

MATHEMATICAL METHODS FOR IMAGE PROCESSING AND MATHEMATICAL MODELING

MÁRIA ŽDÍMALOVÁ ¹, MARIÁN VRÁBEL ², KRISTÍNA BORATKOVÁ ¹,
AND BPB MED S.R.O.

Abstract. This contributions deals with applied mathematical models and its application into image processing. We focus on discrete segmentation mathematical algorithms and its application in the binary segmentation of images. We improve, implement and create new own software's tools for image analyses specially graph cut and grab cut method. We optimize these algorithms and bring concrete examples in real life, biology, medicine and civil engineering.

1. INTRODUCTION

This contribution deals with the proceeding of the image data and the mathematical approach for processing and analyses of image data, [17], [11]. We present medical, biological, and engineering data. Mathematical modeling is a very good and strong tool for the analysis and modeling of this data. Consequently, mathematical modeling brings a good tool, enumeration, and visualization perspective if representing different types of image data. We discuss some methods for medical image analyses. The main point brings perspective from discrete mathematics and discrete algorithms, dealing with graph theory, discrete mathematics, statistics, aggregation functions for image data analyses. We focus on the analyses of the techniques, using image processing and segmentation of the detected objects. Clinicians can use segmentation results to identify abnormal cells, object, pathological items like tumors or cysts and improve therapy planning. The creation of high-quality labeled and annotated data sets are a critical part of achieving the algorithmic goal of automated medical image segmentation.

2010 *Mathematics Subject Classification.* Primary: 68, 94.

Key words and phrases. image processing, segmentation, graph cut, grab cut, aggregation method, applied mathematics, mathematical modeling .

We focus on different approaches towards processing of image data focused on image segmentation. We study concrete through the discrete mathematical algorithms, [8], the algorithms from graph theory. We study in this paper mainly algorithms for finding the maximal flow and minimal cut in a flow network. [4, 5, 6]. The focus is on the analysis of artificial and real data. We present general, biological, medical and engineering data. A part of this contribution will be theoretical and application part where we build own working software's tools for image segmentation.

Image segmentation is a method used in digital image processing to divide an original image into multiple parts based on common characteristics, [11]. The goal of segmentation is to identify objects in the image and separate them from the background. Image segmentation groups pixels into regions that have similar characteristics, such as color intensity or similar textures and others features. There are several general algorithms and techniques for image segmentation. In this work, we focus on image segmentation using discrete graph algorithms, specifically using the Edmonds-Karp algorithm with breadth-first scanning, [7, 10]. The Edmonds-Karp algorithm can also be applied to image segmentation. In this case, the image is represented as a graph where each pixel is a vertex and the edges between neighboring pixels determine the similarity between them. The algorithm finds a cut in the graph that splits the image into object and background. [13, 18, 19]

2. OVERVIEW OF THE KNOWN METHODS

1. Thresholding

It is the simplest technique of all image segmentation methods. This method is constructed on a threshold value, used to transform a grey-scale image into a binary image. The basic objective of this technique is to select a single threshold value. Thresholding is one of the broadly used methods for image segmentation. It is useful in discerning foreground from the background. By selecting suitable threshold value T , the grey level image can be transformed to binary image. The binary image should contain every important information about the position and shape of the objects of interest (foreground). The advantage of gaining first a binary image is to reduce the complexity and image classification. The best way to convert a grey-level image into a binary image is to select a single threshold value T . Then every grey level values below this T will be classified as black, and those above the value of T will be white. The segmentation problem depends on selecting the proper value for threshold T . In actual applications histograms are more complex, with various peaks and not clear valleys. One disadvantage of the method is that it is not always easy to select the value of T . [1, 16, 21]

2. Clustering methods

In these methods, distance between a pixel of an image and a cluster center is used for clustering. The difference is naturally based on certain properties such as colour, intensity, texture and location of a pixel or a weighted grouping of these factors. For these clustering we use parameter K and this parameter K is selected randomly, manually or by heuristic. This method is positive to converge, but it may not yield the optimal solution. The quality of the solution depends on the

preliminary set of clusters and also on the values K . The corresponding K -means algorithm is an iterative method which is used to divide an image into K clusters or segments, [16, 20].

3. Histogram based methods

These techniques are among the simplest and most commonly used methods. These methods are very skillful when compared to other image segmentation approaches because they normally require only one pass through the pixels. During this method, a histogram is calculated for all pixels in the image. To cluster the image into regions, gray levels are chosen using the histogram. For general image there are two attributes: the object and the background. The object of the image is one grey level which is the smaller peak in the histogram. The background of the image is another grey level which represents the large peak in the histogram. The peaks and valleys of the histogram are used to trace the clusters of the image. Colour or the intensity can be used as a measure. This technique can be improved by applying the histogram-seeking method recursively to clusters in an image with a purpose to divide them into smaller clusters. This procedure is repeated with smaller and still smaller clusters until no further clusters are formed [1, 21]. The disadvantage of the histogram-seeking technique is that it can be difficult to identify significant peaks and valleys in the image. This method segments an image, based on active objects and on situations, following in a different type of segmentation which is useful in Video tracking.

4. Split-and-merge methods

This segmentation method is based on a quad-tree partition of an image. Therefore it is sometimes called quad-tree segmentation method. In this method, an image is represented as a tree which is a connected graph with no cycles. The technique begins at the root of the tree. If it starts with non-uniform (not homogenous), the split and merge algorithm has two phases: the split and the merge. In the split phase we recursively split regions into four regions (starting with the whole image as one region) if anticipation of our homogeneity criterion is met in all subregions. Conversely, if four son-squares are identical (homogenous), then they can be merged as some connected components. This process is called the merging process. The segmented region is the node of a tree. This process (splitting and merging) is continued recursively so that no further splitting or merges are possible [1, 16, 21]. The process can be summarized into two different algorithms, i.e. splitting and merging algorithms.

5. Watershed transformation

It is the simple image segmentation technique in which the gradient is treated as the magnitude of an image as a topographic surface. Those pixels which have the highest gradient magnitude intensities, correspond to watershed lines that correspond boundaries of regions. Water placed on any pixels of an image bordered by a common watershed line, flows downward to a common local intensity minimum. Pixels draining to a common minimum form a catch basin, which characterise a segment of an image. Advantage of this technique is that it divides the segmentation process into two separate steps. First detecting the main edges of the image,

and then computing the watershed of the detected gradients. This method has many advantages, particularly in real life applications, see [1, 16, 21].

6. Segmentation using Neural Networks

This segmentation method [1, 16, 21], trusts on processing small areas of an image using an artificial neural network or a set of neural networks. Afterwards the decision-making process labels the parts of an image accordingly to the group recognized by the neural network. A type of network is designed especially for this is the Kohonen map. Pulse coupled neural networks (PCNNs) are the neural network models proposed by modelling a cat's visual cortex and developed for high-performance biomimetic image processing, see [1, 21, 16].

