Medical Image Segmentation with Split-and-Merge Method

Sándor Szénási
Óbuda University, John von Neumann Faculty of Informatics, Budapest, Hungary
szenasi.sandor@nik.uni-obuda.hu

***Abstract*— The processing of microscopic tissue images and especially the detection of cell nuclei is nowadays done more and more using digital imagery and special immunodiagnostic software products. One of the most promising methods is region growing but it is quite memory intensive. The size of high-resolution tissue images can easily reach the order of a hundred megabytes therefore the memory requirement for the region growing is more than one gigabyte. To provide the execution in low-end clients we have to split the whole image into smaller tiles and after the processing of each individual tiles we have to merge the results.**

Keywords—medical image segmentation, parallel algorithm, gpgpu, split-and-merge

I. INTRODUCTION

There are several advantages of digital images (like administration [1], further processing [2], etc.). In the field of medical image processing, several procedures are based on the segmentation of the image and a lot of them need the locations of the nuclei. This is usually a step of crucial importance, since normally this partial result is the basis of the further processing. There are several image processing algorithms for this purpose, but in the context of biomedical analysis there are some factors which could increase the challenge. The size of high-resolution tissue images can easily reach the order of a hundred megabytes (Fig. 1.) [3].

One of the promising alternatives is the region growing approach, which is a classical image segmentation method. The first step is to select a set of seed points which needs some suspicion about the pixels of the required region. After that it examines the neighbouring pixels of the initial seed points and determines whether the pixel neighbours should be added to the region or not (using a special fitness function). This process is iterated until some exit condition is met.

The region growing method has some limitations: speed and memory. In our previous work we have partially solved the speed problem; parallelizing the region growing algorithm and running it on a GPU [4] aims at providing better execution times, while delivering the similar outcome produced by the sequential version. But the memory limitation already exists. The algorithm uses several copies of the original tissue image (modified by some filters) and in case of large images (each instance of a truecolor image with dimensions 8192x8192 pixel requires ~192 megabytes of memory) the region growing approach becomes unstable.

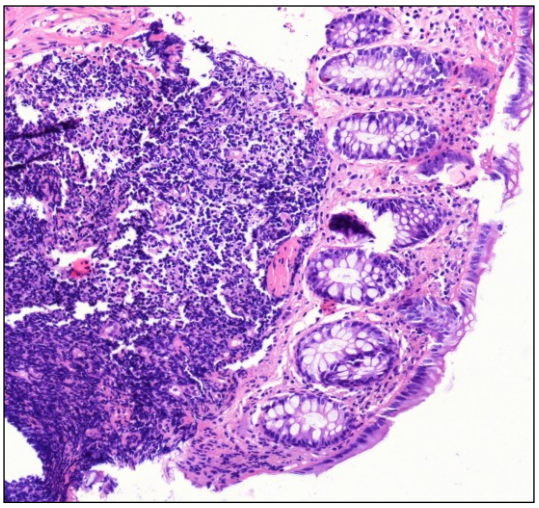

Fig. 1. HE stained colon tissue image

Therefore we have developed a new version of the algorithm which is available to process large images using the classic split-and-merge technique [5] [6]. As the name implies in the first step it splits the whole image into smaller images (tiles), in the next step it runs the region growing on these smaller images, and finally it merges the results of the separate region growings.

II. THE ALGORITHM

A. The split phase

In the cutting up phase we have to intend on not decreasing the accuracy (within the bounds of possibility). Theoretically every single slice can be considered as independents from each other, therefore the parallel processing will not cause inaccuracy. The problem here caused (as in similar tasks many times) by the nuclei on the boundaries of the sections (especially the cross-border ones). Because each of these nuclei can get across from one of the sections into another (in fact in unfortunate occurrences at the corners one nucleus can reach four neighbouring tiles. The main problems are the followings:

- Do not detect the same nucleus more than once in different neighbouring tiles.

- We also have to steer the case when parts of a single nucleus lying in side by side tiles and none of them fits for the criteria, thus the whole nucleus is lingering from the detection.

Fortunately during the cell nuclei segmentation we know the maximum radius of a single cell nucleus for a given zoom level. In the light of this information we are able to significantly decrease the number of problems and we can give a recommendation for the size of the overlapping regions. Since we know the maximum size of any cell nuclei (parameter R), then it is easy to see that it is advisable to complete the cut up procedure with the following parameters:

- The size of the tiles must be as large as possible. The main limit is that the tile must be processable (region growing) in one single step. We also have to steer the case when parts of a single nucleus lying on side by side tiles and none of them fits for the criteria, thus the whole nucleus is lingering from the detection.
- Let the size of the overlapping region between the neighbouring sections 4R in all directions (except the tiles on the edges).

