

Segmentation Algorithm Based on Square Blocks Propagation

V.V. Danilov¹, I.P. Skirnevskiy¹, R.A. Manakov¹, D.Yu. Kolpashchikov¹, O.M. Gerget¹
viacheslav.v.danilov@gmail.com|skirnevskiy@tpu.ru|ram290495@gmail.com|dyk1@tpu.ru|gerget@tpu.ru

¹Medical Devices Design Laboratory, Tomsk Polytechnic University, Tomsk, Russia

This research is devoted to the segmentation of heart and brain anatomical structures. In the study, we present a segmentation algorithm based on the square blocks (superpixels) propagation. The square blocks propagation algorithm checks two criteria. For the first criteria, the current intensity of the pixel is compared to the average intensity of the segmented region. For the second criterion, the intensity difference of the pixels lying on the superpixel sides is compared to the threshold. Once these criteria are successfully checked, the algorithm merges homogeneous superpixels into one region. Then the following superpixels are attached to the final superpixel set. The last step of the proposed method is the spline generation. The spline delineates the borders of the region of interest. The main parameter of the algorithm is the size of a square block. The cardiac MRI dataset of the University of York and the brain tumor dataset of Southern Medical University were used to estimate the segmentation accuracy and processing time. The highest Dice similarity coefficients obtained by the presented algorithm for the left ventricle and the brain tumor are 0.93 ± 0.03 and 0.89 ± 0.07 respectively. One of the most important features of the border detection step is its scalability. It allows implementing different one-dimensional methods for border detection.

Keywords: square blocks propagation, superpixels, region growing, left ventricle segmentation, brain tumor segmentation.

1. Introduction

Medical image segmentation is one of the most challenging tasks in the field of medical image processing. The segmentation and the subsequent analysis of medical images allow clinicians to predict disease, plan surgery procedures or assess the condition of internal organs. At the moment, many robust two- and three-dimensional segmentation techniques have been proposed [19, 22, 27]. The recent and the most popular articles on medical image segmentation are inextricably connected with machine learning and neural networks. Ozan Oktay used neural networks for cardiac image enhancement and segmentation in paper [17]. In paper [20] machine learning algorithms are used for brain tumor segmentation. Similar approaches have been used in many tasks of medical image analysis [14, 16]. However, algorithms based on machine learning often solve a narrow problem and require large training datasets. Despite the popularity of machine learning algorithms, common segmentation techniques remain relevant and keep improving [15, 25]. Semi-automatic image segmentation techniques are still popular because of their simplicity, a small number of parameters, and scalability.

Today classical image segmentation techniques (active contours, region growing, watershed segmentation) are used in many semi-automatic image processing algorithms. It is worth noticing that analysis and processing of three-dimensional images are still difficult especially in the field of cardiology or brain imaging. Therefore, there are cases when clinicians use two-dimensional planes for the analysis and segmentation. Moreover, the more popular machine learning algorithms become, the more data they require for training machine learning models. Thus, there is a need in an easy-to-use environment for data labeling. Two-dimensional segmentation techniques are often used as such environments.

In this study, we present a two-dimensional segmentation algorithm based on square blocks propagation (SBP). Dana Ballard and Rofl Adams' algorithms [2, 4] inspired us to develop this method. We also used the approaches described in work [5]. The proposed algorithm is slightly similar to a classical region growing and based on the merging of the samples with similar properties of the region.

2. Related works

Most of the articles devoted to the region growing (RG) algorithms were presented more than a decade ago. Jun Tang proposed a method for color image segmentation based on a combination of the seeded region growing and watershed algorithm [26]. However, there were no modifications or improvements in the

algorithms. The watershed segmentation is used as an initial step to find a seed region.

Joung Park and Chulhee Lee used the seeded region growing algorithm for the skull stripping [18]. In that algorithm, a morphological mask was used for the automatic identification of the initial seed points of background and foreground. Other region-based methods such as watershed segmentation and morphological segmentation are used in tasks of the skull stripping [8, 23]. However, many of these approaches have drawbacks, such as oversegmentation and noise sensitivity.

Nor Isa in paper [13] proposed a modified seed-based region growing algorithm. Several important blocks of the algorithm such as setting the threshold value, determining the initial seed point, and the growing process were modified and automated. For instance, the automatic determination of the seed point is based on the k-means clustering algorithm. However, the approaches described in the paper have high computational complexity. This is explained by a number of preliminary calculations. For instance, the k-means clustering algorithm is working with an entire image. Jamshid Dehmeshki combined the region growing and the fuzzy connectivity region growing approaches in paper [11].

Paper [12] shows a modified region growing based on the merging superpixels. A superpixel is a group of pixels combined by a certain feature. The superpixel term was introduced by Xiaofeng Ren and Jitendra Malik in [21]. The superpixel concept is used in the presented study. However, the major difference of the proposed algorithm is that it does not check every pixel of the superpixel. In general, a number of segmentation algorithms based on superpixels were proposed before [9, 10, 24]. It should be noted that the clustering methods are used for a superpixel generation in most studies. In this approach, each pixel included in a superpixel should be processed separately, thus causing a relatively high level of complexity. The complexity of such algorithms is $O(n^2)$. A simple linear iterative clustering (SLIC) algorithm is used in studies [12], [20], and [9]. Initially, this method was presented in paper [1], where Radhakrishna Achanta developed a superpixel-based segmentation method using k-means clustering for a five-dimensional feature space. The first three dimensions are the color space and the last two are the pixel coordinates. The SLIC algorithm is a modified k-means clustering algorithm which does not compare each pixel with other pixels in the image. Ovidiu Csillik in paper [10] demonstrated a method based on SLIC superpixels for a high-resolution segmentation. The paper presents a processing time of superpixels at different resolutions. This method processes 1347×1042 and 3701×3301 images for 2 and 26 seconds respectively.

