computer program, a pseudo-3D image of the structure of organs is formed. PET is a radioisotope research method when a radiopharm drug is incerted intravenously, which is distributed throughout the body and selectively accumulates in the organ that is affected by the disease; three-dimensional image of radiopharm drug concentration within the body is then constructed by computer analysis similar to CT. In MRI, a strong magnetic field is used and the picture of its gradients in different parts of the patient’s body is fixed on a sequence of slices; then a pseudo-3D image of the volumetric structure of organs is constructed of them.

Thus, from a technical point of view, the tomographic images being obtained are very close and the differences in the methodology for their segmentation are associated mainly with the specifics of the organs and zones of interest under study. As clinical practice shows, the most complex and at the same time mostly demanded diagnostically are the tasks of segmentation of CT images of the lungs and MRI images of the brain.

For example, on CT images of lung cancer, the malignant nodes are usually small and slightly different from the benign in most features which can be objectively registered [17]. MRI images of the brain are performed simultaneously in four modalities: T1-weighted MRI, T2-weighted MRI, FLAIR and FLAIR with contrast enhancement [18]. During segmentation, the expert physician analyzes all the images in the complex, and the segmentation efficiency is largely determined by the analysis pipeline developed by himself. In addition, diseases of these organs can develop very quickly, and the vital task is to control small – up to 5–10% – changes in areas of interest.

As the basic tools for manual segmentation of medical images workstations are traditionally used. Workstations are stationary software and hardware complexes, aggregating visualization tools for any medical images (not only tomographic), as well as means of communication, storage, etc. Widespread workstations include General Electric Advantage Workstation (https://www.gehealthcare.com/education/advantage-workstation-for-diagnostic-imaging), Philips MR Extended WorkSpace (https://www.learningconnection.philips.com/en/course/ extended-mr-workspace-r2631-overview), OsiriX MD [19]. As a rule, workstations include powerful stationary computers and several specialized monitors. For example, using OsiriX requires certified monitors for medical imaging.

However, in recent years, in addition to such "heavy" decisions, more and more manual segmentation support applications appear in clinical practice. These are "light", portable solutions for the operational support of the work of a radiologist, specialized for imaging of specific types (in particular, tomographic). They are presented both in the form of standalone applications, and in the form of services. Most of these applications are presented as open-source solutions.

The zoo of such applications is already quite wide. In particular, as the most common programs specialized for segmentation and estimation of the size of volumetric brain formations can be mentioned the following: Horos [20], Slicer 3D (https://slicer.readthedocs.io), medinria (http://med.inria.fr/), MRICRON (https://www.nitrc.org/projects/mricron), MITK (http://mitk.org/wiki/The_Medical_Imaging_Interaction_ Toolkit_(MITK)) . Their detailed usability analysis is provided in Section III.

In connection with the development of machine learning and pattern recognition algorithms, new applications and add-ons to them are constantly appearing. As a rule, they are associated with the inclusion of additional semi-automated algorithms in the segmentation process.

For example, in [21] an open-source extension for the Slicer 3D application named DeepInfer was proposed, which provides the possibility of automatic integration into Slicer 3D using a docker, which implements a neural network. The extension uses pre-trained deep learning networks to segment specific medical objects of interest. Docker stores ready-made models on the cloud. DeepInfer allows you to select a model, send it data for processing, and see the result. However, the solution is quite "heavy": the calculations take place on the client side, for efficient use discrete graphics is needed.

Another extension for Slicer3D called TOMAAT [22] implements volumetric medical image analysis as a cloud service. It also uses pre-trained deep learning networks, but the calculations take place on the cloud. TOMAAT allows you to configure the “infrastructure” of the neural network, set the interface (which inputs, which outputs), configure the activation function. Obviously, such things require special qualifications and cannot be performed by a doctor.

The open-source standalone solution of RIL-Contour [23] uses a simple but interesting method of iterative annotation. The doctor first marks out a small part of the dataset on which the “rough” network is trained. This network marks the next piece of dataset. The images marked by the network are corrected by the doctor. On an enlarged dataset, the network is retrained, and so on until the entire dataset is covered. Each iteration, obviously, requires the doctor to adjust the network prediction less and less.

The solution RIL-Contour has several advantages in terms of usability: it is possible to associate files with external medical databases; several doctors can simultaneously work with one dataset, while the results of each remain personalized; version control system MIRMAID is supported. The solution has built-in statistical functions, as well as calculating the volume, area and linear dimensions of the largest objects. However, it also has fundamental shortcomings: it is not possible to display a 3D model, instead, only 2d slices are processed, and for them there is no coordinate reference. The technical disadvantages include the following: local installation is required; only one wrapper format of the Neuroimaging Informatics Technology Initiative (NIfTI) is used.

In addition to the completed applications, start-ups are of great interest, which offer various tools that implement the basic requirements for manual segmentation systems.

For example, the authors [24] propose a set of tools for manual segmentation of medical images that should be implemented in a virtual laboratory SPINE (https://spinevirtuallab.org/public/#/index). As the most common manual segmentation tools, authors name Paint brush, Bezier curves (control points) and Fill closed contour, and as the most perspective methods of semi-automatic they use Active contours 2D and 3D.

In [25] the tool for semi-automatic and manual editing of lesion masks is presented. The peculiarity of the tool is that the doctor creates a clipping mask, and then, when editing it, marks false-positive and false-negatives points. Then the tool automatically adds or removes fragments of lesions mask on the marked coordinates.

The idea of using individual image points set by an expert for semi-automatic mask construction is also being implemented in other works. For example, in [26] it is proposed to retrain a deep neural network based on two types of points that the expert notes: ”foreground clicks”, which should be placed within the zone of interest, and “background clicks” that are not in this zone. For smoothing, a Gaussian filter is used, which, according to the authors of the work, increases the accuracy and speed of retraining. However, the proposed architecture is implemented on only one selected area, which increases the overall setup time of the algorithm.

In [27], an algorithm for segmenting images by extreme points of an object of interest is proposed. The user marks 4 extreme points (top, bottom, left-most, right-most) of the segmented object, the algorithm cuts a rectangular area with sides passing along these points, with a small gap to capture the context. Then, the pre-trained network resnet101 is used to refine the segmentation. The authors report high accuracy and speed of the algorithm, however, the experiments in the work were performed on a general type dataset (PASCAL), and not on medical images.

The above analysis shows a wide variety of existing and proposed tools and algorithms to support the segmentation of medical images, and in many ways they duplicate each other. In the vast majority of cases, the authors evaluate them from a technical point of view, and not from the point of view of expert doctors as end users of the proposed developments.

At the same time, in the Introduction the leading role of clinical experience and the knowledge of the physician in the segmentation of tomographic images is justified. In order for doctors to make the best use of them, segmentation support programs must meet certain usability requirements, primarily in terms of achieving specified goals [28], namely increasing the productivity and the quality of the segmentation.

Various aspects of the usability of systems to support manual segmentation of medical images are discussed in the literature [29]. Ergonomic requirements are mainly considered, such as the doctor's posture for analysis, the quality and nomenclature of monitors [30], the use of other visual means - touchscreens, holographic, kinetic sensors and eye tracking, holographic display and augmented reality [31]. At the same time, [31] notes that the mouse and keyboard remain the most utilized user interfaces for radiologists.

