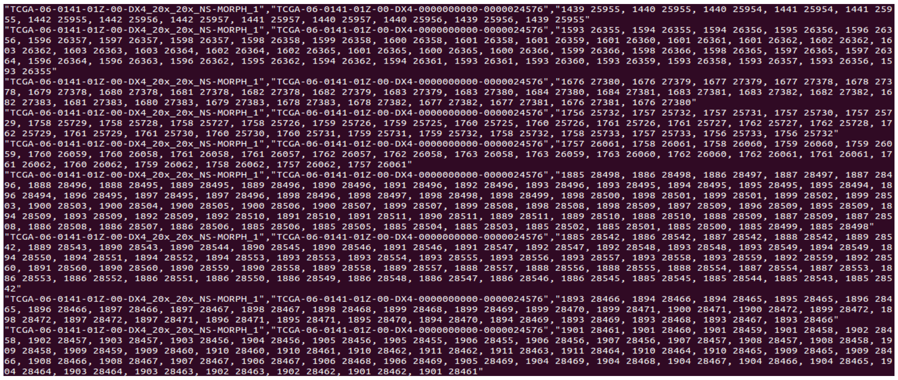
CS554 Database Systems Project Report ***Kai Wang***

Discovery of Spatial Density Patterns in Large Scale Medical Images

Abstract

The project aims to detect regions that have significant and different density compared with its neighborhood. I summarized significant-different regions into three categories: 1) Over-Density, 2) Under-Density, 3) regions surrounded by Ridges/Ditches. The third one may capture regions surrounded with a very thin edge, either a ridge or ditch, which may be easily missed out. Significant-different region detection has many useful applications. It can be applied to detect the outbreak area of a possible epidemic where the number of disease cases is significantly larger than expected , identify crime hot-spots/cold-spots where crime rates are unusually high/low, detect anomaly regions of organs, where the density of cells is much higher/lower than its surroundings.