7. Region growing methods

This method is based on an expansion of an object detected inside of a given object. The main idea is in choosing object seed pixels (inside an area to be detected) and then we are searching for neighbouring pixels with similar intensities as has the object seed pixel (in 4 or 8 different directions). The algorithm continues to run until no pixel remains that is comparable to any other pixel that remains similar. The entire thing is then found, see [1, 16, 21].

The criterion for adding a neighbouring pixel to the area is mostly the value of intensity, comparing with the intensity of the object seed pixel. There should be some tolerance towards to lower or higher values. Therefore the intensity of the object seed pixel is from the closed interval $[a, b]$, where a is the accepted intensity value of the neighbouring pixel and b is the highest. This method is suitable for images containing a noise.

8. Fuzzy approach

Fuzzy approaches for image segmentation can be categorized into four classes: segmentation via thresholding, segmentation via clustering, supervised segmentation and rule-based segmentation. These methods are divided into fuzzy integral based segmentation approaches, fuzzy rule, fuzzy geometry, fuzzy thresholding, and fuzzy clustering by Tizhoosh. A large amount of research work has been carried out in this area. A membership function of the image intensity value was designed to minimize a global fuzzy measure, and based on the optimal membership function, the object could be segmented from the background. A fuzzy thresholding scheme was proposed in [1, 16].

9. Methods based on regions

Unlike the methods presented above, in [16], these methods do not seek border bounding regions but the regions themselves formed by a set of pixels. These methods are appropriate in cases of noisy pictures when it is hard to find the boundaries between objects. The basic idea of these methods is the division of the image into regions with the greatest possible homogeneity; the homogeneity criteria may be based on the level of grey, color, and texture, see [1, 21].

10. Methods based on a comparison

They are used to locate objects that are similar to the pattern designated by the users. In [1, 16, 21], the correlation between the object and the pattern is determined by the optimal criterion based on the characteristics of the object.

The value of optimal criteria is calculated for each position and rotation of the pattern in the image.

11. Active contour models

We place an initial curve around the object that we wish to segment, [1, 21]. The curve is then deformed in order to minimize the energy functional. This functional is formed as the sum of internal and external energy. The internal energy is designed by the shape of the curve. A smooth curve without significant changes shall take the form of low value.

Additionally, segmentation techniques can be divided into:

- 1) Automatic methods
- 2) Interactive methods.

Automatic methods do not require the user interaction during the process of segmentation. The segmentation by these methods is, therefore, faster and more convenient. However, these methods do not always give the desired results, as in the case of medical data in which objects are ambiguously defined or contain strong noise. Interactive methods can overcome this problem by involving humans in the process of segmentation. The disadvantage is a long segmentation time and the need for participating of a person in the process of segmentation [21].

12. The edge-oriented methods

They are based on edge detection using edge detector. These edges mark places where there is a change in the level of grey, color, and texture. The most common problem with this approach, due to noise or incorrect information in the image, is finding the edges in the inappropriate locations or absence of edges in the places where they should be located. [1, 21]

3. ALGORITHMS BASED ON SEARCHING MAXIMAL FLOW OF THE NETWORK

3.1. Maximum flow. The maximum flow in the network is sought using an algorithm that incrementally increases the value of the flow in the network until the maximum flow is reached, [10, 19, 12]. In this thesis, we will use the Ford - Fulkerson algorithm to find the maximum flow. The algorithm starts with all flows in the network set to 0. It then interactively searches for increasing half-paths from source s to estuary t and updates the flow at these half-paths to increase the total flow in the network. The search for an increasing half-path can be done in different ways, for example, by searching the graph in width or in depth. At each iteration of the algorithm, an increasing half-path is searched for and the flow at the edges of the half-path is updated. The algorithm repeats this process until there is no increasing penumbra from source to mouth. At this point, the flow going through the network is maximum. The algorithm is terminated and the maximum flow $F(f(e))$ is moving through the network is returned. The value of the maximum flow in the network is important for various applications, for example, in modeling flows in transport networks, electrical networks, or in image segmentation. The

Ford-Fulkerson algorithm is one of the efficient methods for finding the maximum flow in a network.

The "flow" can be thought of as the amount of some quantity flowing through the arrows of a network. For example, the flow of water in a pipe or cars in a transportation network. In our case, it will be the amount of a quantity transported from a source S toward a T output along the edges. We say that a flow is maximum if its value is maximum among all flows in the network:

$c(e)$ - arrow capacity, $f(e)$ - flow at the edge, $r(e)$ - edge reserve.

Each edge in the network has its capacity $c(e)$, which is the maximum volume that can be transmitted over a given edge. The reserve $r(e)$ of an edge represents the amount of volume that can still be carried along that edge. The reserve is calculated as the difference between the capacity of the edge and the flow already flowing down the edge. To maintain physically meaningful flow properties, the edge flow $f(e)$ must be non-negative and also less than or equal to the capacity of the edge:

$$0 \leq f(e) \leq c(e), e \in E(G).$$

Further, Kirchhoff's law holds for the flow. Simplistically, Kirchhoff's law states that the same amount of flow must flow into a vertex v as flows out of it. It holds for all vertices of the network except the source S and the mouth T . It also holds that as much flows out of the source as flows into the mouth.

3.2. Network. We construct a graph $G = (V(G), E(G), c(e))$, where $V(G)$ is a set of vertices and $E(G)$ is a set of edges, [2, 19]. Every edge is assigned a non-negative cost $c(e)$ (capacity). The vertices correspond to pixels p in $V(G)$. We add two new vertices, an „object vertex“ (source, input S) and a background vertex (sink, output T), i.e., two extra vertices called terminals.

The set of edges E consists of two types of un-oriented edges:

N -links (neighbouring links) and T -links (terminal links). Thus every pixel p has two T -links $\{p, S\}$ and $\{p, T\}$ connected to every terminal. Every couple of neighbouring pixels $\{p, q\}$ in the grid is connected by N -link. An arbitrary N -link connecting neighbours p and q , is denoted by $\{p, q\}$. Thus it holds $E(G) = N \cup \{\{p, S\}, \{p, T\}\}$. The cut is a subset of edges $C \subset E(G)$ such that the terminal nodes will be separated on an induced graph $G(C) = (V(G(C)), E(G(C)) \setminus C)$. The cost c of the graph is defined as the sum of edge weights (cost) as follows:

$$|C| = \sum_{e \in C} c(e).$$

To find the minimal $S - T$ cut we used the result of Ford and Fulkerson [3] saying that the maximal flow from S to T saturates the set of edges in such a way that it corresponds to the minimal cut (i.e. to minimal $|C|$)—the minimal $S - T$ cut and maximal flow are dual problems. This fact is based in the well-known "Max-flow Min-cut theorem", see [10].