Various sequences of actions for segmentation [16] are also studied with the goal of developing an optimal scenario. [16] proposed a sequence of Stage actions, which in experiments showed an average of 37% reduction in the interpretation errors, and improved user satisfaction. But here it is noted that the effectiveness of the application of such a scenario very much depends on the individual preferences and experience of the radiologist.

A number of works [32], [33] formulated requirements for tools for segmenting medical images. However, they are focused only on workstations and are not so much functional as ergonomic in nature, for example: fast and easy availability of image data, simple postprocessing etc. At the same time, as follows from the above review, the functional requirements for workstations and applications for segmentation of medical images vary significantly, and the range of tools that can implement the same functionality expands very quickly. Unfortunately, in the available literature we were unable to find works that would analyze the requirements for applications for segmentation of tomographic medical images from the point of view of doctors as end users.

Thus, the authors of this article set themselves the following tasks:

1. to identify the applications requirements for the segmentation of tomographic medical images from the doctors' point of view of as end users, as well as to the nomenclature and functionality of tools that implement these requirements;

2. to consider the possibility of implementing the identified requirements;

3. to conduct experimental studies of the main functionalities of the developed service in terms of meeting the identified requirements.

III. REVEALING OF USABILITY REQUIREMENTS FOR MANUAL SEGMENTATION OF TOMOGRAPHY IMAGES

To create a list of usability requirements for tomographic image segmentation support applications, we used the following methodology.

1. We analyzed the tools for manual segmentation and highlighted the characteristics of their usability. For analysis, we selected programs that are commonly used in clinical practice, as well as the most promising startups (a brief review is given in the previous part of the article). In total, 20 different programs were analyzed.

2. We interviewed ten expert radiologists, each of whom independently described his own pipeline being used in segmentation of lung and brain tomographic images, and the usability characteristics that are desirable for its effective implementation. The experience of the involved radiologists ranged from 3 years to 21 years, with the average 8.4 years.

3. We combined the lists obtained in clauses 1 and 2 into a common list containing 27 requirements.

4. We invited each radiologist to select five most significant usability characteristics from the list constructed in clause 3 (see Table I). The characteristics with 3 or more votes were included in the resulting list..

TABLE I. SELECTION OF THE MOST SIGNIFICANT CHARACTERISTICS OF USABILITY

Characteristics	Radiologists									
	R1	R2	R3	R4	R5	R6	R7	R8	R9	R10
Choice of the number of windows	+									
XLS-segmented data upload			+							
3 D point		+				+	+			
Multiformat support			+			+		+		
Quick screenshot				+						
File manager		+								
Function navigator					+					
Quick access to segmentation volume calculation	+			+	+					
Subtraction of contrasting images						+				
Density determination by HU		+								
Image inversion							+			
Change segment transparency	+									
Report window (report template)		+								
Neural network segmentation								+	+	+
Corporate chat in a separate window							+			
Standard interface (toolbar)	+	+		+	+	+		+	+	+
Segmentation smoothing tool			+							
Scissors tool				+						
Possibility of multimodal segmentation			+				+		+	
Semi-automatic threshold segmentation	+	+			+			+		+
Anatomical segmentation					+					
Adding filters							+			
Extreme point segmentation				+		+		+		
Thresholding algorithm selection									+	
Image registration										+
Setting pixels to 0									+	
Missing ROI points generation										+

Thus, the final list of requirements was formed, consisting of 8 items:

(1). A set of tools for visual adjustment, including scrolling slices with the mouse wheel, quick adjustment of the left and right mouse buttons, toolbar (contrast, offset, zoom, ruler).
(2). Ability to import and export images in standard formats including: DICOM, mgh.gz, Mgh, Nii, Nii.gz, VTK, Nrrd, MRML, Seg.nrrd, Nrrd, ROi, mha.
(3). Introducing a common reference point on all modalities.
(4). Support for multimodal segmentation.
(5). Possibility to quickly estimate the volume of a region of interest.

(6). Availability of threshold-based semiautomatic segmentation.
(7). Availability of semi-automatic segmentation based on extreme points and neuron networks.
(8). Availability of semiautomatic segmentation based on neuron networks.

Thus, the physician should obtain convenient graphical tools (requirement (1)). Requirement (2) provides the doctor with the opportunity to quickly compare any available graphic materials, without being distracted by converting formats. Requirements (3) and (4) mean that the doctor has the ability to compare MRI images made in different modalities, not only integrally, but also by matching images of identical zones, which is especially important for a detailed analysis of the configuration of the area of interest. Requirements (5)–(8) provide the doctor with preliminary estimates of the most important parameters of the segmented image areas, namely the volume and configuration, which he can refine during manual segmentation.

Note that all experts chose the threshold contrast value and extreme points on the contour as the base for semi-automatic segmentation. Although these parameters do not provide the best quality of segmentation, they are physically clear to the expert and do not contradict their own context. At the same time, neural networks, as a tool for semi-automatic segmentation, although they demonstrate the highest segmentation accuracy, remain a black box for the expert; apparently, therefore, requirements (6)–(8) are stated in the list of requirements together, as a set.

The radiologists involved in the work carried out an experimental study of the applications most often used in clinical practice for segmentation and volumetry of volumetric brain formations from the point of view of fulfilling requirements (1)–(8). The results of the study are presented in Table II. A question mark indicates a function that could not be experimentally evaluated. It should be noted that part of the programs for segmentation announced on the market are not included in the comparison, since they either caused installation errors or are add-ons to other software.

TABLE II. EXPERT ASSESSMENT OF COMPLIANCE OF EXISTING APPLICATIONS AND SELECTED REQUIREMENTS

Requirement	Application				
	horos	3d sclicer	medinria	MRICRON	MITK
(1)	+	-	+	-	-
(2)	-	+	-	-	-
(3)	-	-	-	-	-
(4)	-	+	+	-	+
(5)	+	+	-	-	-
(6)	+	+	+	-	+
(7)	-	-	-	-	+
(8)	-	-	-	-	+

Thus, expert assessments made it possible to identify the requirements for applications for the segmentation of tomographic medical images from the point of view of doctors as end users, and also showed that existing programs on the market for facilitating the marking of tomographic images do not satisfy the set of identified requirements. Thus, the relevance of the ongoing development was confirmed.

IV. SERVICE STRUCTURE AND REALIZATION

A. Service architecture

To meet all of the mentioned requirement criteria, we propose an architecture in a fashion "application as a service" [34]. Since the application shall be easily accessible from any device, the core of design is based on web-technologies. Client-side is responsible for user interactions such as data representation, data segmentation as well as import/export features. Server-side consists of two main parts: rest handler from the client's request and a module with DL-module for performing segmentation algorithms.

Figure 1 shows the architecture of the application at the service level using a deployment diagram. Use of the service is carried out through any modern web browser. The client part uses Angular.js and can be run on a separate virtual machine on ServerNode. The server part can be run on a separate virtual machine and uses the Falcon webframe framework, sharpened for powerful applications in REST architecture in Python. The MongoDB database is also deployed there. In the presented implementation, client and server are designated as a single node, however this is not obligatory.