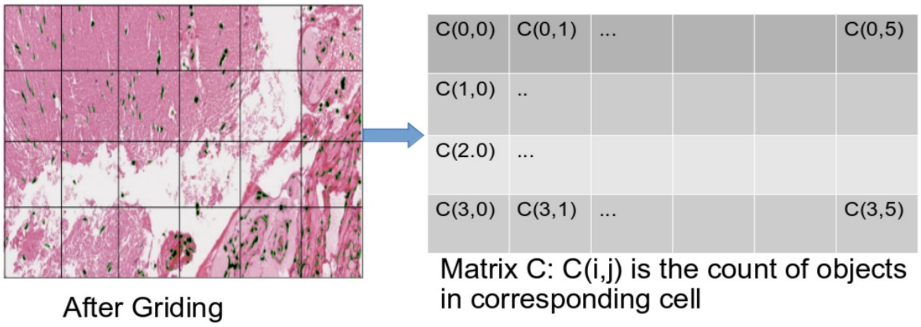
Specifically, this project is targeted at the whole slide image from PAIS dataset. The original data (*Figure 1*) I got is the text files exported from databases, with each line representing the boundary of a biological object such as cell, vessel, etc. The boundaries are composed of a sequence of points with the first point same to the last. Then, my goal in this setting is to detect regions that have significant-different object density.



```
"TCGA-06-0141-01Z-00-DX4_20x_20x_NS-MORPH_1","TCGA-06-0141-01Z-00-DX4-0000000000-0000024576","1439 25955, 1440 25955, 1440 25954, 1441 25954, 1441 259
55, 1442 25955, 1442 25956, 1442 25957, 1441 25957, 1440 25957, 1440 25956, 1439 25956, 1439 25955"
"TCGA-06-0141-01Z-00-DX4_20x_20x_NS-MORPH_1","TCGA-06-0141-01Z-00-DX4-0000000000-0000024576","1593 26355, 1594 26355, 1594 26356, 1595 26356, 1596 263
56, 1596 26357, 1597 26357, 1598 26357, 1598 26358, 1599 26358, 1600 26358, 1601 26358, 1601 26359, 1601 26360, 1601 26361, 1601 26362, 1602 26362, 16
03 26362, 1603 26363, 1603 26364, 1602 26364, 1602 26365, 1601 26365, 1600 26365, 1600 26366, 1599 26366, 1598 26366, 1598 26365, 1597 26365, 1597 263
64, 1596 26364, 1596 26363, 1596 26362, 1595 26362, 1594 26362, 1594 26361, 1593 26361, 1593 26360, 1593 26359, 1593 26358, 1593 26357, 1593 26356, 15
93 26355"
"TCGA-06-0141-01Z-00-DX4_20x_20x_NS-MORPH_1","TCGA-06-0141-01Z-00-DX4-0000000000-0000024576","1676 27380, 1676 27379, 1677 27379, 1677 27378, 1678 273
78, 1679 27378, 1680 27378, 1681 27378, 1682 27378, 1682 27379, 1683 27379, 1683 27380, 1684 27380, 1684 27381, 1683 27381, 1683 27382, 1682 27382, 16
82 27383, 1681 27383, 1680 27383, 1679 27383, 1678 27383, 1678 27382, 1677 27382, 1677 27381, 1676 27381, 1676 27380"
"TCGA-06-0141-01Z-00-DX4_20x_20x_NS-MORPH_1","TCGA-06-0141-01Z-00-DX4-0000000000-0000024576","1756 25732, 1757 25732, 1757 25731, 1757 25730, 1757 257
29, 1758 25729, 1758 25728, 1758 25727, 1758 25726, 1759 25726, 1759 25725, 1760 25725, 1760 25726, 1761 25726, 1761 25727, 1762 25727, 1762 25728, 17
62 25729, 1761 25729, 1761 25730, 1760 25730, 1760 25731, 1759 25731, 1759 25732, 1758 25732, 1758 25733, 1757 25733, 1756 25733, 1756 25732"
"TCGA-06-0141-01Z-00-DX4_20x_20x_NS-MORPH_1","TCGA-06-0141-01Z-00-DX4-0000000000-0000024576","1757 26061, 1758 26061, 1758 26060, 1759 26060, 1759 260
59, 1760 26059, 1760 26058, 1761 26058, 1761 26057, 1762 26057, 1762 26058, 1763 26058, 1763 26059, 1763 26060, 1762 26060, 1762 26061, 1761 26061, 17
61 26062, 1760 26062, 1759 26062, 1758 26062, 1757 26062, 1757 26061"
"TCGA-06-0141-01Z-00-DX4_20x_20x_NS-MORPH_1","TCGA-06-0141-01Z-00-DX4-0000000000-0000024576","1885 28498, 1886 28498, 1886 28497, 1887 28497, 1887 284
96, 1888 28496, 1888 28495, 1889 28495, 1889 28496, 1890 28496, 1891 28496, 1892 28496, 1893 28496, 1893 28495, 1894 28495, 1895 28495, 1895 28494, 18
96 28494, 1896 28495, 1897 28495, 1897 28496, 1898 28496, 1898 28497, 1898 28498, 1898 28499, 1898 28500, 1898 28501, 1899 28501, 1899 28502, 1899 285
03, 1900 28503, 1900 28504, 1900 28505, 1900 28506, 1900 28507, 1899 28507, 1899 28508, 1898 28508, 1898 28509, 1897 28509, 1896 28509, 1895 28509, 18
94 28509, 1893 28509, 1892 28509, 1892 28510, 1891 28510, 1891 28511, 1890 28511, 1889 28511, 1889 28510, 1888 28510, 1888 28509, 1887 28509, 1887 285
08, 1886 28508, 1886 28507, 1886 28506, 1885 28506, 1885 28505, 1885 28504, 1885 28503, 1885 28502, 1885 28501, 1885 28500, 1885 28499, 1885 28498"
"TCGA-06-0141-01Z-00-DX4_20x_20x_NS-MORPH_1","TCGA-06-0141-01Z-00-DX4-0000000000-0000024576","1885 28542, 1886 28542, 1887 28542, 1888 28542, 1889 285
42, 1889 28543, 1890 28543, 1890 28544, 1890 28545, 1890 28546, 1891 28546, 1891 28547, 1892 28547, 1892 28548, 1893 28548, 1893 28549, 1894 28549, 18
94 28550, 1894 28551, 1894 28552, 1894 28553, 1893 28553, 1893 28554, 1893 28555, 1893 28556, 1893 28557, 1893 28558, 1893 28559, 1892 28559, 1892 285
60, 1891 28560, 1890 28560, 1890 28559, 1890 28558, 1889 28558, 1889 28557, 1888 28557, 1888 28556, 1888 28555, 1888 28554, 1887 28554, 1887 28553, 18
86 28553, 1886 28552, 1886 28551, 1886 28550, 1886 28549, 1886 28548, 1886 28547, 1886 28546, 1886 28545, 1885 28545, 1885 28544, 1885 28543, 1885 285
42"
"TCGA-06-0141-01Z-00-DX4_20x_20x_NS-MORPH_1","TCGA-06-0141-01Z-00-DX4-0000000000-0000024576","1893 28466, 1894 28466, 1894 28465, 1895 28465, 1896 284
65, 1896 28466, 1897 28466, 1897 28467, 1898 28467, 1898 28468, 1899 28468, 1899 28469, 1899 28470, 1899 28471, 1900 28471, 1900 28472, 1899 28472, 18
98 28472, 1897 28472, 1897 28471, 1896 28471, 1895 28471, 1895 28470, 1894 28470, 1894 28469, 1893 28469, 1893 28468, 1893 28467, 1893 28466"
"TCGA-06-0141-01Z-00-DX4_20x_20x_NS-MORPH_1","TCGA-06-0141-01Z-00-DX4-0000000000-0000024576","1901 28461, 1901 28460, 1901 28459, 1901 28458, 1902 284
58, 1902 28457, 1903 28457, 1903 28456, 1904 28456, 1905 28456, 1905 28455, 1906 28455, 1906 28456, 1907 28456, 1907 28457, 1908 28457, 1908 28458, 19
09 28458, 1909 28459, 1909 28460, 1910 28460, 1910 28461, 1910 28462, 1911 28462, 1911 28463, 1911 28464, 1910 28464, 1910 28465, 1909 28465, 1909 284
66, 1908 28466, 1908 28467, 1907 28467, 1906 28467, 1906 28468, 1906 28469, 1905 28469, 1904 28469, 1904 28468, 1904 28467, 1904 28466, 1904 28465, 19
04 28464, 1903 28464, 1903 28463, 1902 28463, 1902 28462, 1901 28462, 1901 28461"
```