As shown above, segmentation algorithms based on superpixels are quite popular. However, most of the considered

approaches have a high level of algorithmic complexity. In this regard, we propose a 2D semi-automatic segmentation algorithm based on a superpixel growing where superpixels have a floating size. The latter allowed us to achieve better algorithmic performance.

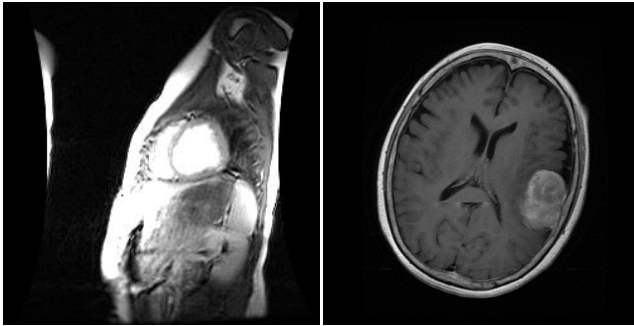
3. Data and methods

3.1 Data description

In order to develop and validate the proposed algorithm, we used open-access datasets. The first dataset was provided by the University of York (York, United Kingdom) and contains 33 subjects [3]. Each subject's sequence consists of 20 frames and 8-15 slices (256x256 pixels) along the long axis, for a total of 7980 images. Two clinicians manually segmented all the images of the dataset. The ground truth of the left ventricles' endocardial and epicardial was acquired.

The second source of data is the brain tumor dataset. This dataset includes 3064 T1-weighted contrast-enhanced images. The dataset was acquired at Southern Medical University and contains data from 233 patients with three kinds of brain tumors: meningioma, glioma, and pituitary tumor (Guangzhou, China) [6, 7]. The size of MRI images is 512*512 pixels. Examples of the heart and the brain tumor images are shown in Fig. 1.

Both datasets were processed offline on the computer equipped with Intel Core i7-4820K 3.7GHz CPU and NVIDIA GeForce 960 GT using MATLAB (MathWorks, Natick MA).



(a) Heart MRI sample (b) Brain tumor MRI sample

Fig. 1. Examples of source data

3.2 Region growing algorithm

Starting from a seed of the region of interest (ROI) the region growing algorithm performs a segmentation. The region is growing due to the connection of the neighboring pixels, which satisfy the criterion of homogeneity. There are two versions of the algorithm: a seeded version with a manual selection of the seed point and an unseeded version with a random seed point. A classical region growing algorithm is conceptually shown in Fig. 2.

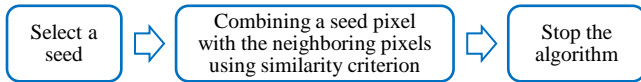


Fig. 2. The workflow of the region growing algorithm

A criterion of merging neighboring pixels is presented below and also shown in [2].

$$P(x, r) = |f(x) - \mu_r| < T, \quad (1)$$

where $f(x)$ is the intensity of the current pixel, μ_r is the arithmetic mean intensity of the region, T is the threshold level. The approach described above is a standard implementation of a region growing algorithm.

3.3 Square blocks propagation

There is no conceptual difference between the classical seeded region growing and the proposed algorithm. However, the SPB algorithm assumes a translation to the domain of superpixels. This allows reducing the complexity of the algorithm and increasing the processing speed. In the proposed algorithm a superpixel represents a square block comprising of the pixels. All pixels inside the block have a 4-connected neighborhood by default. The SPB algorithm checks two criteria, described in Section 3.6 in more detail, and merges superpixels into one region. Then points lying on superpixels borders are used for spline generation. The workflow of the proposed algorithm is shown below in Fig. 3.

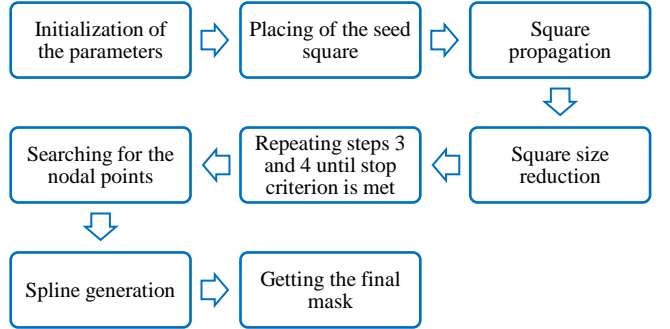
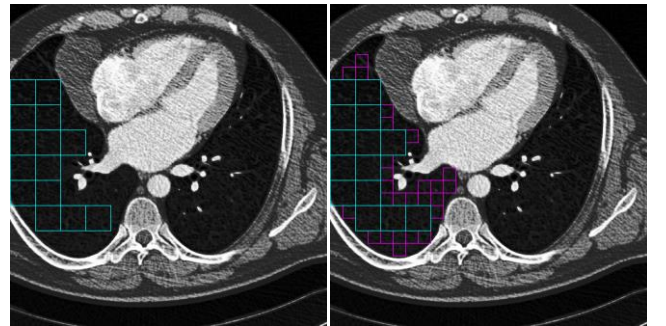


Fig. 3. The workflow of the square blocks propagation algorithm

The main difference between the proposed algorithm and the methods reflected in papers [12, 28] is that the proposed algorithm does not analyze single pixels belonging to superpixels. For instance, the standard region growing algorithm processes all 100 pixels of a 10x10 superpixel. In turn, the presented algorithm processes only 50 out of 100 pixels. Thus, the larger the initial square size is, the higher the algorithm speed is. The concept of the square propagation and the size reducing procedure is shown in Fig. 4.