3.3. Graph Cut Method. In the following we introduce the notation and terminology we use later. We transform picture with pixels and intensitiies into graph /network structure. We recall that the links connecting two neighbouring pixels

(vertices) p and q are called N -links. Links connecting a pixel with a terminal or T are called T -links (see Fig. 1 a) and b)) where a pixel is depicted as a grey cube, N -links as horizontal lines and T -links as vertical lines) see Fig.1.

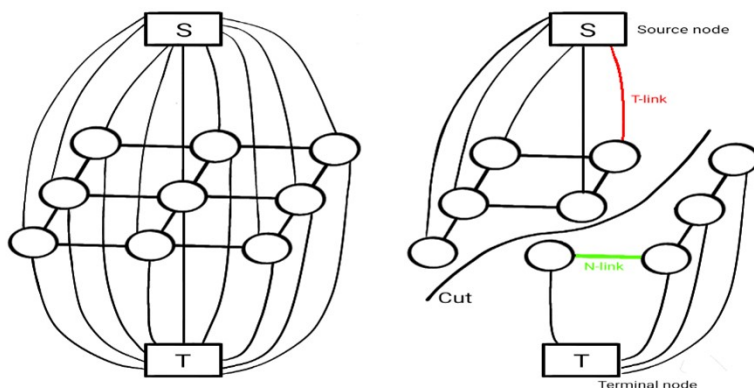


FIGURE 1. Graph Cut Method

The next step is to evaluate the edge capacities of this graph with respect to our demand that the minimum $S - T$ cut divides nodes into an object segment and a background segment. We use the following notation, see [13, 19]:

- P is the set of all pixels,
- (p, q) is the link connecting neighbouring pixels p and q ,
- I_p is the value of the intensity of the pixel p ,
- M is the maximal value of the intensity of the pixel (of the responsible figure),
- D the difference of the maximal and minimal value of the intensity of the pixel (of the responsible figure),
- O_{avr} the average value of the intensity of object seed pixels,
- B_{avr} the average value of the intensity of the background seed pixels,
- $S(p)$ the capacity of the edge(link) connecting the initial vertex s and corresponding pixel p ,
- $T(p)$ the capacity of the edge (link) reaching output output vertex t and concrete pixel (the vertex p),
- $N(p, q)$ the capacity of the edge (link) reaching neighbours pixels p and q ,
- λ the weighing constant.

The weighing constant λ enables us to modify the result of segmentation. If λ is chosen to be low, we prefer the boundary property. We usually get fewer objects with smooth borders. By choosing a large λ , i.e., $\lambda \in (1, \infty)$ we prefer the regional property resulting in more objects with unsmooth border. Usually, at the beginning we select $\lambda = 1$ and then we modify it to get a desirable result. $N(p, q)$ expresses the relationship between intensities of p and q , $S(p)$ and $T(p)$ express the relationship between intensity values of pixels and the values O_{avr} and B_{avr} .

N -links connecting links with similar intensities have their capacities greater than those with larger intensity differences. Similarly, T -links, connecting the terminal T and a pixel express the difference between a pixel and O_{avr} : the higher the difference the smaller the capacity. Likewise the capacities of T -links depend on a “similarity” of a pixel to B_{avr} .

If we add direction to an edge, we get the arrows. Then we orient the edges as follows: In the grid we add two arrows to every edge, vice versa. On the N -links we add orientations from the vertex N toward the grid, and from the vertex S -toward to the grid. All calculation of capacities and counting of the maximal flow in such network follows the same steps as before.

3.4. Implementation of Graph Cut method. We implemented graph cut algorithm and its optimization in C++ programming language. We describe some concrete libraries, and function for implementing our program.

The `on_btn_image_clicked()` is the function which is responsible for retrieving the supported image from the user’s device. The loading is performed using standard QT library functions. The application can handle image files with the following formats: `.png`, `.jpg`, `.jpeg` and `.jif`.

Among the most important functions of the code itself is the `createGraph()` function. This function represents the input image data as a grid. Each pixel of the image is represented as a vertex of the mesh and is stored in both a two-dimensional and a one-dimensional vector of class `myPixel`, which represents the vertices of the mesh. Each vertex of the `myPixel` class contains color information extracted from the input image. The vector of `nlinks` vectors contains the stored n link classes based on the n vertices of the network. For a vertex v_i , the `nlinks[i]` vector stores the `nlinks` to its neighbors, containing information about the capacity, margin, and indices of the vertices it connects.

Further, S -links and T -links are stored in `slinks` and `tlinks` respectively, containing information about their capacity, reserve and vertex index. The edge capacities are computed according to the literature. The implementation of these functions and classes is crucial for correctly representing the input image data as a network and assigning the correct capacities to each edge. It is also important to correctly manipulate the vectors and access the individual vertices and edges of the network. Proper implementation and calculation of capacities is essential for the subsequent processing of image segmentation using the Edmonds-Karp algorithm. The beginning function sets all vertices of the network as non-circular.

The main and most important function is `maximumFlow()`, which is responsible for finding the maximum flow and then highlighting the pixels of the object. Initially, the function sets all vertices of the mesh as undiscovered using the `uncheckAll()` function. The vector `foundPixels` is denoted with -1 values and will be successively populated with the indices of the vertices reached. It is obvious that the shortest paths in the network will be of length two, i.e. $s - v_i - t$, where v_i is a vertex in the network. Therefore, we search these shortest paths before the main loop to speed up the solution. The main loop of the function runs until the

maximum number of iterations $maxIter$ is reached or until no increasing half-path from source s to mouth t has been found.

After finding the destination vertex, the function reconstructs the path backwards and calculates the value of the minimum capacity of this path. In the loop, the coordinates of the predecessor of the $nLinkTo$ vertex are sequentially searched and their coordinates are stored in the auxiliary variable $temp$. When a predecessor with coordinates $[-1][-1]$ is encountered in the loop, we know that we have reached the source vertex. In this way, we have successfully reconstructed the entire path, whose coordinates of the vertices defining its arrows are stored in the variable $temp$.

When the maximum number of iterations has been reached or if no increasing half-path has been found, the loop is terminated. At this stage, the colors of the reached pixels of the *object* are changed to visually match the object. The loaded image is modified in the application, allowing the user to see the resulting augmented image.

At first traverses all S -lines (s, v_i) and records all reachable vertices in the $foundPixels$ vector. These vertices are marked as found and their $nLinkTo$ attribute is set to coordinates $[-1][-1]$, which will represent the coordinates of source s . This notation is important for later path reconstruction. We then traverse all the n -links between the vertices of the network. We search for undiscovered reachable neighbors of the found pixels and store their indices in $foundPixels$ and mark them as discovered. The ancestor coordinate of a given vertex is stored in the $nLinkTo$ attribute. This procedure continues until we get from some vertex to the mouth of t , which means that the T -link margin from the vertex under study is non-zero.