A component diagram of the service is shown in Fig. 1. The high-level contributor to the client's architecture is an Angular.js-based application with an integration of a third-party library for performing graphics manipulations by the means of X Tool Kit library. From a client's structure stand-point, the application is organised according to the Model-View-Controller pattern, whereas Model is functionating as storage, View is responsible for User Interface and its interactions, Controller stands for handling application's logic.

Server's architecture is divided into the request handler based on Python web-framework Falcon and DL-module designed upon the Pytorch library. Moreover, the server-side is also shall be regarded as a container for all the user content being uploaded as well as the user's info. As a database provider, a NoSQL MongoDB is chosen for the design. In the diagram presented, the server part is described on one physical node, although this is not mandatory.

In order to fully satisfy the list of requirements, we use the design pattern approach [35], which is illustrated by the class diagrams (Fig. 2, 3). Fig. 2 shows a class diagram of the client side, on which all the main classes involved in user interaction are highlighted. In order not to overload it, it shows only how events between Tool class and Project class are synchronized. A separate diagram (Fig. 3) shows the structure of all child (non-abstract) classes inherited from the Tool class, which explains the fulfillment of requirement (5).

Consider components of the proposed architecture (see Fig. 1–3) meeting the requirements of the above list.

To distinguish different regions of interest, the Region class is used, which contains information about the extreme points of the region, color, and the selected region itself - Area. The volume of the selected area is calculated от the server both pointwise and with accordance with an algorithm for meshed objects (see algorithm description below), the result is transferred to the client and immediately displayed. The Comments class is responsible for comments, which contains the text of the comment and the date it was created. The Scetch class is responsible for storing drawn objects - it essentially stores the line drawn with one click of the mouse button.

Interaction with selected regions and comments is carried out through the Project class, which serves as an aggregator of all additional information implemented using tools, as well as responsible for updating displayed objects. The Tool class is an abstract class from which other classes are inherited responsible for inserting additional information to the project, namely: marking points for regions, adding a comment, debugging, changing the color of regions.