(a) Propagation using initial size squares (b) Propagation using squares with the reduced size

Fig. 4. Square propagation and square size reduction. Blue blocks show the first step of the square propagation with an initial superpixel size. Purple blocks demonstrate the second step of the square propagation and the square size reduction

The proposed algorithm applies superpixel growing instead of pixels merging. This approach accelerates the segmentation process. Similar to the region growing algorithm, a merging process occurs as long as there is at least one group of pixels that could be included in the final set. The superpixel is included in the final superpixel set when the superpixel sides do not cross the border of the ROI (see Section 3.6). If there is at least one border crossing on the checked segments, then the square block is not included in the final superpixel set.

The first square center is the starting point chosen manually. A square of a given size is placed around the first point. The diagonals and the sides of the square are checked for the border crossing. If the square does not cross the border of the ROI, it is placed in an

image. Otherwise, the algorithm can continue only if the size of the initial square block is reduced.

3.4 Search for outer squares

Superpixels are validated and attached to the final superpixel set as long as possible. If there are no squares with the given size which can be added to the studied region, the size of the square is halved and the propagation process continues. An example of the outer squares found by the algorithm is shown in Fig. 5. The green points indicate an intersection of the squares with the ROI.



Fig. 5. Example of outer squares that cannot be added to the ROI (yellow squares) because of the intersection with the segmented area border. Green points demonstrate this intersection

The square size reducing procedure can be repeated several times until the minimum size of the square is reached. The minimum square size is one of the parameters of the algorithm. At the final step of the algorithm, the superpixels with the smallest size are located close enough to the border of the segmented area but never cross it. An obligatory condition for completing this stage is the impossibility of further attaching the square blocks.

3.5 Delineation and masking

At this step, the algorithm bypasses the outer superpixels which have not been attached to the region. These squares form a contour in accordance with the mandatory condition described above in Section 3.4. The contour consisted of outer superpixels is bypassed clockwise. The intersection points are saved for each superpixel crossing the contour of the ROI, creating a list of nodal border points. The same procedure is performed to bypass the inner borders of the region if such exist.

Having a list of nodal points, it is possible to construct a Hermite cubic spline describing the inner and outer contours of the segmented region (see Fig. 6). Thus, the result of the algorithm is a spline or a set of splines.

Fig. 6 shows a conceptual scheme of spline generation. As shown, the blue line has an unusual shape for a cubic spline. The output segmentation mask obtained using spline generation is shown in Fig. 7.

Among the entire set of image pixels L , the algorithm finds the set of intersection points P . Each intersection point represents a point where a square block sides or diagonals cross the border of a region. One of the ways to obtain a contour is to construct a regression of these points. However, we did not use a linear spline, as it gives a significant error in constructing the contour borders. We also refused to use B-splines because they do not pass through the extreme points. The latter is not acceptable since it significantly reduces the accuracy of segmentation.

Thus, the result of the algorithm is the set of cubic splines describing the contours. Such an analytical presentation may be more preferable than representing a segmented area in the form of

a mask. However, it is possible to switch from spline to mask using existing methods.

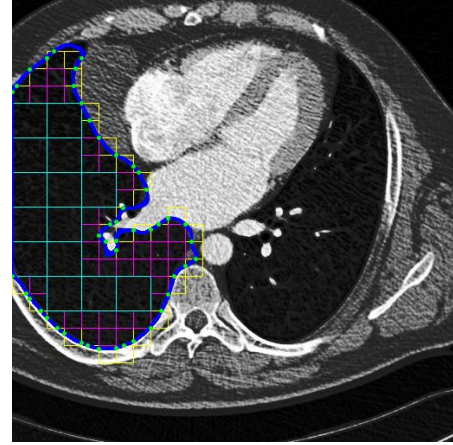


Fig. 6. Hermite cubic spline (blue line) passing through the nodal points (green points)

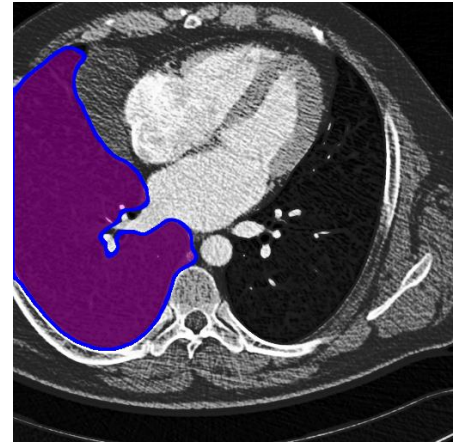


Fig. 7. Obtaining a segmentation mask. Cubic spline (blue) and segmentation mask (purple) represent the region of interest

3.6 Border detection

To detect the border of the ROI, the proposed algorithm applies two conditions. For the first condition, the intensity of the side/diagonal pixel is compared with the mean intensity of center pixels of already placed squares. The threshold parameter is a configurable parameter of the algorithm.

$$\Delta I = \left| p_i - \frac{\sum_{j=1}^k p_{c_j}}{k} \right| \geq T, \quad (2)$$

where p_i is the intensity of the current pixel, p_{c_j} is the intensity of the center pixel of a certain square, k is the number of already placed squares, T is the threshold level.