3.5. Grab Cut Segmentation Technique. Grab Cut technique is used as a part of refining process in an initial user image segmentation between foreground and background. The Grab Cut technique was first created at Microsoft Research in Cambridge, United Kingdom. The effort for the user is as follows: The user selects an image to perform the image segmentation and draws a rectangle around the object of interest that should be segmented. Grab Cut Technique provides interactive assistance to consumers and users. The user needs to input the very rough segmentation only between foreground and background. Normally this is done by drawing a rectangle around the body of interest.

Grab Cut is an image segmentation method based on graph cuts. Starting with a user-specified bounding box around the object to be segmented, the algorithm estimates the color distribution of the target object and that of the background using a Gaussian mixture model, Fig. 2. It is an iterated and mixed method.

Grab Cut needs the user to provide a bounding box to segment an object. After getting an initial segmentation, the user can provide scribbles for refinement. Graph Cut needs the user to provide a set of scribbles for the foreground and background to segment an object. Refinement is also allowed by giving more scribbles. The algorithm used for foreground extraction here is Grab Cut Algorithm. In this algorithm, the region is drawn in accordance with the foreground, a rectangle is

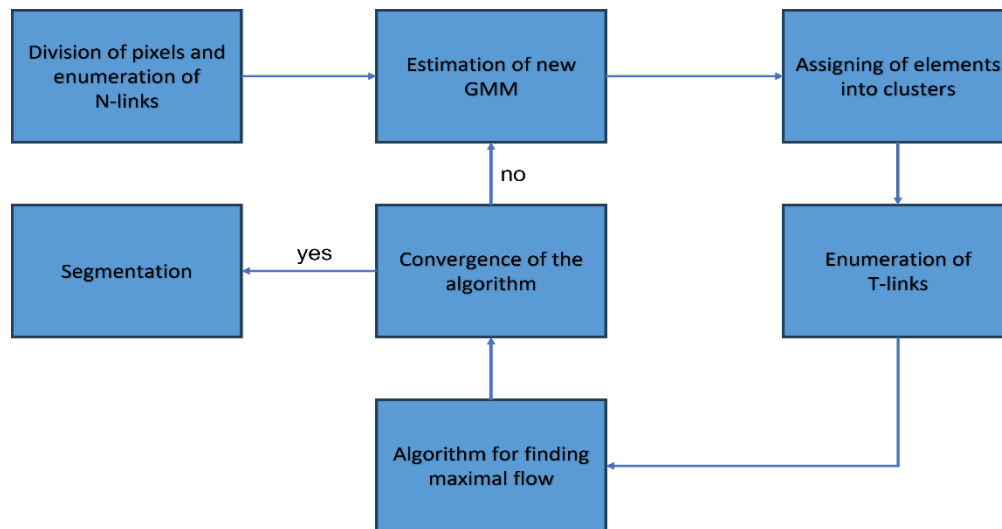


FIGURE 2. Grab Cut Method

drawn over it. This is the rectangle that encases our main object. The region coordinates are decided over understanding the foreground mask. [8, 20, 14].

4. RESULTS

We implemented the above mentioned algorithms and created our own software tools for image segmentation. We used C and C++ programming language and created own software for special request of people From genuine prax calls like from biologist, clinical doctors and civil engineering This unique program has the benefit of being made based on specific requests and duties from actual real calls. Free online software were not able to make segmentation of good quality or reply some special questions about percentage of cells on the data and others.

4.1. Results of Graph Cut. In this subsection we present concrete results from our software using implemented and optimized Graph Cut Segmentation method.

Data proceed in this chapter comes from [9, 14, 22]. We proceed pictures by own software tool with implemented Graph Cut algorithm and its optimisation.

In the Fig. 3 and Fig. 4, we present examples of the global segmentation provided by own software tool based on graph cut. We can observe comparison of two different results of segmentation provide by two different choosing of λ , see Fig. 4.

In the Fig. 5 we see original picture, choosing of the object and background pixels and its segmentation.

In the Fig. 7 and 8 we observe graph cut segmentation and its global segmentation and examples with different choosing of λ .



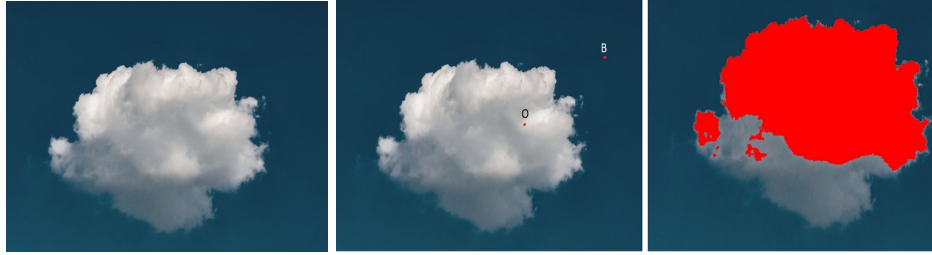
FIGURE 3. Original picture and its segmentation.



FIGURE 4. Comparison of the segmentation with $\lambda = 1$ and $\lambda = 50$

4.2. Results applied in medical image analyses. In this section we proceed data with our own software tool with implemented Grab Cut techniques. We search on the brain MRI scan tumors and we segmented them into solid and compact objects with sharp boundaries, see Fig. 9 and Fig. 10. Data comes from [23].

4.2.1. *Graph Cut.* In this part we segment image real data from sono graph/the diagnostic of the some clusters of the breast. As we can see, we used the method of graph cut and we tried to define the segmented clusters in the image sample. Results of the segmentation are obtained by the red color. For this kind of sample Graph Cut Technique was chosen for segmentation. This technique is more suitable for this kind of "breast" analyses, see Fig. 11.