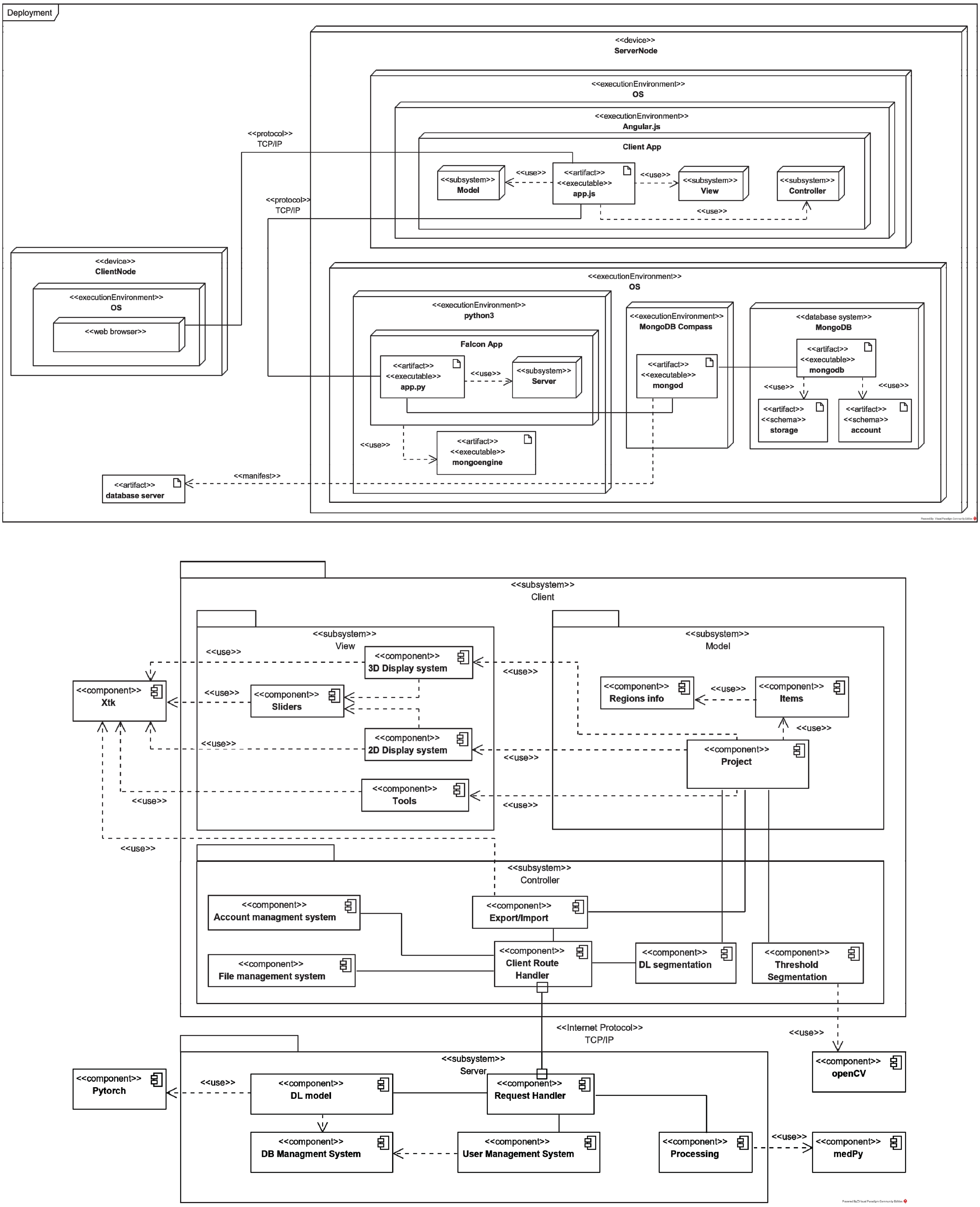

Fig. 1. Component diagram of the service

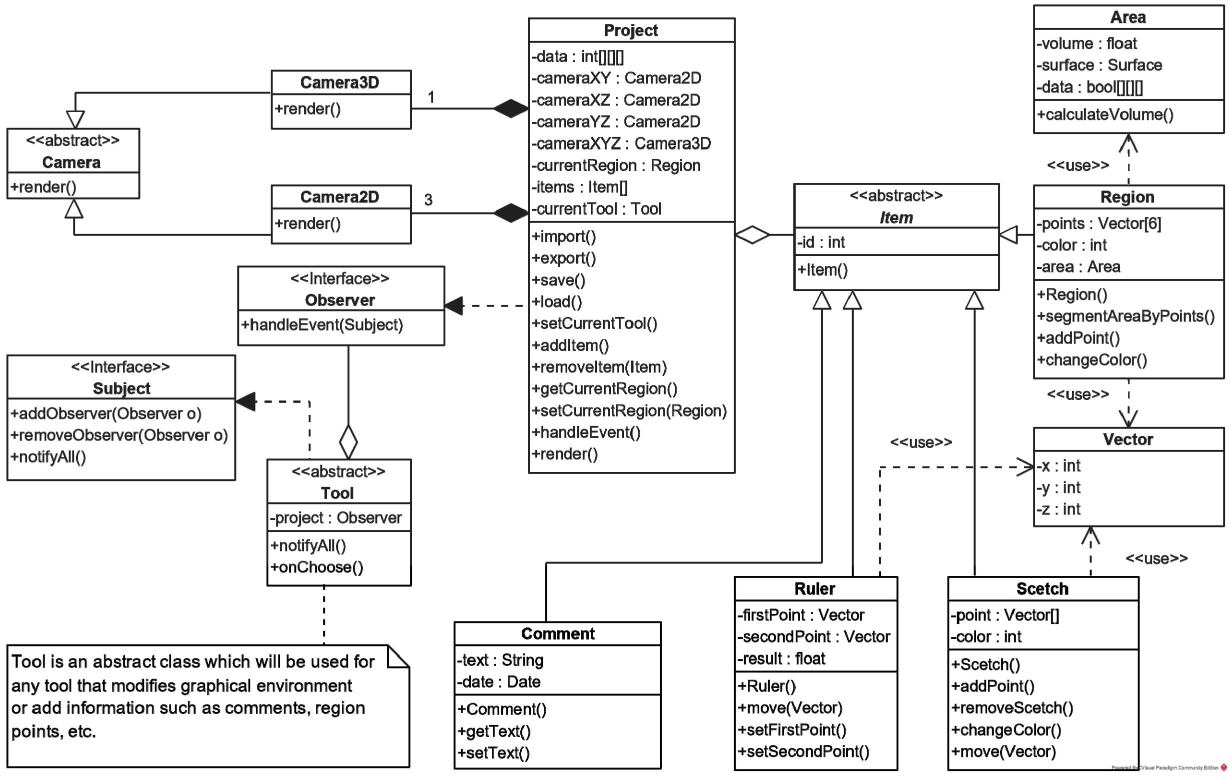

Fig. 2. Class diagram of the service

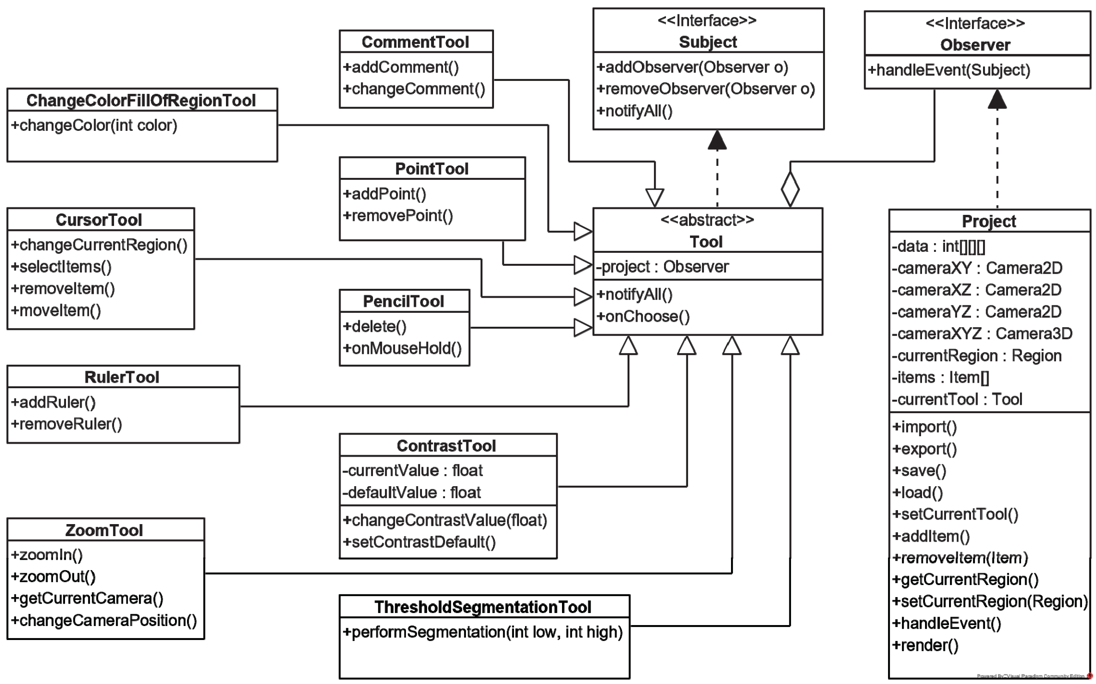

Fig. 3. Detailed class diagram of all Tools of the service

For introducing a common reference point on all modalities we use a window system, each of which is a section of the planes XY, XZ, YZ, as well as a 3D data model respectively. The Camera2D class is defined for displaying 2D planes, the Camera3D class – for 3D images, both classes being responsible for displaying a rendered object. In order to synchronize all changes, we use the Observer pattern with two interfaces. Namely, we implement the Observer interface for Project class and the Subject interface for all children of the abstract Tool class. Thus, when adding information using Tools, new objects will be rendered.

To implement the individual tools mentioned in requirement (1), we use the Team pattern and create classes inherited from the class Tool (see Figure 3):

- ChangeColorFillOfRegionTool – change the fill color of the area
- PointTool – control of setting of fixed points and segmentation
- CursorTool – cursor control, selection of objects, their movement
- RulerTool – a tool for creating objects of the Ruler class, which calculates the distance between two points
- ZoomTool – camera zoom for a specific window
- PointTool – setting points for segmentation
- PencilTool – pencil tool
- ContrastTool – manipulation with the contrast of a specific window
- ThresholdSegmentationMethod – modifier for segmenting by error-based algorithm
- CommentTool – allow user to manipulate with comments in the project

Thus, due to the selected architectural solutions of the service, the fulfillment of the requirements (1), (3) and (4) is ensured.

B. Estimation of the volume of region of interest

According to requirement (5), the service should provide the doctor with the opportunity to quickly assess the volume of the allocated area of interest. Information about the selected area is stored either as an array of pseudo-three-dimensional pixels (voxels) (voxel representation), or as an array of triangles bounding the surface of the selected area (polygonal representation). Besides, in Region class we store meta-information, including the coordinates of the six boundary points of the selected area, its color coordinates, etc.

The estimation of the volume of the selected area in the voxel representation can be done by counting the number of voxels in the array, however, this estimation is performed much faster in the polygonal representation. For this purpose the service uses the approach [36] based on the polygonal mesh, with relatively little computational complexity. Considering the region to consist of elementary tetrahedrons, we calculate each elementary volume and then add them up in accordance with the expression (1):

$$V_a = \frac{1}{6}\left(-x_3y_2z_1 + x_2y_3z_1 + x_3y_1z_2 - x_1y_3z_2 - x_2y_1z_3 + x_1y_2z_3\right), \quad (1)$$

where x_i, y_j, z_k are the coordinates of vertices of elementary tetrahedrons. The pseudo-code of the algorithm is presented below:

Algorithm 1 Volume estimation

```
def countVolume(Mesh m) :
sum = 0
for triangle in m.triangles :
  var p1 = triangle.point1
  var p2 = triangle.point2
  var p3 = triangle.point3
  sum += 1/6(− p3.x * p2.y * p1.z + p2.x * p3.y * p1.z +
p3.x * p1.y * p2.z − p1.x * p3.y * p2.z − p2.x * p3.y *
p3.y + p1.x * p2.y * p3.z)
  return sum
```

Both types of volume assessment are provided in the service; the doctor selects a specific option through the user interface.