Technically the implementation of the points mentioned before is the following: after the loading of the complete picture, an algorithm starts to partition it into smaller tiles according to the above and puts these slices (to be more precise the tasks for the processing of these slices) into a processing queue. Several independent processes can be started, which always get an element out from this queue, they are running the region growing algorithm onto that, then they are placing the results back to the object representing the appropriate task. The scheduler listens to the statement of the tasks after cutting the picture into pieces, and when each of them finished, and then it starts the next step, the merging of the intermediate results.

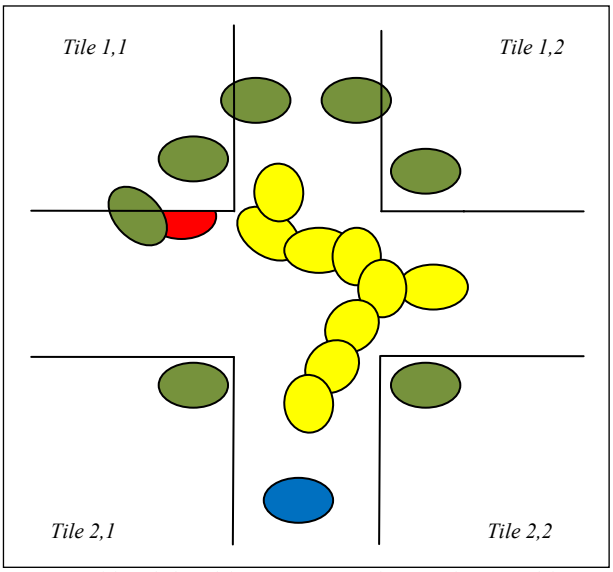

Fig. 2. Type of founded nuclei. Green ones: region growing started from a nonoverlapping area therefore these are accepted (red one will be deleted); Blue one: region growing started from the overlapping area but there is only one nucleus or more than nucleus with exact match (one of them is accepted); Yellow ones: region growing started from the overlapping area and these nuclei overlap each other (needs the backtracking search).

B. The merge phase

Using the methodology mentioned above, we can use the following method to merge the intermediate results into the final result:

- In case of every picture-section, those nuclei whose enlargement did not start from an overlapping area, we can accept them as valid nuclei without further checks. In case of two tiles side by side the distance between two seed points like that is going to become 4R at least. Therefore they cannot meet in the most unfortunate cases. This is true into all the four directions, so these cell nuclei can be handled as quite independent from the others in another picture slice, thus these are immediately acceptable.
- Processing overlapping regions
  - The processing method is the same for the independent (not overlapped with others) nuclei, in the merge phase we can accept them.
  - But this method is not valid in the case of overlapping cell nuclei. In the case of two overlapping neighbouring picture slice, there can be several nuclei, which have been detected in both parts. Since this overlap pointed out the pixels perfectly identical in both cases, we have to accept one of the nucleus candidates and reject the other one.

Those nuclei raise problems which are detected in different picture-slices and which are in only partial overlap. In the final result overlapping regions are not allowed, that is why the algorithm has to sort out a set of nuclei where neither overlaps the other. There are several solutions to this problem and we have to choose among the potential solutions which one presumes the largest summarized accuracy.

C. Selection of not overlapping nuclei

When all the picture-segments have been processed, we have to collect all overlapping nuclei into one set. In the first step we have to execute a clustering procedure to find the sets of cell nuclei in which the nuclei have some effect on each other. This is a classical component search of a graph where the nuclei are represented by the vertexes and there is an edge between two vertexes if the corresponding nuclei overlap each other. There are several clustering techniques, for instance in the first circle every nuclei have considered as separate to components, then in every iteration we look at the edges continuously and as an edge is connecting two different components, we merge these together. After the next iteration the remaining components represent sets of cell nuclei in which there is a way across the overlaps between every nucleus.

We have to post process these clusters to sort out one set of nuclei in which there are not any overlapping nuclei and the accuracy is maximal among all possible solutions. The accuracy of a cell nucleus set is the aggregate value of the accuracy of all the nuclei in the given set.

Where the accuracy is according to the following:

$$Score(X) = W_{size}*T_{size}[X_{size}]+W_{radius}*T_{radius}[X_{radius}]+ W_{circularity}*T_{circularity}[X_{circularity}]$$

Where

- W_{size}, W_{radius}, $W_{circularity}$: Weight factor for size, radius and circularity.
- T_{size}, T_{radius}, $T_{circularity}$: Density tables for size, radius and circularity (see above).
- X_{size}, X_{radius}, $X_{circularity}$: Size, radius and circularity properties of the X nucleus.