(A) Original picture (B) Object, background pixels; The segmentation result

FIGURE 5. Original picture and its global segmentation

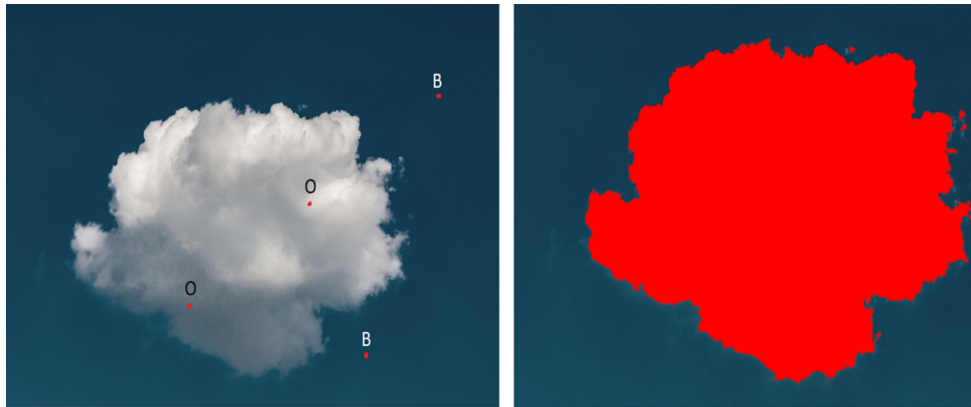


FIGURE 6. Different choosing of object and background pixels

4.2.2. *Statistical analyses.* In this section we provide statistical proceeding of the numerical data of the BPB Med Radiology s.r.o., Strečnianska, Bratislava, Slovakia. The Radiology BPB Med s.r.o. provided to use statistical data, devoted to the number of anonymous patients who come to x-ray or ultrasound/sonography. We can see results in the Fig. 12, till Fig. 16. Data set is provided by anonymous patients, males-M, female-F and their visiting of the BPB Radiology clinic, for roentgen and sonography/ultrasound treatment. Data are for the 2019, 2020, 2021, and 2022 years. Data set represents number of patients taking x-ray and ultrasound treatment during provided years. We can observe and conclude that numbers of patients are increasing, and there is still lack of enough diagnostic medicine machines for image diagnostic in our country.

In the Fig 12 we present data provided for the years 2019-2022. We see comparison for every year for females/F and males/M.

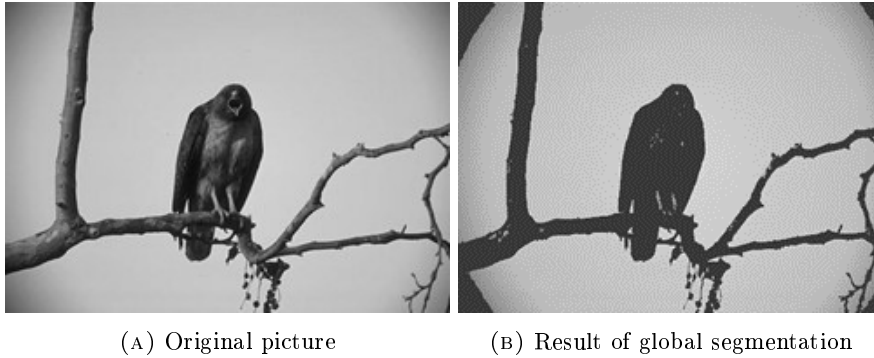


FIGURE 7. Graph cut segmentation

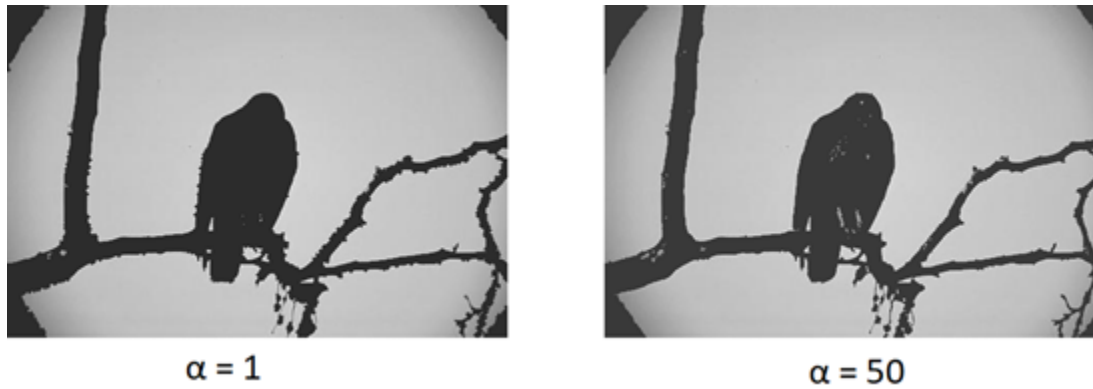


FIGURE 8. Different results of segmentation by different parameters α

In the Fig. 13 and Fig. 14 we present statistical analyses for concrete years from 2019 till 2022 for X-ray diagnostic treatment. Analysis is done for males/M and females/F in the BPB Radiology BPB Med. s.r.o.

Analogically, in the Fig. 15 and Fig. 16 we present statistical analyses for concrete years from 2019 till 2022 for ultrasound diagnostic treatment. Analysis is done for males/M and females/F in the BPB Radiology BPB Med. s.r.o. All data, numerical even visual, were anonymized before their processing.

It is easy to see that the total amount of patients has been growing in the corresponding years. There appears an increasing request for ultra-sound and x/ray diagnosis by patients. Therefore it has sense to study image analyses of images in roentgen, sono, ultra-sound and also other CT and MRI images. These kind of diagnostic approach has a big potential in to the future and improving of the data analyses of visualizing data set.