C. Organization of the threshold-based segmentation procedure

In order to provide the threshold-based semiautomatic segmentation (the requirement (6)) we formed a three-stage pipeline:

- search for potential boundaries of objects of interest in the original tomographic image
- morphological transformation
- formation of the resulting contour

In the first stage, we use an algorithm Canny [37] consisting of the following steps:

- The original image is smoothed by 5×5 Gaussian filter.
- Gradients are searched by filtering with the Sobel kernel in horizontal and vertical directions.
- Only local maxima are marked as boundaries. To do this, each pixel is tested for a local maximum in its vicinity in the direction of the gradient
- Potential boundaries are determined by the threshold values defined by the user.

For software implementation, the Canny () method of the OpenCV library was used, which accepts the following parameters: image – input 8-bit image, treshold1 – lower threshold threshold, treshold2 – upper threshold threshold.

In the second stage, the morphological transformation is performed, which allows you to close the contours. In the developed service, a closing operation is applied, in which the resulting pixel takes the value 1 only if all the pixels of the original image under the core are 1, otherwise it will be reset to zero. The implementation uses a 3x3 kernel and two iterations of the passage of the kernel through the image.

At the final stage, the actual contour search based on the algorithm [38i] is performed. For software implementation, we use the findContours () method of the OpenCV library, which allows not only to find contours, but also to build them in a hierarchy. The following parameters are accepted at the method input: image – 8-bit single-channel image, mode - contour search mode, method – contour approximation method. In the developed system, the loop search mode cv.RETR_TREE is used, which restores the full hierarchy. As an approximation method, cv.CHAIN_APPROX_SIMPLE is used, which compresses vertical, diagonal and horizontal segments, leaving only their end points.

The pseudo-code of the pipeline developed is as follows:

Algorithm 2 Threshold-based segmentation

```
edges =OpenCV.Canny(input_image, treshold1=tr1,
treshold2=tr2)
kernel =
OpenCV.getStructuringElement(OpenCV.MORPH_ELLIPSE,(3,3))
closed = OpenCV.morphologyEx(edges,
```

OpenCV.MORPH_CLOSE, kernel, iterations=2)
contours, hierarchy = OpenCV.findContours(closed, OpenCV.RETR_TREE, OpenCV.CHAIN_APPROX_SIMPLE)

To carry out the developed pipeline in the general structure of the service, a separate tool is implemented, selected as the ThresholdSegmentationTool class in the tools class diagram (fig.2). The tool has a slider on which a doctor can specify the interval for threshold brightness values for segment selection. The image changes dynamically when the sliders change (see examples of images in Fig. 4).

D. Choice of neural network architectures for semi- automatic segmentation

As mentioned in section I and stated by the radiologists in the requirement (7) and (8), the semi-automatic approach is extremely relevant in simplifying segmentation. Several suggested solutions in non-medical applications are described in section II. To highlight the solution that is most suitable for our task, we compared two approaches of semi-automatic segmentation of tomography images based on neural networks. The first, *interactive* approach assumes that the user will perform some preliminary steps (provide the network with information about each segment to be segmented), which reduces the time to refine the results (due to the higher quality of such segmentation). The second, *non-interactive* approach does not need preliminary additional actions made by user. This is implemented as follows: the user selects the raw image, then the network segmentes all areas of interest (lesions, plaques), after which the user updates the segmentation results with standard drawing tools if necessary.

First approach is interactive and is implemented by using DEXTR [27] method of interactive segmentation, which suggests that the user provides the four extreme points (top, bottom, leftmost, rightmost) of the pursued region into the neural network. The network then fulfills a semantic segmentation of the proposed region and returns a mask of the region. The process continues iteratively until all the desired regions are segmented by the network, and at each iteration, the user puts the four extreme points on the desired region. At the final stage, the radiologist is invited to correct the results of segmentation of the neural network with standard drawing tools.

Albeit DEXTR method shows an remarkable results, it uses Deeplab-v2 [39] architecture with pre-trained resnet101 backbone, which is a rather heavyweight and resource-intensive solution. On the other hand, we use the network only for facilitating segmentation, that is the network fulfilles only the preliminary segmentation, followed by refining results by radiologist based on his own context about the subject area (for example, knowledge of the features of plaque multiple sclerosis or lesions of a brain tumor). So, this approach provides rather small inference time. But the total time spent by a radiologist on segmentating one image depends on the context (in particular, on the number of zones of interest, which he must mark with points) and can reach tens of seconds. For an experimental assessment of the effectiveness of the interactive approach, we selected several images with a different number of lesions on each, and measured the Dice coefficient after all regions of the lesion were segmented, and the time to work with each image (including the time for placing points and the inference time for calculating the segmentation).

A non-interactive approach conduct pre-segmentation of the scan with the network without forcing user to provide preliminary information about the region of interest. The user simply selects the desired image and makes a request to the system to do preliminary segmentation. Then the user provides refining of the results, if necessary, with standard drawing tools, and the process continues iteratively down to the desired result.

In order to optimize this method, we compared several appropriate network architectures in the inference time and segmentation quality. Namely, we compared several lightweight CNN architectures designed for semantic segmentation: UNet [40], FPN [41] and PSPNet [42] with resnet50 backbone. In solving the compromise between the segmentation speed and the resulting quality of the prediction mask, we adhere to the idea that the proposed tool should work quickly without long delays on the side of the backend network to provide user a smooth experience in interacting with the tool. However, worse predictions lead to an increase in the overall annotation time. To find the balance, we conducted an experiment in which we compared the aforementioned networks in terms of inference time and segmentation quality in accordance with the Dice coefficient.

In the experiment we used the dataset from the BRATS 2018 challenge [43] containing scans of glioblastoma and lower-grade glioma in the brain. For the experiment we take a random 20 patients and conducted the dataset augmentation with random crop, random flip and random rotation techniques, which resulted in the total 975 scans in the training set and 311 scans in the validation set. As a loss function we used a custom function (2) composed of BCELoss (3) and DSC coefficient (4), which in practice shows better segmentation results at the edges.

$$L = (1-\lambda)\cdot BCELoss\left(y_{pred}, y_{true}\right) - \lambda \cdot \log\left(Dice\left(y_{pred}, y_{true}\right)\right) \quad (2)$$

$$BCELoss = \frac{1}{N}\sum_{0}^{N} -w_n\left[y_n \cdot \log\sigma(x_n) + (1-y_n)\cdot\log\left(1-\sigma(x_n)\right)\right] \quad (3)$$

$$DSC = \frac{2|X \cap Y|}{|X|+|Y|}. \quad (4)$$

IV. RESULTS AND DISCUSSION

A. Results

The fulfillment of requirements (1), (3)–(5), logically justified in Sections III.A and III.B, was successfully tested directly on the software implementation.