In the current phase, each of the weight factors are 1 equally, but in the near future it is advisable to refine these values. The correct filling of the density tables is far more important. For this reason, we have used the already existing values [7] based on the statistics of the Gold Standard slides (manually annotated by qualified pathologists). We have examined all of the nuclei in these samples and calculate the values before for each of them and drew the distribution. We separated the entire range of distribution into 100 equally sized intervals where every interval has the following values:

- $T_{size}[i]$ = number of nuclei in the given size interval / number of nuclei
- $T_{radius}[i]$ = number of nuclei in the given radius interval / number of nuclei
- $T_{circularity}[i]$ = number of nuclei in the given circularity interval / number of nuclei

To sum up it can be established that as more nuclei have found in interval i as great the T[i] value is. Therefore we can already use these values for the evaluation of the nuclei. We would like to give higher score values to nuclei similar to the reference nuclei (another approach is the usage of fuzzy operators [8]).

We have developed an algorithm to find the best set of not overlapping nuclei. The algorithm based on the backtrack method, where the number of subtasks equals to the number of nuclei in the cluster. Every subtasks represent the decision whether the corresponding nuclei is in the result set or not. The backtracking search examines all potential solutions quite efficiently, and then relying upon these findings it selects the combination of nuclei with the largest aggregate accuracy.

In case of overlapping nuclei the outcome is always the best combination. We merge these sets with the already found nuclei and this leads to the final result of the whole merge task.

III. EXAMINATION OF THE RESULTS

Unfortunately we do not have any full-sized annotated samples and we do not have any region growing algorithm implementation which could process a full-sized tissue sample (8092x8092). Therefore we can not use full-sized images as reference for the accuracy evaluation that is why we use the following method: we split already existing reference slides and choose a relatively small tile size. This is causeless because the region growing can process the whole picture in one step, but with this method we can compare the results of the new algorithm to the reference values.

As the results are indicating (TABLE I.), the split-and-merge technique does not cause significant degradation of the accuracy (the average difference between the original one-step processing method and the new merge and split method is about 0.37%). In case of great-sized pictures the ratio of the overlapping and the not overlapping areas is far more favourable, so we could expect better results. But of course it is an essential followup itself, that we can run the region growing algorithm in full-sized pictures at all.

TABLE I.

Comparison of the accuracy of the original region growing and the new split-and-merge based method.

#	Slide ID	Original accuracy	Split-merge accuracy	Difference
1	0259_ES_02	69.15%	69.70%	0.54%
2	0259_ES_03	76.29%	76.48%	0.18%
3	0259_PR_01	80.82%	80.95%	0.13%
4	0259_PR_02	83.06%	83.47%	0.42%
5	1031_ES_01	94.23%	94.11%	-0.11%
6	1031_ES_02	77.29%	77.52%	0.23%
7	1429_PR_02	70.27%	70.55%	0.27%
8	2167_ES_01	67.20%	67.15%	-0.04%
9	2167_ES_02	71.95%	71.96%	0.01%
10	2167_ES_03	75.83%	76.53%	0.70%
11	2224_PR_01	80.83%	80.71%	-0.12%
12	2224_PR_02	80.21%	80.58%	0.37%
13	2225_ES_01	84.22%	84.36%	0.14%
14	2225_ES_02	81.43%	81.46%	0.03%
15	2225_ES_03	80.52%	80.40%	-0.12%
16	2508_PR_01	83.60%	83.67%	0.07%
17	2508_PR_02	80.72%	79.83%	-0.89%
18	2819_ES_01	68.73%	68.70%	-0.03%
19	2819_ES_02	76.97%	77.17%	0.20%
20	2819_ES_03	63.79%	64.15%	0.36%
21	2819_PR_01	77.25%	78.31%	1.06%
22	2819_PR_02	80.31%	80.48%	0.17%
23	2819_PR_03	76.38%	76.58%	0.20%
24	2856_ES_01	67.80%	68.80%	1.00%
25	2856_ES_02	75.23%	75.98%	0.75%
26	2857_ES_01	92.47%	91.36%	-1.12%
27	2857_ES_02	81.86%	80.53%	-1.33%
28	2857_ES_03	83.10%	82.53%	-0.57%
29	2924_PR_01	89.14%	88.20%	-0.94%
30	2924_PR_02	83.20%	82.24%	-0.96%
31	2924_PR_03	75.72%	74.50%	-1.22%

32	3019_PR_01	82.67%	82.64%	-0.02%
33	3019_PR_02	80.91%	80.53%	-0.38%
34	3019_PR_03	93.06%	93.02%	-0.04%
35	3381_ES_01	81.39%	81.39%	0.00%
36	3381_ES_03	75.63%	75.63%	0.00%
37	3381_PR_01	65.32%	65.24%	-0.08%
38	3381_PR_02	72.38%	72.34%	-0.04%
39	3381_PR_03	70.68%	70.68%	0.00%