Figure 1. Dataset: Boundaries of Objects

At the beginning, the original textual object boundary dataset is partitioned into an M*N grid and transformed into a density matrix, where each entry *(i, j)* represents the count of objects inside *(i,j)th* cell. It's shown in *Figure 2*. Based on this density matrix, which is processed as an intensity image, image processing techniques such as image morphological operators and other image adjust and enhancement techniques are applied and then a region growing based algorithm is applied to extract the significant-different regions. Compared with other techniques, which only output a rough rectangular bounding box, this algorithm can return the exact shape of the significant-different regions.



C(0,0)	C(0,1)	...			C(0,5)
C(1,0)	..				
C(2.0)	...				
C(3,0)	C(3,1)	...			C(3,5)

Matrix C: C(i,j) is the count of objects in corresponding cell
After Griding

Figure 2. Intensity Matrix

Related Work

I summarized the relevant work into two groups: traditional density clustering based algorithms and grid-based algorithms.

1. Traditional Density Clustering Based Algorithms

Some density clustering based approaches for spatial objects include CLARANS, BIRCH and DBSCAN. These kind of algorithms usually have a time complexity of at least N^2, where N is the total number of objects. For my project, this is unacceptable due to the large number of objects to process. A textual file for a single tile, from which the graphs in *Figure 1* were generated, can have size near 1 GB, for example. Second, it also deviates from my goal to find the region with significant-different density statistics. It's natural to solve this problem in a more abstract and higher level.

2. Grid-based Approaches

Instead of processing the dataset directly, this category of methods first partition the space into a grid with uniform-sized cells. It computes the summarized statistics for each cell. Typically, the statistics include the total number of objects inside. But for data from different domains, for example, min, max, mean, standard deviation, etc [3], if the objects have applicable numeric attributes. Then the clustering process is based on this grid data. Grid-based approaches have the advantage of efficiency since it deals with summarized data. Actually, for my project, it can offer suitable granularity and avoid too much details and noises with a good grid size. Relevant grid-based approaches include GridScan [1], Sting [3], WaveCluster [4], continuous density queries for moving objects [5].