For the second condition, the intensity difference of the pixels lying on the superpixel sides is compared with the threshold. The approximation is performed using the method of least squares at 5 points. The coefficient, representing a slope of the straight line is calculated as follows:

$$\tan(\varphi) = \left| \frac{n \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \sum_{i=1}^n y_i}{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2} \right| \geq \text{slope}, \quad (3)$$

where x is the edge points varied in a certain range (in our case this range is from 1 to 5), y are intensity values, n are positions of the edge points. When constructing the vector x , it should be considered that the distance between pixels is the Euclidean.

For the current pixel, the approximation is done using two pixels on the right and two on the left. If the pixels are in the corner

of a superpixel, the additional pixels that slightly exceed the borders of the square block are taken. The slope module allows the algorithm to accurately detect the region borders and makes the algorithm resistant to noise. Border search is applied on the sides and diagonals of the square blocks using conditions described above. If there is at least one side/diagonal crossing of the region border, the square block is not included in the final set. Thus, the reliability of the algorithm rises.

As shown in Fig. 8, each square block has six-line segments (AB, BC, CD, AD, AC, and BD). These lines consist of pixels. All that we need to do is to perform a one-dimensional segmentation for each line. This approach significantly reduces the execution time of the algorithm. Another advantage of the proposed solution is the possibility to apply any set of one-dimensional segmentation methods to a line segment. Therefore, any segmentation method can be implemented in the proposed algorithm as a plug-in for the additional verification of the border crossing. It is worth noticing that if at least one segment has crossed the border of the region, the algorithm does not process the rest of the line segments. The latter allows reducing the algorithm runtime and increasing the performance of the algorithm by 5 times in the extreme case.

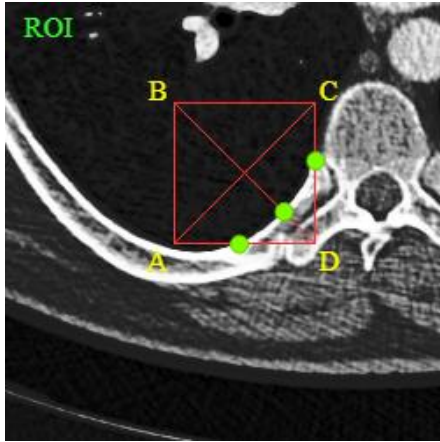


Fig. 8. Square block (red) crossing the ROI in three points (green)

3.7 Size of square blocks

The region growing algorithm often leaks through the holes in borders. Some modifications of the region growing [21] can help to avoid this effect, but these methods still have high computational complexity. The proposed algorithm eliminates this disadvantage due to the variability of the square blocks size. The minimum square size also has an influence on the contour details and its smoothness. The larger the side of the square is, the smoother and less detailed the contour is. As the square size decreases, the border becomes more detailed while the probability of negative leakage effect increases.

The initial square size defines the propagation speed. It should also be noted that the initial square size depends on the segmented area and the image resolution. For initial square size K we used the method presented in paper [12], where parameter K is calculated as follows:

$$K = \frac{N}{SF} \quad (4)$$

where N is a number of image pixels, and SF is the size of the smallest segmented object in the image.

3.8 Algorithm complexity

In the case of the region growing, at least N^2 steps are required to process a block of $N \times N$ pixels. Consequently, the complexity of the region growing is $O(n^2)$. Each image pixel is processed separately by the region growing algorithm. For the proposed algorithm, a number of iterations for the block size of $N \times N$ pixels varies from $3 \times N$ to $5 \times N$. However, there is an exceptional case for

the first square where a number of iterations is equal to $6 \times N$. In this regard, the asymptotic complexity of the SBP algorithm is $O(n)$. As shown, the proposed approach moves from the pixel level to the level of the pixel groups and fragments. The latter allows remarkably reducing the execution time of the algorithm.

4. Results

In this section, we studied how the accuracy and processing time changed with respect to different sizes of the squares. The Dice Similarity Coefficient (DSC) was used as the main metric for the accuracy assessment.

4.1 Left ventricle segmentation

The left ventricle segmentation of the presented algorithm, region growing algorithm, and the ground truth (GT) manual segmentation is presented in Fig. 9. In the case of patient 2 and patient 3, region growing leaks out through the gaps in the borders of the ROI. This is because the region growing approach processes an image at the pixel domain. In turn, the SBP algorithm avoids the problem of the border gaps.