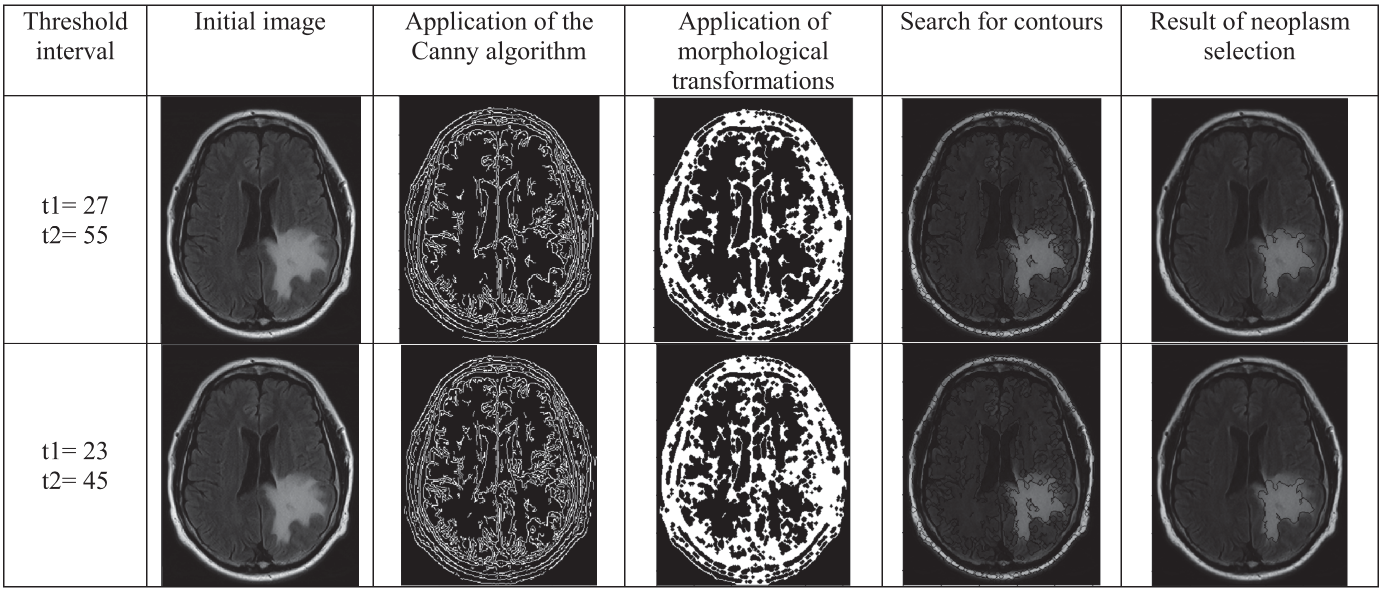

Fig. 4. Transformations of the original image with the threshold-based segmentation

To verify compliance with requirement (6), an experimental study of the developed pipeline for the threshold-based segmentation (see Section III.C) was carried out. Obviously, the key parameters here are the threshold values. Depending on them, we get a different selection of borders. Transformations of the original image obtained during the pipeline at different thresholds are illustrated in Fig. 4, and the accuracy values of the allocation of areas of interest at different threshold values, measured by the Dice coefficient, are presented in table III.

TABLE III. THE ACCURACY OF THE CONTOURING OF THE ZONE OF INTEREST BY THE THRESHOLD METHOD FOR DIFFERENT THRESHOLD VALUES

	Treshold values, t1 / t2				
	25 / 50	23 / 45	27 / 55	24 / 49	26 / 53
Accuracy	83.28047	76.6675	88.00937	80.58909	85.50029

In order to verify compliance with requirements (7) and (8), we have fulfilled the experimental procedure described of in Section III.D. The results are presented in the table IV for non-interactive approach and in Table V for interactive one.

Based on a comparison of the results of Table IV, the FPN with resnet34 backbone architecture was chosen to implement a non-interactive approach in the developed service.

TABLE IV. COMPARISON OF CNN ARCHITECTURES FOR A NON-INTERACTIVE APPROACH

Accuracy	**Average Dice metric for validation subset**	**Inference time, ms**
UNet	0.865	~ 7.6
FPN	**0.958**	**~ 5.8**
PSPNet	0.961	~ 6.9

The architectures proposed in the Table IV are the alternatives for implementing the second non-interactive approach. Based on the obtained values, we decided to choose an FPN with a resnet34 backbone as the network architecture for a non-interactive approach, as this is the best compromise between average quality and inference time. It should be noted once again that the results of such segmentation require more time for manual post-processing correction than the results of an interactive approach which could bee seen in Fig. 5.

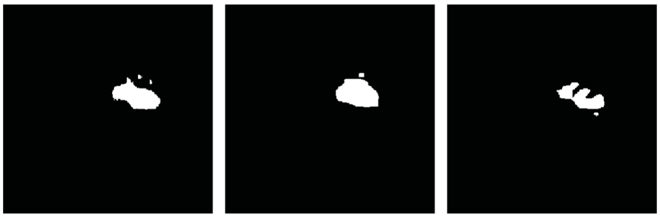

Fig. 5. Sample of human groumdtruth segmentation (left), segmentation results by interactive (center) and non-interactive approach (right)

The drawback of the interactive approach (based on DEXTR), in turn, requires a few seconds for the user to put dots in the picture, and for the network to work.

In order to show the approximate values of the speed and quality of the interactive approach as compared with the non-interactive one, we selected several images and pre-segmented each image in accordance with each approach.

In the case of non-interactive segmentation, the user should just upload a snapshot and start the network. For interactive segmentation, extreme points were plotted in accordance with the extreme lesion points on the groundtruth markup.

After all lesion regions were segmented, we measured the Dice coefficient for each result and the time to work with each image (including the time to set up the points and the time for segmentation inference) for interactive and non-interactive approaches. It should be noted that in each case only segmentation was performed without manually adjusting the segmentation results. The quantity results of the experiments are presented in Table V.

Such a considerable difference in inference time is explained by the fact that in some images there were several objects for selection, and this meant the user needed to spend more time – on each of them. In the case of a non-interactive approach, the network worked out each time in about 7.8 ms.

TABLE V. INTERACTIVE VS NON-INTERACTIVE APPROACH COMPARISON

Name of sample image (last 3 digits – slice number)	Dice coefficient for segmentation results		Inference time	
	Interacive (DEXTR)	Non-interactive (FPN)	Interactive, s	Non-interactive time, ms
Brats18_2013_4_1_100	0,8499	0,7219	30	~ 5.8
Brats18_2013_4_1_101	0,8188	0,8229	25	
Brats18_2013_4_1_102	0,8635	0,8333	14	
Brats18_2013_4_1_103	0,8945	0,8272	15	
Brats18_2013_4_1_104	0,9104	0,7640	15	
Brats18_2013_4_1_110	0,8394	0,8043	28	
Brats18_2013_4_1_111	0,8589	0,7079	30	

According to Table V, both approaches: interactive (based on DEXTR) and non-interactive (based on FPN with resnet34 backbone) have their own drawbacks and benefits. For example, in case there is a large number of objects of interest, it may be faster for the radiologist to use preliminary segmentation running the non-interactive method, and when the objects of interest are more obvious and few, then it could be more accurate and faster to segment them using the interactive approach, which allows user to get more accurate results at the stage of preliminary segmentation and spend less time on post-processing images.