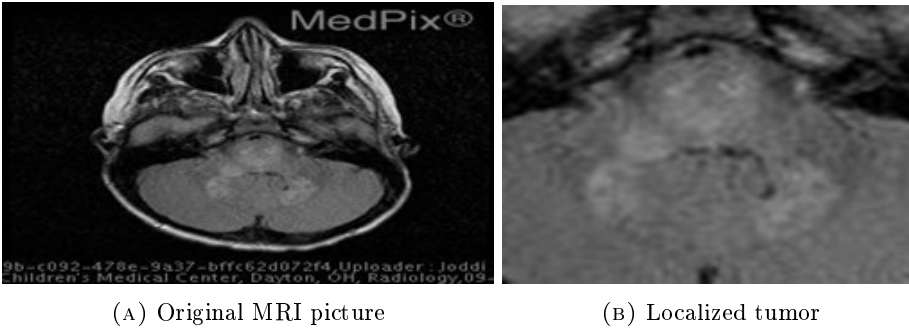


FIGURE 9. Original data and zoomed tumor

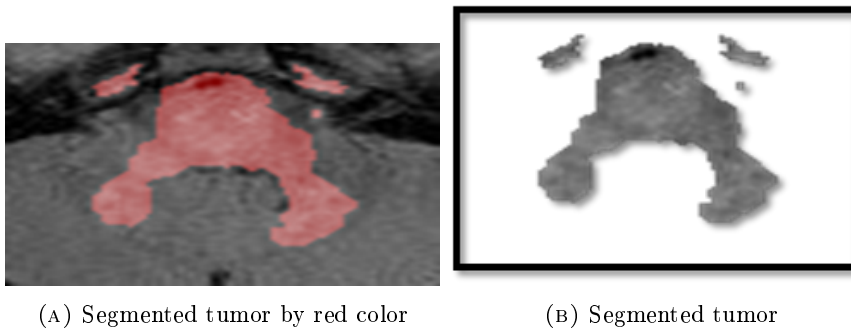


FIGURE 10. Segmented results

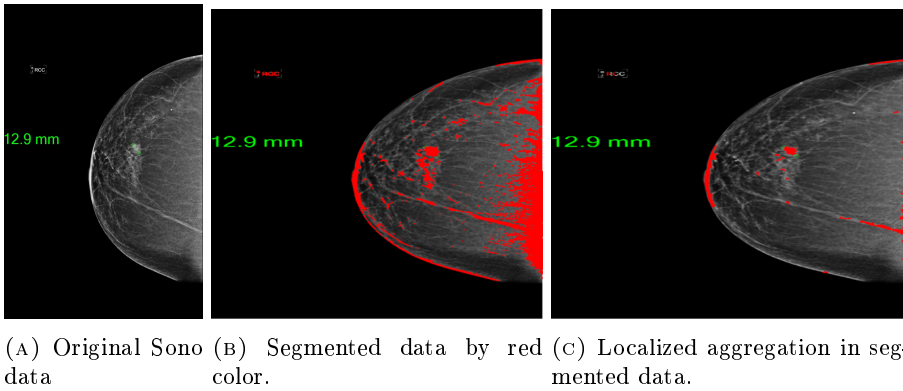
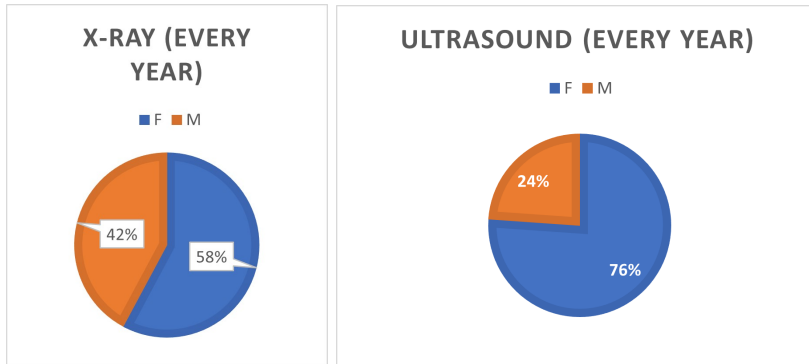


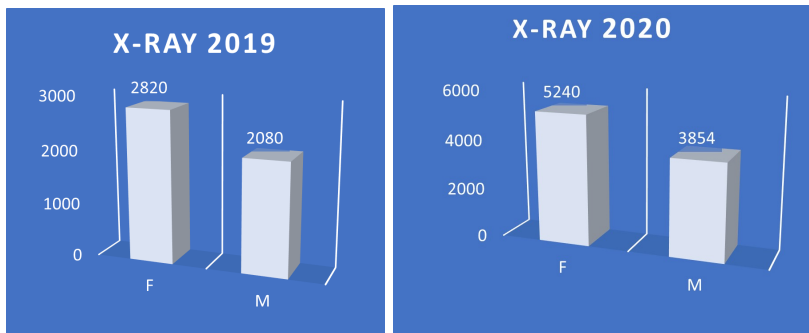
FIGURE 11. Ultrasound "breast" data analyses.

4.3. Civil Engineering results. We use the Graph Cut Method to analyze civil engineering data. We study cracs in different technical materials and civil engineering buildins and structures. There are two key reasons why trustworthy techniques



(A) Comparison of females/F (B) Comparison of females/F and males/M data for 2019-2022 for X-ray and males/M data for 2019-2022 for ultrasound for every year 2019-2022

FIGURE 12. Comparison of females/F and males/M results for 2019-2022 for X/ ray and ultrasound data



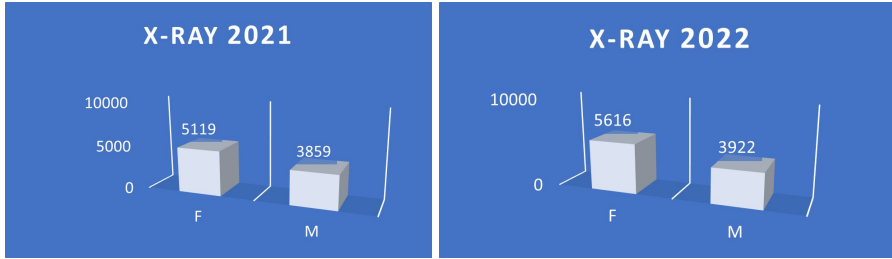
(A) Statistical analyses for 2019

(B) Statistical analyses for 2020

FIGURE 13. Statistical analyses for "X-ray" 2019, 2020

for identifying crack-representing pixels in photos acquired for or digital material and building assessments are needed. First, various alternative techniques based on discontinuous fields employ the segmented crack maps as an input. Second, identified fracture patterns can be used as inputs into empirical predictive models to determine the extent of a body's damage. This sub chapter's goal was to identify visually distinct features in the buildings using crack segmentation on technical photos and images obtained from some campaigns on various walls or stones.

In the Fig. 17, 18, 19 we present and segment civil engineering data. We can observe cracs and data analyses if the engineering data.



(A) Statistical analyses for 2021

(B) Statistical analyses for 2022

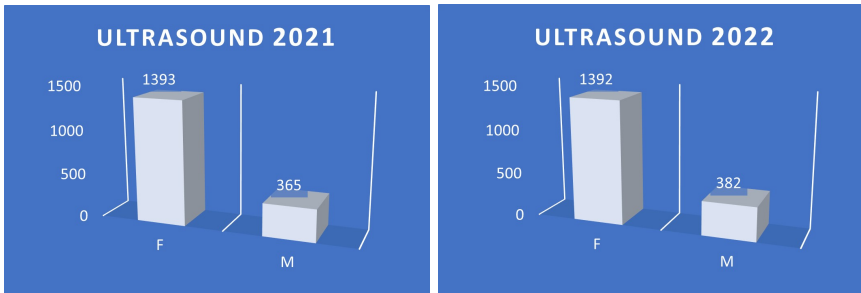
FIGURE 14. Statistical analyses for "X-ray" 2021, 2022



(A) Statistical analyses for 2019

(B) Statistical analyses for 2020

FIGURE 15. Statistical analyses for ultrasound 2019, 2020



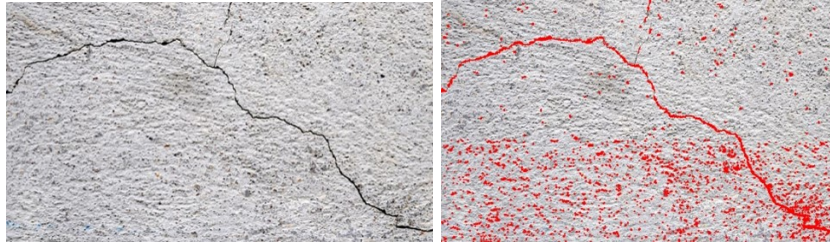
(A) Statistical analyses for 2020

(B) Statistical analyses for 2021

FIGURE 16. Statistical analyses for ultrasound 2021, 2022

5. AGGREGATION APPROACH INTO IMAGE ANALYSES

Another new approach to image segmentation is provided by the approach through aggregation function. The creation of high-quality labeled and annotated data set is a critical part of achieving the algorithmic goal of automated medical