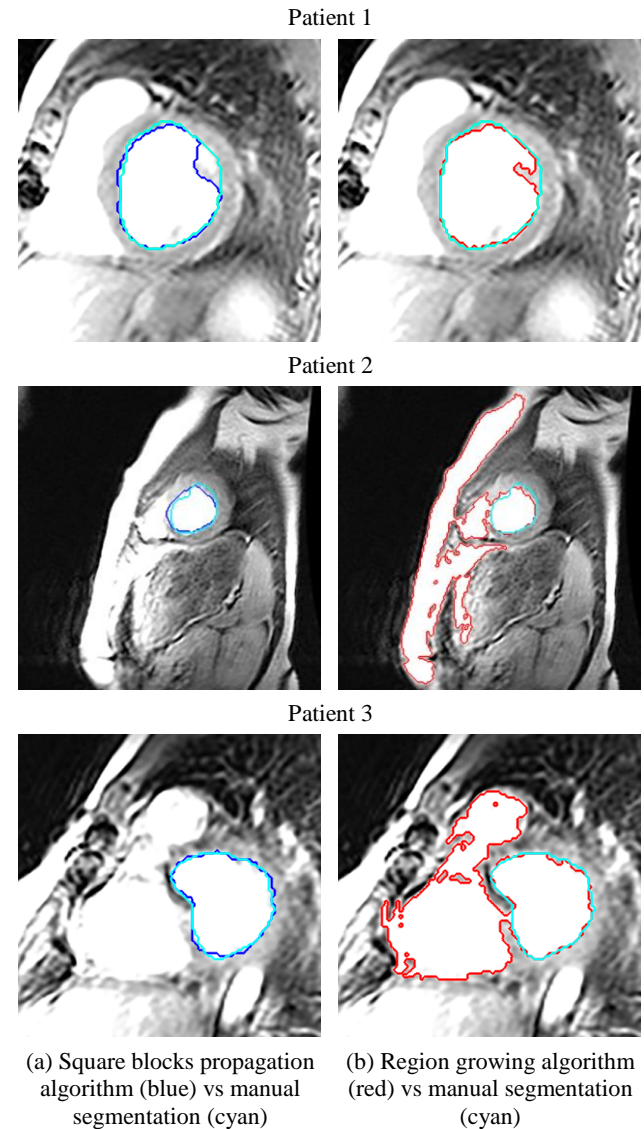


Fig. 9. Segmentation of the left ventricle using the proposed SBP and RG algorithms in comparison with the ground truth manual segmentation

To test the segmentation accuracy and processing time of the left ventricle, we used a dataset comprised of 156 slices. The left ventricle segmentation accuracy for different square sizes is shown

in Fig. 10 and Table 1. Additionally, the total number of low-accuracy cases when DSC is less than 0.5 is shown in Fig. 11.

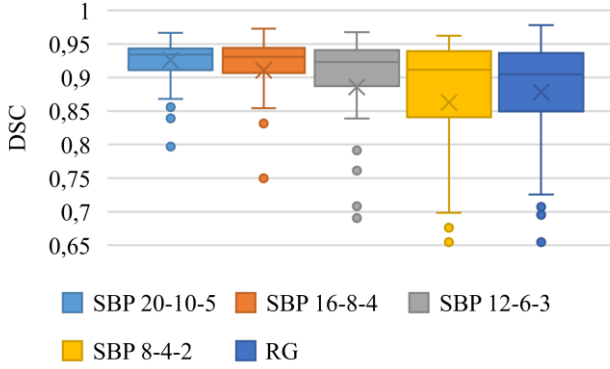


Fig. 10. Left ventricle segmentation accuracy of the proposed algorithm

Table 1. DSC obtained on the left ventricle dataset for different algorithms.

SBP 20-10-5	SBP 16-8-4	SBP 12-6-3	SBP 8-4-2	RG
0.93±0.03	0.91±0.07	0.89±0.10	0.86±0.11	0.88±0.09

As shown in Fig. 10 and Table 1, DSC values of SBP 8-4-2, SBP 12-6-3 and RG do not differ significantly. However, the DSC interquartile range of SPB with parameters 20-10-5 and 16-8-4 is significantly better than RG's one. The average accuracy of the SPB algorithm has grown significantly due to the fact that superpixels do not leak through the border gaps.

Better performance of the proposed algorithm is indirectly confirmed by a number of low-accuracy segmentation cases (see Fig. 11). A low-accuracy case is a leakage case or a case with the value of DSC less than 0.5. In 32% of the studied cases, RG leaks through the borders defects what confirms its unreliability.

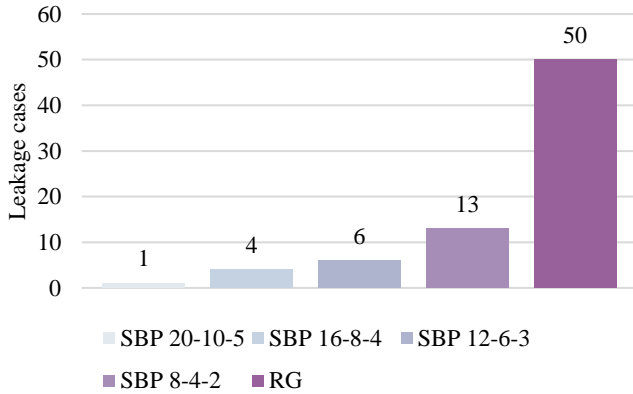
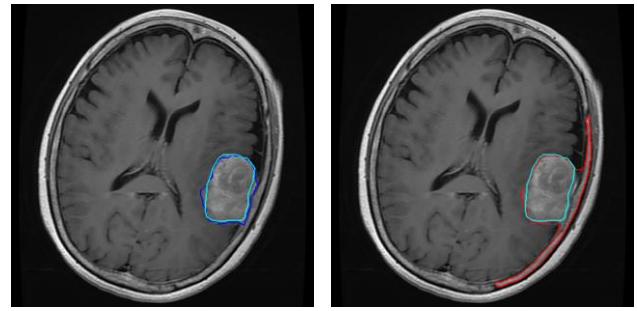