Thus, the user has a choice of two alternative semi-automatic layout tools, and is free to choose the most convenient for each specific situation.Project materials regarding the neural network architectures are available at https://github.com/toshiks/dextr3d.

Fulfillment of requirement (2) encounters fundamental difficulties. As a detailed analysis of the structure of various tomographic image storage formats showed, the Raw format is common to all, but all of them differ significantly in stored metadata, which imposes restrictions on the possibility of mutual conversion between them without loss.

In this regard, an experimental study of the effectiveness of using individual formats both when importing into a developed service, and when exporting from it, has been conducted. The study used the XTK library, which has native support for some formats of interest. The results of the study are presented in Table VI.

TABLE VI. CHARACTERISTICS OF TOMOGRAPHIC IMAGE STORAGE FORMATS

Format	Acronym	Specialisation	Software/Company	Suitable Parsers
dcm, dicom	Digital Imaging and Communications in Medicine	A standard for a 2D image	The medical industry standard for creating, storing, transmitting and visualizing digital medical images and documents of examined patients	medPy, xtk
nii, nii.gz	NIfTI-1 Data Format	A common format for working with volumentric 3D images	Neuroimaging Informatics Technology Initiative	medPy, xtk
vtk	Visualization Toolkit	An unified visualization format for a software product vtk	VTK/Kitware	medPy, xtk
nrrd	Nearly raw raster data	A format and a library for manupalating with n-dimensional raster data	Open-source library / teem project	medPy, xtk
mha	ITK MetaImage	Designed for 3D images	ITK/Kitware.	medPy
seg.nrrd	Segmentation .nrrd	Designed to add information about segmented region directly into the .nrrd format	Slicer	Internal part of Slicer
mrml	Medical Reality Markup Language	A format being used to visualise and operate with graphical scenes within the application	Slicer	Internal part of Slicer
roi	Region of Interest	Stands out as a separate part of an image data	-	Is not distinguished as a separate format
mgh, mgh.gz	-	High-resolution structural data is designed by Center to satisfy its internal need in operating with high-quality images	FreeSurfer / NMR Center (at Massachusetts General Hospital)	Needed to be implemented manually / mrtrix

Based on the results of the study we have indentified formats for which adequate use in the developed service can be guaranteed - these are DCM, Nii, Nii.gz, Nrrd, VTK and mha. Formats which conversion cannot be provided with readily available means should be considered separately. Namely, the formats mgh, mgh.gz can be integrated individually with their own implementation of parsers based on the existing specification or using the existing rmatrix library, however, their connection will increase the load on the server. The roi format is not specified as a standalone format, and its use is conditionally provided. The seg.nrrd and mrml formats, due to the lack of specification and focus on the internal needs of specific software, are not recommended for use.

Service development is implemented as an open source project. Project materials regarding the service are available at https://github.com/nikolay-egorov/DEXTR3D-Service

B. Discussion

As the results of testing and experimental studies of the developed service components have shown, the adopted architectural and algorithmic solutions satisfy the requirements of the radiologist for software tools for manual segmentation of tomography images highlighting potential areas of interest.

In addition, the architectural and functional advantages of the proposed solution solution are identified.

The use of a client-server architecture with a cloud access point eliminates the need for binding to a stationary workstation and does not require an expensive and heavy software and hardware complex. This fact significantly increases the efficiency of the use of working time by a radiologist and improves ergonomic features of the service.

The functional advantages include the following:

- performance when implementing requirements (5)–(8);
- the availability of alternative methods for implementing requirements (5) and (8);
- accuracy in the implementation of requirements (6) – (8): all methods for identifying a zone of interest have shown results at a convencional level that are quite comparable with the world average results (in their classes of approaches).

V. Conclusion

The article addressed the issue of developing of software tools for manual segmentation of tomography images supporting radiologist's personal context.

With the involvement of the expert community of radiologists, we have identified the requirements for such software tools from the doctors' point of view of as end users, as well as to the nomenclature and functionality of tools that implement these requirements. In order to meet the identified requirements, we have developed a solution based on a client-server architecture with a cloud access point. The nomenclature of tools for marking tomographic images implemented in the solution, as well as the methodology for working with them, fully complies with the identified requirements.

The following functions were developed and implemented: calculation of the volume of the region of interest, as well as three options for the semiautomatic segmentation of the image based on threshold, extreme points and neuron networks. All functions have customizable parameters and (or) implementation options, which provides flexibility in solving specific markup problems.

Experimental studies have shown that the constructed service meets all the requirements put forward by radiologists and corresponds to the global level in terms of accuracy and speed (performance) of segmentation.

Acknowledgment

This work was financially supported by Russian Science Foundation, Grant 19-19-00696.

References

[1] Medical dictionary. https://medical-dictionary.thefreedictionary.com/tomography

[2] Gonzalez RC, Woods RE. Digital image processing. 3rd ed. 2008. Pearson Education.

[3] Pratt KW. Digital image processing. 4th ed. Willey; 2007. pp. 551. ISBN: 978-0-471-76777-0

[4] Sharma N., Aggarwal1L.M. Automated medical image segmentation techniques. *J Med Phys*. 2010 Jan-Mar; 35(1): 3–14. doi: 10.4103/0971-6203.58777

[5] Norouzi A., Rahim M.S.M., Altameem A., Saba T., Rad A.E., Rehman A. & Uddin M. (2014) Medical Image Segmentation Methods, Algorithms, and Applications, *IETE Technical Review*, 31:3, 199-213, DOI: 10.1080/02564602.2014.906861

[6] Kumar SN, Lenin F., Muthukumar S., Kumar A., Varghese S. A voyage on medical image segmentation algorithms. *Biomedical Research* 2018; Special Issue: S75-S87

[7] Feng Y., Kawrakow I., Olsen J., Parikh P.J., Noel C., Wooten O., Du D., Mutic S., Hu Y. A comparative study of automatic image segmentation algorithms for target tracking in MR-IGRT. *J Appl Clin Med Phys.* 2016 Mar 8;17(2):441-460. doi: 10.1120/jacmp.v17i2.5820.