GridScan: The data processed in the GridScan paper [1] is a little different. It contains two kinds of points. namely case points and control points. So the definition of density is based on the ratio of case points to control points. The main idea of this method is that given a grid, it grows the current region with its neighbor cells ***c*** and merge current region with other regions if they can be connected by ***c*** and the log likelihood ***L*** of merged region is higher than ***L*** of both regions. The region growing algorithm is greedy, which means that it always tries to grow current region with the neighboring cell that maximize ***L***. Besides, it also computes the p-value to evaluate the statistical significance of extracted clusters by re-sampling.

Sting: Sting is the first work that give an insight for the grid-based spatial data mining. It uses grids on multiple levels and store summarized statistical information on each level. It's designed for answering SQL like queries and didn't actually do the clustering process.

WaveCluster: WaveCluster is a grid based approach that uses wavelet transform to extract homo-density areas. This method doesn't have any assumptions and works with any clusters of any shapes. When applying to images, it can use different filters, e.g. high-pass, low-pass filters, to extract details and remove outliers. By down-sampling after each iteration of wavelet transform, WaveCluster can achieve a clustering process at different granularity levels.

Continuous density queries for moving objects: The difference from above approaches is that this paper [5] addresses the density queries, over density and sparse density for moving objects rather than static points. Their approach applies two indexing structures, namely Quad-tree and TPR-tree to accelerate query processing. They introduced the notion of safe interval for continuous query answering, which enables dynamic maintenance of region states .

Rapid Over-density Detection: In [6][7], they introduced a rapid over-density detection method based on d-dimensional grid. Compared with the original *spatial scan statistic (Kulldorff, 1997)* that is computational infeasible for large spatial regions, the new method can achieve up to 1400x speedup and is very scalable. They built an overlap-kd tree to index the spatial regions in a hierarchical way. Compared with ordinary kd-tree, the overlap one can detect dense regions that spread over several adjacent regions. Tree pruning is applied when searching through the tree for acceleration. For each dataset, it contains two kinds of n-dimensional points, namely case point and baseline point. The ratio of count of case points to baseline points in each region is used to determine degree of over-density. Statistical importance is tested by re-sampling baseline points. The overlap technique [6,7] applied in kd-tree structure is unnecessary in region growing based approach since it can merge two splits together.

Proposed Approach: Region Growing based Algorithm

As discussed, the size of a single text file in the original dataset often in the order of hundreds of megabytes and the over-detailed information such as the boundaries of small cells is not necessary actually in this project. In fact, when we manually annotate the significant-different regions, we don't necessarily to observe the image tiles in full resolution. The objects are already like points at a suitable zoom level. Therefore, I don't consider the area of objects here, which I believe won't add more additional benefit. So, the data space is divided into a M by N grid at first, with the aggregate statistics stored for each cell in a matrix (*Figure 2*). Then a region growing based algorithm is executed to merge homogeneous neighboring cells into one big region. An advantage of region growing based method is that it can handle different cases such as over-density, under-density and regions contains ridges, at the same time. Important issues to consider for region growing include 1) selection of seed points, 2) minimum area threshold, 3) similarity threshold, etc.

Each separate text file in the dataset is composed of object boundaries inside a tile, which is often of a size around 5000 pixels in each dimension. In the griding process, the number of objects in each cell is calculated without consideration of area property. *Figure 3* is the gray image with each pixel value as the number of objects in corresponding cell. The white rectangle is the hand marked region. As we can see, the significant-different region is where the intensity of the image changes relatively great.

Description of Proposed Procedure

1. Prepare the density image:
   a) One pass through the dataset to get the maximum space ranges (max_width and max_height) of objects and the domain of x-y coordinates of all objects.
   b) Grid the whole image space with cell size (alpha*max_width, beta*max_height) and count the number of objects in each cell. Thus we get a intensity matrix with each entry equals the count of objects in corresponding cell. This part is done by the python scripts, *img.py* or

img_mark.py. The latter one also included the bounding box as in *Figure 3*.

c) Convert this matrix into gray-level image, *Figure 3*.