Fig. 11. Low-accuracy cases during heart segmentation

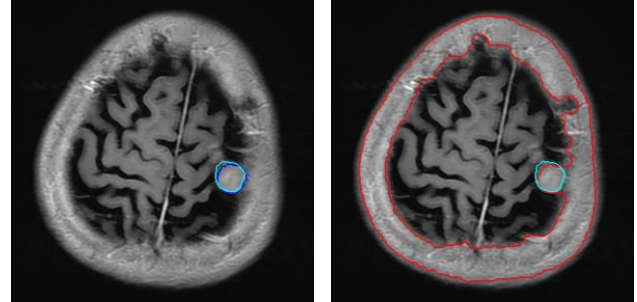
4.2 Brain tumor segmentation

The brain tumor segmentation of the presented algorithm, region growing algorithm, and the ground truth manual segmentation is presented in Fig. 12. As shown, the region growing algorithm has a problem related to the leakage through the border gaps. In this case, the bone tissue is mistakenly segmented for the three presented patients. Such properties of the image lead to low accuracy of the region growing. In turn, SPB allows configuring the size of superpixels to avoid oversegmentation and then segmenting the tumor successfully.

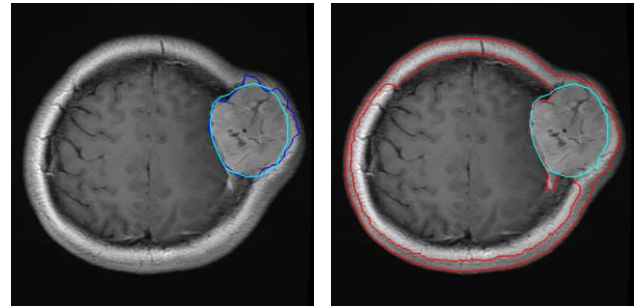
Patient 1



Patient 2



Patient 3



(a) Square blocks propagation algorithm (blue) vs manual segmentation (cyan) (b) Region growing algorithm (red) vs manual segmentation (cyan)

Fig. 12. Segmentation of the brain tumor using the proposed SBP and RG algorithms in comparison with the ground truth manual segmentation

To test the segmentation accuracy and processing time of the brain tumor, we used a dataset comprised of 300 slices. The brain tumor segmentation accuracy for different square sizes is shown in Fig. 13 and Table 2. Additionally, the total number of low-accuracy cases when DSC is less than 0.5 is shown in Fig. 14.

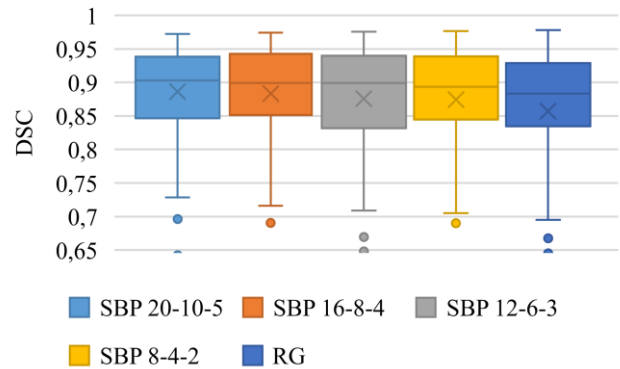


Fig. 13. Brain tumor segmentation accuracy of the proposed algorithm

Table 2. DSC obtained on the brain tumor dataset for different algorithms.

SBP 20-10-5	SBP 16-8-4	SBP 12-6-3	SBP 8-4-2	RG
0.89±0.07	0.88±0.08	0.88±0.08	0.87±0.09	0.86±0.10

In the case of the brain tumor segmentation, pseudo proportionality between the DSC and the size of the squares is observed. The latter means that the smaller the square size is, the less the DSC value is. It should be noted that the reason for these leaks is not the borders defects. In this case, the tumor has approximately the same level of intensity as external bone tissue.

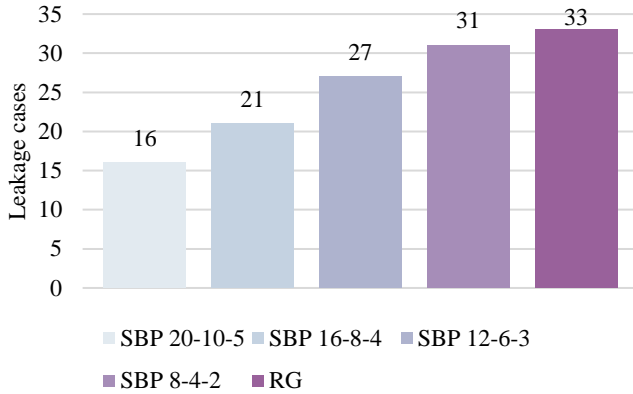


Fig. 14. Low-accuracy cases during brain tumor segmentation

4.3 Execution time testing

To compare the propagation speed of the region growing algorithm and the proposed algorithm, a synthetic test image with a white circle in the center and the black background was generated. This test image was created in different sizes. The dependence between processing time and image sizes for both algorithms is represented in Fig. 15 and Fig. 16. As seen, both algorithms have asymptotic complexity $O(n^2)$ but the region growing algorithm is much slower, and cannot be adapted for optimal speed.