[8] Withey D.J., Koles Z.J. Medical Image Segmentation: Methods and Software. 2007. Joint Meeting of the 6th International Symposium on Noninvasive Functional Source Imaging of the Brain and Heart and the International Conference on Functional Biomedical Imaging

[9] Zhao F., Xie X. An Overview of interactive medical image segmentation. *Annals of the BMVA* Vol. 2013, No. 7, pp 1–22 (2013)

[10] Kalra A/ Developing FE Human Models From Medical Images. In *Basic Finite Element Method as Applied to Injury Biomechanics*, 2018. Pp. 389-415

[11] Despotović I., Goossens B., Philips W. MRI Segmentation of the Human Brain: Challenges, Methods, and Applications. *Comput Math Methods Med*. 2015; 2015: 450341. . doi: 10.1155/2015/450341

[12] Wallner J., Mischak I., Egger J. Computed tomography data collection of the complete human mandible and valid clinical ground truth models. *SCIENTIFIC DATA* | 6:190003 | https://doi.org/10.1038/sdata.2019.3

[13] Collier DC, Burnett SS, Amin M, Bilton S, Brooks C, Ryan A, Roniger D, Tran D, Starkschall G. Assessment of consistency in contouring of normal-tissue anatomic structures. *J Appl Clin Med Phys.* 2003 Winter; 4(1):17-24.

[14] Havsteen I., Ohlhues A., Madsen K.H., Nybing J., Christensen H., and Christensen A. Are Movement Artifacts in Magnetic Resonance Imaging a Real Problem?—A Narrative Review. *Front Neurol*. 2017; 8: 232

[15] Ehman E.C., Yu L., Manduca A., Hara A.K., Shiung M. M., Jondal D., et al. Methods for clinical evaluation of noise reduction techniques in abdominopelvic CT. *RadioGraphics* 2014, V. 34, pp. 849–862.

[16] Moise A., Stella M. Designing Better Radiology Workstations: Impact of Two User Interfaces on Interpretation Errors and User Satisfaction*. J Digit Imaging*. 2005 Jun; 18(2): 109–115.

[17] Snoeckx A., Reyntiens P., Desbuquoit D., Spinhoven M.J., van Schil P.E., van Meerbeeck J.P., Parizel P.M. Evaluation of the solitary pulmonary nodule: size matters, but do not ignore the power of morphology. *Insights Imaging*. 2018 Feb; 9(1): 73–86.

[18] Liu J., Li M., Wang J., Wu F., Liu T., and Pan Y. A Survey of MRI-based brain tumor segmentation methods. Tsinghua Science and Technology. Volume 19, Number 6, December 2014. Pp. 578-595

[19] Ruggerio S., Weisser G. Integrating Mac Systems into a Medical IT Infrastructure: Creating an affordable radiology workstation with OsiriX. https://www.osirix-viewer.com/pdf/IntegratingMacSystems.pdf

[20] Horos: Horos Project – Free DICOM Medical Image Viewer https://horosproject.org/

[21] Mehrtash A., Pesteie M., Hetherington J., Behringer P.A., Kapur T., Wells W.M. III, Rohlinga R., Fedorov A., Abolmaesumi P. DeepInfer: Open-Source Deep Learning Deployment Toolkit for Image-Guided Therapy. Proc SPIE Int Soc Opt Eng. 2017 February 11; 10135: . doi:10.1117/12.2256011.

[22] Milletari F., Frei J., Ahmadi S.-A. TOMAAT: volumetric medical image analysis as a cloud service. arXiv:1803.06784v2 [cs.CV] 25 Apr 2018

[23] Philbrick K.A., Weston A.D., Akkus Z. et al. RIL-Contour: a Medical Imaging Dataset Annotation Tool for and with Deep Learning. J Digit Imaging (2019) 32: 571. https://doi.org/10.1007/s10278-019-00232-0

[24] Guttmann C,, Pinzon A.M. Marciniak A. Implementation of manual and smart semi-automatic brain lesion segmentation tools in SPINE. https://cni.bwh.harvard.edu/implementation-manual-and-smart-semi-automatic-brain-lesion-segmentation-tools-spine

[25] Moreno-Dominguez D. Manual & semi-automatic lesion-mask editing (Lesion Cleanup). https://support.qmenta.com/hc/en-us/articles/360007221051-Manual-semi-automatic-lesion-mask-editing-Lesion-Cleanup-

[26] Sakinis T., Milletari F., Roth H. et al. Interactive segmentation of medical images through fully convolutional neural networks. https://arxiv.org/abs/1903.08205
[27] Maninis K.-K., Caelles S., Pont-Tuset J., Van Gool L. Deep Extreme Cut: From Extreme Points to Object Segmentation. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 616–625.
[28] ISO 9241-11:2018(en). Ergonomics of human-system interaction — Part 11: Usability: Definitions and concepts. *International Organization for Standardization*, Geneva, Switzerland, 2018.
[29] What radiologists want in a diagnostic workstation workspace. *Appl Radiol.* By Staff News Brief| February 07, 2018.
[30] Sharma A, Wang K, Siegel E. Radiologist Digital Workspace Use and Preference: a Survey-Based Study. J Digit Imaging. 2017 30;6:687-694.
[31] Iannessi A., Marcy P.-Y., Clatz O., Bertrand A.-S., Sugimoto M. A review of existing and potential computer user interfaces for modern radiology. Insights into Imaging. August 2018, Volume 9, Issue 4, pp 599–609
[32] van Ooijen P/M.A., Koesoema A.P., and Oudkerk M. User Questionnaire to Evaluate the Radiological Workspace. *J Digit Imaging*. 2006 Dec; 19(Suppl 1): 52–59.
[33] ACR–AAPM–SIIM technical standard for electronic practice of medical imaging / The American College of Radiology / Revised 2017. https://www.acr.org/-/media/ACR/Files/Practice-Parameters/ Elec-Practice-MedImag.pdf
[34] Columbus L. 83% Of Healthcare Organizations Are Using Cloud-Based Apps Today. Jul 17, 2014. Availible from: https://www.forbes.com/sites/louiscolumbus/2014/07/17/83-of-healthcare-organizations-are-using-cloud-based-apps-today/#487db5b23b72
[35] Gamma E., Johnson R., Helm R., Vlissides J. Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley Professional, 1994. 416 p.
[36] Zhang C., Chen T. Efficient feature extraction for 2D/3D objects in mesh representation. Proceedings 2001 International Conference on Image Processing, Vol. 3, p. 935–938.
[37] Canny J. A computational approach to edge detection, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679–698, 1986.
[38] Satoshi Suzuki et al. Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing*, 30(1):32–46, 1985.)
[39] Chen L.-C., Papandreou G., Kokkinos I., Murphy K., and Yuille A. L., Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2017.
[40] Ronneberger O., Fischer P., and Brox T. U-net: Convolutional networks for biomedical image segmentation. *International Conference on Medical image computing and computer-assisted intervention*, Springer, 2015, pp. 234–241.
[41] Lin T.-Y., Doll´ar P., Girshick R., He K., Hariharan B., and Belongie S. Feature pyramid networks for object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2117–2125.
[42] Zhao H., Shi J., Qi X., Wang X., and Jia J. Pyramid scene parsing network. Proceedings of the IEEE conference on Computer Vision and Pattern Recognition, 2017, pp. 2881–2890.
[43] Bakas S., Reyes M., Jakab A., Bauer S., Rempfler M., Crimi A., Shinohara R. T., Berger C., Ha S. M., Rozycki M., et al. Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the brats challenge. arXiv preprint arXiv:1811.02629, 2018.