(A) Original data (B) Segmented crack by red color

FIGURE 17. Original and segmented engineering data



(A) Original data (B) Segmented crack by red color

FIGURE 18. Original and segmented engineering data



(A) Original data (B) Segmented crack by red color

FIGURE 19. Original and segmented engineering data

image segmentation. Also engineers consider the study of materials, its strength, its solidity for prediction, where can appear cracks. This type of analysis is equally relevant to civil engineering analyses. We also consider a bottom-up aggregation procedure in which regions are merged based on probabilistic considerations. Here we use the merge strategy proposed for the Segmentation by Weighted Aggregation denoted also (SWA) algorithm which employs a hierarchy construction procedure

inspired by Algebraic Multigrid solutions for differential equations. The SWA algorithm begins with a weighted graph representing image pixels, and in a sequence of steps creates a hierarchy of smaller (“coarse”) graphs with soft relations between nodes at subsequent levels. The approach of the aggregation-based segmentation utilizes adaptive parametric distributions whose parameters are defined locally using image information. Segmentation relies on an integration of intensity and texture cues, with priors determined by the geometry of the regions. The method is modular. It can be extended to handle additional cues.

6. CONCLUSION AND REMARKS

We summarize our results. We developed and created new software for image data analyzing focusing on segmentation methods. Our software are based on Graph cut and Grab Cut method. The novelty approach is in optimizing the time running, in better edge detection of boundaries and also our programs deal very much with the noise in the image. We need to note that other available software do not deal with the "noisy" problem in images. We optimized our programs that they can deal and directly process with the noisy data without the need of pre-processing data. The main originality and novelty in our program and software implementation is the following one:

Our program is made to segment a specific object, (others software not) that it means it greatly reduces noise and takes into account the relationship between neighboring pixels, not just the value of the pixel itself. This is our new discover in implementation.

We present how our programs and our programs tools look like, see Fig. 20.

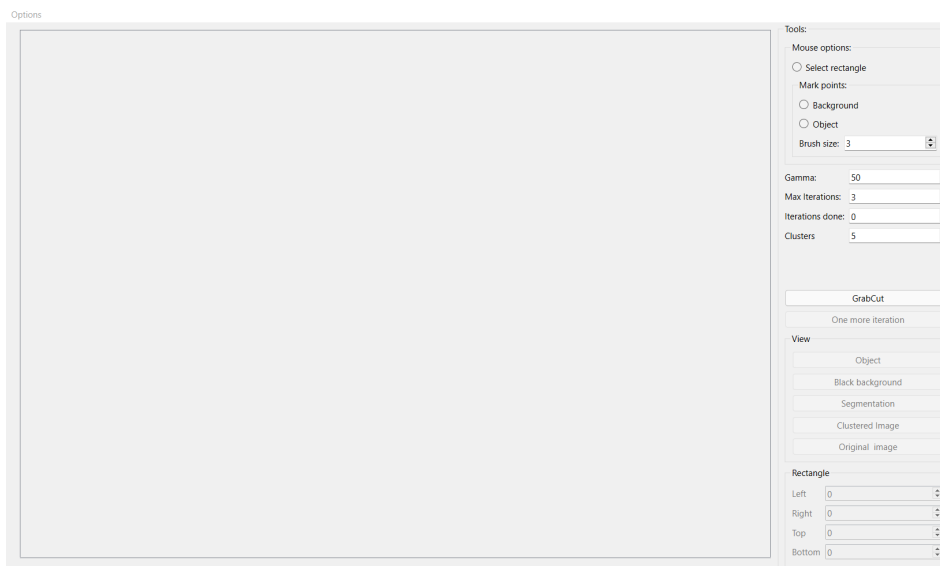
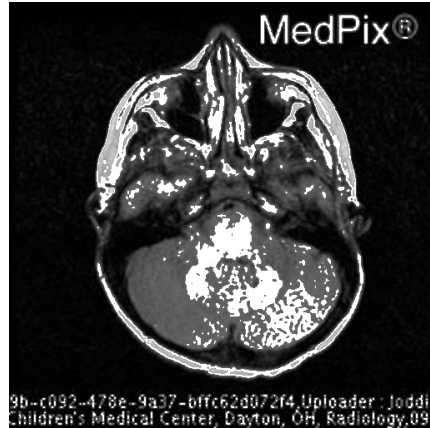
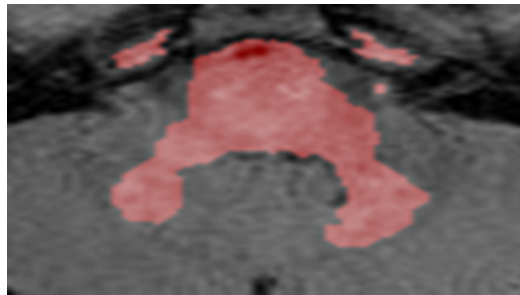


FIGURE 20. Interface for users



(A) Segmented data by threshold techniques

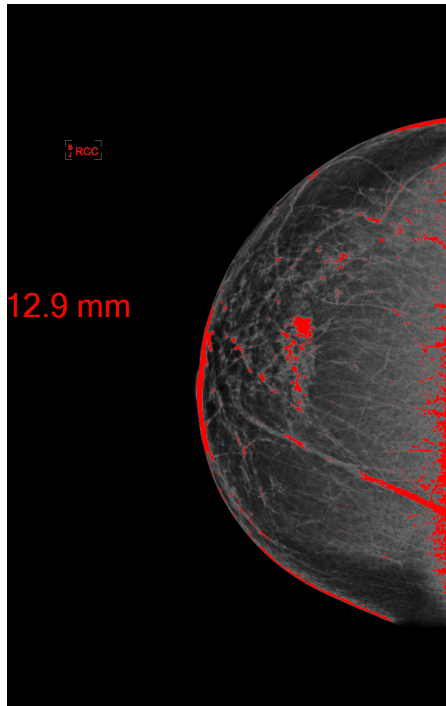


(B) Segmented data by graph techniques

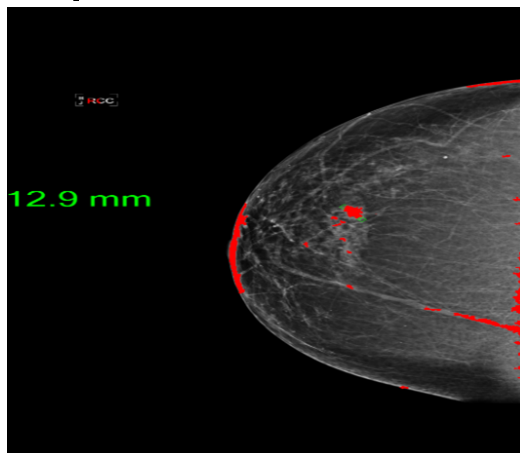
FIGURE 21. Comparison of segmentation techniques

At the end we provide a comparison of our results with other techniques. We proceed data with our implemented programs based on graph algorithms and results based on threshold algorithms. It is possible to observe visually in Fig. 21 and 22. With our graph approach we get much precise and more clear segmentation of objects as with threshold techniques.