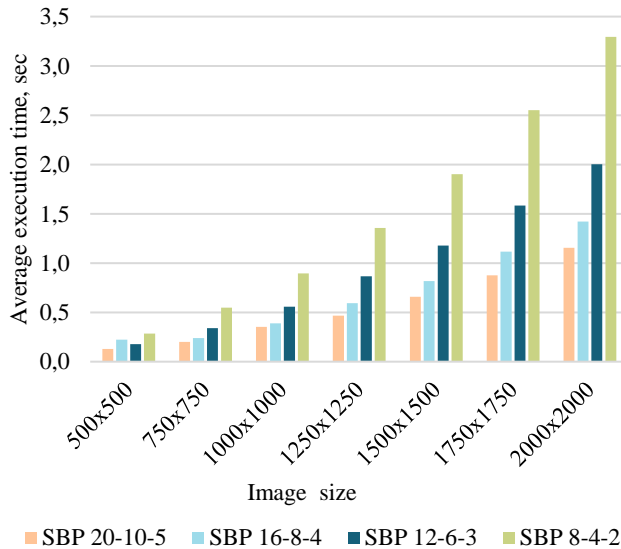


Fig. 15. The average execution time of the SBP algorithm for different sizes of images and square blocks

5. Conclusion

The proposed algorithm is devoted to the segmentation of images with high resolution or medical images with ROI border defects and low contrast. Additionally, this algorithm can be

applied to data labeling. The most significant factor of the algorithm speed is the maximum square size and the sequence of sizes of the square blocks in common. Increasing the size of the largest square from the chosen sequence makes the image processing faster. On the other hand, the sizes should be chosen in the way that at least one square block is placed in the ROI.

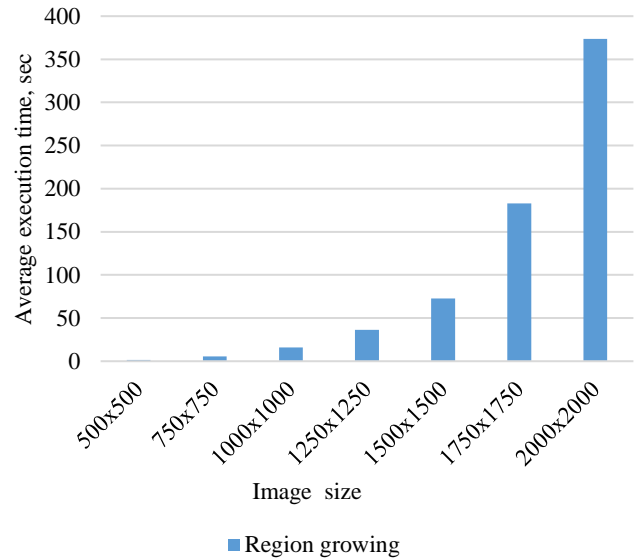


Fig. 16. The average execution time of the RG algorithm for different image and square block sizes

The proposed algorithm has opportunities to improve execution time, robustness, and final accuracy. All squares are processed independently to each other, so the algorithm can be paralleled on GPU for minimizing execution time.

An important feature of the algorithm is its scalability. It means that several different algorithms can be used for border detection at the same time. We used two criteria: one-dimensional region growing and intensity gradient check. As an additional method, machine learning or one-dimensional neural networks can be applied to the border detection. It should also be noted that the algorithm can be extended for three-dimensional imaging.

6. Acknowledgments

This work was supported in part by the Russian Federation Governmental Program “Nauka” № 12.8205.2017/БЧ (additional number: 4.1769.Г3Б.2017).

7. References

- [1] Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P. and Süsstrunk, S. 2012. SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 34, 11 (2012), 2274–2281. DOI:https://doi.org/10.1109/TPAMI.2012.120.
- [2] Adams, R. and Bischof, L. 1994. Seeded Region Growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 16, 6 (1994), 641–647. DOI:https://doi.org/10.1109/34.295913.
- [3] Andreopoulos, A. and Tsotsos, J.K. 2008. Efficient and generalizable statistical models of shape and appearance for analysis of cardiac MRI. *Medical Image Analysis*. 12, 3 (Jun. 2008), 335–357. DOI:https://doi.org/10.1016/j.media.2007.12.003.
- [4] Ballard, D.H. and Brown, C.M. 1982. *Computer Vision*.
- [5] Bankman, I.N. 2000. *Handbook of Medical Imaging*.
- [6] Cheng, J., Huang, W., Cao, S., Yang, R., Yang, W., Yun,