During the development it was an important consideration that the final algorithm must be well parallelizable to utilize the possibilities of the distributed architectures. Processing the individual picture-sections and the merge of the nuclei candidates are both very resource-demanding operations. Therefore it is a tangible benefit that we can write parallel codes to utilize the multi-core architectures. Processing the tissue sections is obviously parallelizable, moreover the procedure to find the best set of non overlapping nuclei is also executable in multiple threads: each thread can work with one nuclei set to find out the best combination (we can utilize this feature with multi-core CPUs and multi-GPU environments, our next paper will contains the experimental results about this).

TABLE II.

Runtime of the main steps of the split-and-merge algorithm for images with different dimensions (2048x2048pixel, 4096x4096pixel)

Slide ID	Dim	Split (ms)	Region Growing (ms)	Merge (ms)
10359-04ep	2048	437	95394	0
10359-04ep	4096	2028	194532	2730
1050-04IIadenomavill+dyspl	2048	359	75691	0
1050-04IIadenomavill+dyspl	4096	2044	391997	6022
1160-05CRCA-B	2048	452	129527	0
1160-05CRCA-B	4096	2012	367615	6833
12138-03Adenomavillosum	2048	421	113100	0
12138-03Adenomavillosum	4096	2090	215764	4571
12532-04CRCA-B	2048	421	103709	0
12532-04CRCA-B	4096	1997	282772	4805
2877-04Ihyperpl	2048	421	94240	0
2877-04IHyperpl	4096	2137	438642	11326
6134-04p	2048	437	94146	0
6134-04p	4096	1966	184344	7301
8658-04IHyperpl	2048	452	115643	0
8658-04IHyperpl	4096	2636	460497	12215
986604Chron	2048	421	114286	0
986604Chron	4096	1966	279771	8315
986604Crohn	2048	437	94349	0
986604Crohn	4096	2153	356632	9610

We have examined the speed loss caused by the merge and split procedures. TABLE II. shows the execution time of some tissue images (these are not the same as in the previous table). The split part consists of the following steps: loading the image, split it into smaller images and save these images as separate files (most of the required time caused by file operations). The region growing part is the runtime of the old region growing algorithm for the new images. The merge part consists of the following steps: collecting nuclei candidates, finding the optimal combinations. The execution parameters were the recommended values (dimension of the tiles is 2048x2048 pixels, overlap size is 128 pixels). Obviously the runtime of the merge and split phases is insignificant compared to the runtime of the region growing. It is worth mentioning that in case of small images (processed by one title) the split-and-merge runtime is not a drawback.

REFERENCES

[1] J. Tick, A. Tick, “Business Process Modeling - Simulation of Administrative Activities”, ICCC 2013, Proceedings of IEEE 9th International Conference on Computational Cybernetics, Tihany, Hungary, 2013. pp. 345-348.

[2] Cseri, O. E.; Kerti, A.; Vámossy, Z., "3-D reconstruction system," 7th International Symposium on Intelligent Systems and Informatics, 2009. SISY '09., 2009, pp.175,179

[3] Sergyan, S., "Useful and effective feature descriptors in content-based image retrieval of thermal images," 4th IEEE International Symposium on Logistics and Industrial Informatics (LINDI), 2012, pp.55-58

[4] Szénási, S.; Vámossy, Z.; Kozlovszky, M., "GPGPU-based data parallel region growing algorithm for cell nuclei detection," IEEE 12th International Symposium on Computational Intelligence and Informatics (CINTI), 2011, pp.493,499, 21-22 Nov. 2011

[5] M. Sonka, V. Hlavac, R. Boyle, “Image Processing, Analysis, and Machine Vision, 3rd edition,” Thomson Learning, 2007

[6] Robert M. Haralick, Linda G. Shapiro, “Image segmentation techniques, Computer Vision, Graphics, and Image Processing,”, vol. 29, no. 1, 1985, pp. 100-132.

[7] S. Szénási, Z. Vámossy, M. Kozlovszky, "Preparing initial population of genetic algorithm for region growing parameter optimization", 4th IEEE International Symposium on Logistics and Industrial Informatics (LINDI), 2012, 5-7 Sept. 2012, pp. 47-54.

[8] E. Tóth-Laufer, M. Takács, “Comparative Study of Fuzzy Operators in Risk Level Calculation,”, 11th International Conference on Global Research and Education in Engineers for Better Life, Budapest, Hungary, 2012, ISBN:9786155018374, pp. 237-246.