In the future we consider also extension into 3D object and 3D modelling in the space and 3D objects.



(A) Segmented data by threshold techniques



(B) Segmented data by graph techniques

FIGURE 22. Comparison of segmentation techniques

Acknowledgement. Mária Ždímalová was funded by the Slovak National Research grant Slovak Grant Vega 1/0036/23 of the Grant Agency of the Ministry of Education and Slovak Academy of Science.

REFERENCES

- [1] B. Basavaprasad, S. Ravindra S., A. Hegadi, *A survey on traditional and graph theoretical techniques for image segmentation*, International Journal of Computer Applications, Recent Advances in Information Technology, (2014), 38–46
- [2] Bondy J.A., Murty U.S., *Graph theory with applications*, Great Britain, The Macmillan Press Ltd., 1976, pp. 264
- [3] Behura A., The Cluster Analysis and Feature Selection in Perspective of Machine Learning and Image Processing Book Editor(s):Rabinarayan Satpathy, Tanupriya Choudhury, Suneeta Satpathy, Sachi Nandan Mohanty, Xiaobo, Zhang First published: 18 January 2021, <https://doi.org/10.1002/9781119785620.ch10>, chapter 10
- [4] Boykov J., Funka-Lea G., *Graph Cuts and Efficient N-D Image Segmentation*, International Journal of Computer vision 70(2), (2006), 109-131
- [5] Boykov, Y. Y. - Jolly, M. P., *Interactive Graph Cuts for Optimal Boundary & Region Segmentation of Objects in N-D Images*, Proceedings of Image Segmentation, International Journal of Computer Vision Issue 2, (2006), 109-131
- [6] Boykov Y., Versler O., *Graph Cuts in vision and graphics: theories and applications*, Handbook of Mathematical models on computer vision, New York, Springer, (2006), 100-118
- [7] J. Edmonds, R. M. Karp, *Theoretical improvements in algorithmic efficiency for network flow problems*, Journal of the ACM (JACM), (2),(19), (1972), 248–264
- [8] P. F., Felzenszwalb, D. P. Huttenlocher, D.P., *Efficient Graph-Based Image Segmentation*, International Journal of Computer Vision, (59), No. 2, (2004), 167–181
- [9] M., D., Fowkles, C. Tal, D., J. Malik, J., *A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics*, (2001), (2), 416–423
- [10] J.R. Ford, L.R. Fulkerson, D.R., *Flows in Networks*, Princeton: Princeton University Press, 1962, pp. 216
- [11] Guha, T., Ward, R., A sparse reconstruction based algorithm for image and video classification, in: Proc. IEEE Conf. on Acoustics, Speech and Signal Processing (ICASSP), (2012), 3601–3604
- [12] A. V. Karzanov, *Determining the Maximal flow in a Network by the Method of Preflows*, SovietMath. Dokl., (15), 2, (1974), 434–437
- [13] J. Loucký, T. Oberhuber, *Graph Cuts in Segmentation of a Left Ventricle from MRI Data*, Proceedings of Czech-Japanese Seminar in Applied Mathematics 2010, COE Lecture Note, (36), (2012), 46–54
- [14] McKinney Wes, *Python for Data Analysis* 3eAutor: 550EAN: 9781098104030ISBN: 109810403XID: 38818436, Printer: O'Reilly Media, 26. August 2022, pp. 550
- [15] M. Peng , L. Zhang , D. Zang , *A survey of graph theoretical approaches to image segmentation*, Pattern recognition, Elsevier, Vol. 46., (2013), 1020-1038.
- [16] L. Pham Dzung, P. Zu Chenuang P., L. Jerry , *Current methods medical image segmentation*, Biomedical Engineering and Annual Review, (2000), 315–337
- [17] M. Sonka, J. Hlavac , R. Boyle, *Graph cuts segmentation*, Image processing, analysis and machine vision, 3.rd ed. TORONTO, Thompson learning, (2008), 980–306
- [18] A.V. Goldberg, R.E. Tarjan, *A new approach to the maximum flow problem*, Journal of Association for Computing Machinery, (35), (1988), 4, 921–940
- [19] M. Ždímalová, Z. Krivá, T. Bohumel, *Graph cuts in image processing*, APLIMAT 2015 - 14th Conference on Applied Mathematics Proceedings, (2015), 774 —78
- [20] Z., WU, R. Leahy, *An Optimal Graph Theoretic Approach to Data Clustering: Theory and Its Application to Image processing*, IEEE Transactions of pattern analyses and machine intelligence, (15), no.11, (1993), 1101–1113
- [21] Peng B., Zhang L., Zang D., A survey of graph theoretical approaches to image segmentation, Pattern recognition, Elsevier, Vol. 46., (2013), 1020–1038.
- [22] <https://wiki.cancerimagingarchive.net/pages/viewpage.action?pageId=52758223>, the page name: Cancer image archive

[23] <https://medpix.nlm.nih.gov/home>, the page name: The National Library of Medicine presents MedPix®, National Library of Medicine

MÁRIA ŽDÍMALOVÁ
SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA,
FACULTY OF CIVIL ENGINEERING,
RADLINSKÉHO 11, BRATISLAVA 810 05 BRATISLAVA, SLOVAKIA
Email address: mzdimalova@gmail.com, maria.zdimalova@stuba.sk

MARIÁN VRÁBEL
SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA,
FACULTY OF ELECTRICAL ENGINEERING AND INFORMATION TECHNOLOGY OF STU
ILKOVIČOVA3, 841 04 BRATISLAVA, SLOVAKIA
Email address: marian.vrabel@is.stuba.sk

KRISTÍNA BORATKOVÁ
SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA,
FACULTY OF CIVIL ENGINEERING,
RADLINSKÉHO 11, BRATISLAVA 810 05 BRATISLAVA, SLOVAKIA
Email address: kristina.boratkova@is.stuba.sk

BPB MED S.R.O.
SREČNIANSKA 13,
SREČNIANSKA 13, BRATISLAVA,85105, SLOVAKIA

Received 30.10.2023

Revised 3.4.2024

Accepted 17.4.2024