- Z., Wang, Z. and Feng, Q. 2015. Enhanced performance of brain tumor classification via tumor region augmentation and partition. *PLoS ONE*. 10, 10 (2015). DOI:<https://doi.org/10.1371/journal.pone.0140381>.
- [7] Cheng, J., Yang, W., Huang, M., Huang, W., Jiang, J., Zhou, Y., Yang, R., Zhao, J., Feng, Y., Feng, Q. and Chen, W. 2016. Retrieval of Brain Tumors by Adaptive Spatial Pooling and Fisher Vector Representation. *PLoS ONE*. 11, 6 (2016). DOI:<https://doi.org/10.1371/journal.pone.0157112>.
- [8] Chiverton, J., Wells, K., Lewis, E., Chen, C., Podda, B. and Johnson, D. 2007. Statistical morphological skull stripping of adult and infant MRI data. *Computers in Biology and Medicine*. 37, 3 (2007), 342–357. DOI:<https://doi.org/10.1016/j.compbiomed.2006.04.001>.
- [9] Crommelinck, S., Bennett, R., Gerke, M., Koeva, M.N., Yang, M.Y. and Vosselman, G. 2017. SLIC Superpixels for Object Delineation from UAV Data. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences* (2017), 9–16.
- [10] Csillik, O. 2017. Fast segmentation and classification of very high resolution remote sensing data using SLIC superpixels. *Remote Sensing*. 9, 3 (2017). DOI:<https://doi.org/10.3390/rs9030243>.
- [11] Dehmeshki, J., Amin, H., Valdivieso, M. and Ye, X. 2008. Segmentation of pulmonary nodules in thoracic CT scans: A region growing approach. *IEEE Transactions on Medical Imaging*. 27, 4 (2008), 467–480. DOI:<https://doi.org/10.1109/TMI.2007.907555>.
- [12] Duong, T.H. and Hoberock, L.L. 2018. DUHO image segmentation based on unseeded region growing on superpixels. *2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC)* (Jan. 2018), 558–563.
- [13] Isa, N.A.M., Sabarudin, S., Ngah, U.K. and Zamli, K.Z. 2005. Automatic detection of breast tumours from ultrasound images using the modified seed based region growing technique.
- [14] Kamnitsas, K., Ledig, C., Newcombe, V.F.J., Simpson, J.P., Kane, A.D., Menon, D.K., Rueckert, D. and Glocker, B. 2017. Efficient multi-scale 3D CNN with fully connected CRF for accurate brain lesion segmentation. *Medical Image Analysis*. 36, (2017), 61–78. DOI:<https://doi.org/10.1016/j.media.2016.10.004>.
- [15] Kang, H.C., Lee, J. and Shin, J. 2016. Automatic four-chamber segmentation using level-set method and split energy function. *Healthcare Informatics Research*. 22, 4 (2016), 285–292. DOI:<https://doi.org/10.4258/hir.2016.22.4.285>.
- [16] Litjens, G., Kooi, T., Bejnordi, B.E., Setio, A.A.A., Ciompi, F., Ghafoorian, M., van der Laak, J.A.W.M., van Ginneken, B. and Sánchez, C.I. 2017. A survey on deep learning in medical image analysis. *Medical Image Analysis*.
- [17] Oktay, O., Ferrante, E., Kamnitsas, K., Heinrich, M., Bai, W., Caballero, J., Cook, S.A., De Marvao, A., Dawes, T., O'Regan, D.P., Kainz, B., Glocker, B. and Rueckert, D. 2018. Anatomically Constrained Neural Networks (ACNNs): Application to Cardiac Image Enhancement and Segmentation. *IEEE Transactions on Medical Imaging*. 37, 2 (2018), 384–395. DOI:<https://doi.org/10.1109/TMI.2017.2743464>.
- [18] Park, J.G. and Lee, C. 2009. Skull stripping based on region growing for magnetic resonance brain images. *NeuroImage*. 47, 4 (2009), 1394–1407. DOI:<https://doi.org/10.1016/j.neuroimage.2009.04.047>.
- [19] Pham, D.L., Xu, C. and Prince, J.L. 2000. Current Methods in Medical Image Segmentation. *Annual Review of Biomedical Engineering*. 2, 1 (2000), 315–337. DOI:<https://doi.org/10.1146/annurev.bioeng.2.1.315>.
- [20] Pinto, A., Alves, V. and Silva, C.A. 2016. Brain Tumor Segmentation using Convolutional Neural Networks in MRI Images. *IEEE Transactions on Medical Imaging*. 35, 5 (2016), 1240–1251. DOI:<https://doi.org/10.1109/TMI.2016.2538465>.
- [21] Ren, X. and Malik, J. 2003. Learning a classification model for segmentation. *Proceedings Ninth IEEE International Conference on Computer Vision*. 1, c (2003), 10–17 vol.1. DOI:<https://doi.org/10.1109/ICCV.2003.1238308>.
- [22] Rogowska, J. 2009. Overview and fundamentals of medical image segmentation. *Handbook of Medical Image Processing and Analysis*. 73–90.
- [23] Roy, S. and Maji, P. 2015. A simple skull stripping algorithm for brain MRI. *ICAPR 2015 - 2015 8th International Conference on Advances in Pattern Recognition* (2015).
- [24] Saxen, F. and Al-Hamadi, A. 2014. Superpixels for Skin Segmentation. *Www-E.Uni-Magdeburg.De*. (2014).
- [25] Soliman, A., Khalifa, F., Elnakib, A., El-Ghar, M.A., Dunlap, N., Wang, B., Gimel'farb, G., Keynton, R. and El-Baz, A. 2017. Accurate lungs segmentation on CT chest images by adaptive appearance-guided shape modeling. *IEEE Transactions on Medical Imaging*. 36, 1 (2017), 263–276. DOI:<https://doi.org/10.1109/TMI.2016.2606370>.
- [26] Tang, J.T.J. 2010. A color image segmentation algorithm based on region growing. *Computer Engineering and Technology (ICCET), 2010 2nd International Conference on*. 6, (2010), 634–637. DOI:<https://doi.org/10.1109/ICCET.2010.5486012>.
- [27] Tsechpenakis, X.H.G. 2013. Medical Image Segmentation. *Advanced Materials Research*. i (2013), 1–35. DOI:<https://doi.org/10.1201/9781420090413-c10>.
- [28] Wang, L., Pei, M., Codella, N.C.F., Kochar, M., Weinsaft, J.W., Li, J., Prince, M.R. and Wang, Y. 2015. Left ventricle: Fully automated segmentation based on spatiotemporal continuity and myocardium information in cine cardiac magnetic resonance imaging (LV-FAST). *BioMed Research International*. (2015). DOI:<https://doi.org/10.1155/2015/367583>.