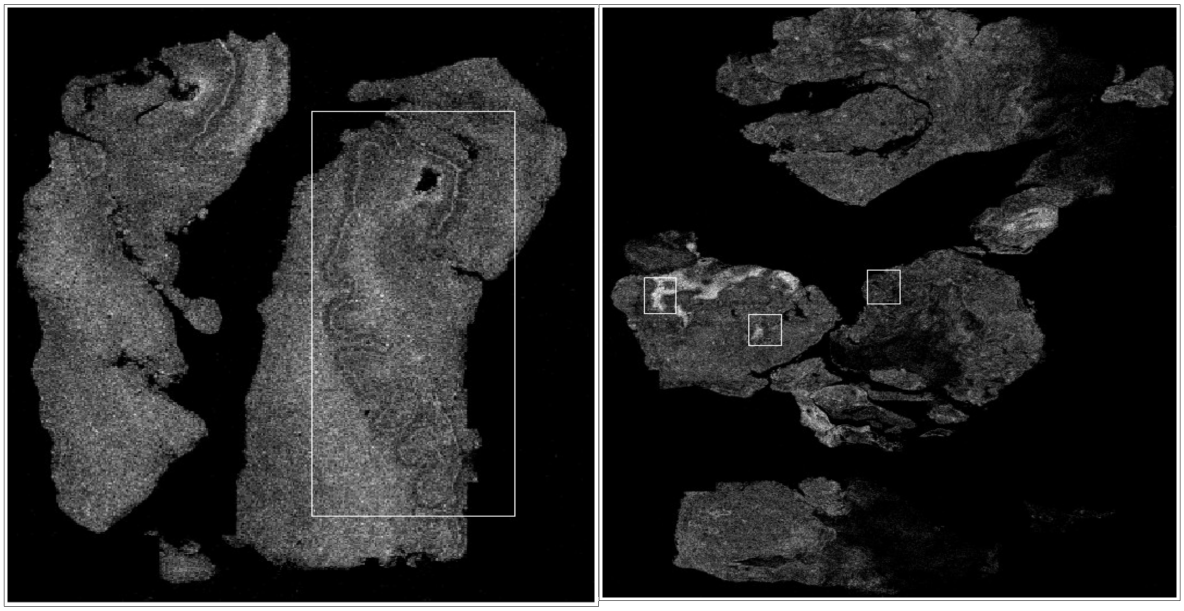


Figure 3. Example of annotated abnormal region (transformed gray image)

2. Region Growing based Detection

Code of this part is written in *Matlab*. For details of the algorithm, please refer the .m files, which have detailed documentation. Below are some issues to be considered:

a) Improve the density image:

Apply image processing techniques such as the morphological operators dilation, erosion, open, close to ***remove noises*** like very small regions and ***increase contrast*** between adjacent heterogeneous regions and ***connect separate homogeneous regions that are near to each other*** and ***smooth the image***.

Here are some useful Matlab functions I experimented with for your reference: imadjust, imdilate, imerosion, imopen, imclose, strel, histeq, brighten, strechlim, fspecial, imfilter, imreconstruction, imareaopen, imsharpen, **imfill** (remove holes), **imboundaries**, bwconncomp (neighborhood is only defined as the pixels directly ajacent, not suitable for density-based clustering), etc.

b) Rule out background and normal regions. We only want to detect the rapidly changing regions within organs rather than the edges of organs or the boundaries of normal organ regions, or the noisy fading regions. Some hints are used in a post-processing step such as average intensity values and total area of the extracted region to exclude undesirable results.

c) Hierarchical detection by down-sampling. Down-sampling with a factor ***(x,y)*** from current density image to get a more abstract image, smaller in resolution. Each pixel in the down-sampled image is the average of the x*y pixels in its corresponding rectangular region. For each down-sampled cell, calculate the statistics such as max, min, average, standard deviation as cell properties from the parent cells. These statistics vector stored in this hierarchical indexing structure are used in the region growing process. However, simple count information is proved to work great enough in experiments.

d) Seed selection. In the implementation, both automatic seed selection and interactive thresholding is provided. For the automatic seeding, seed points for region growing is selected according to the low, high thresholds parameters. Automatic threshold seeding allows users to interactively set the seed points.

e) Adaptive thresholding: Otsu's method for Adaptive Distance Threshold

Experiments

This section shows the results of results of the proposed region growing based algorithm, and watershed and hierarchical watershed or waterfall algorithm for comparison purposes.

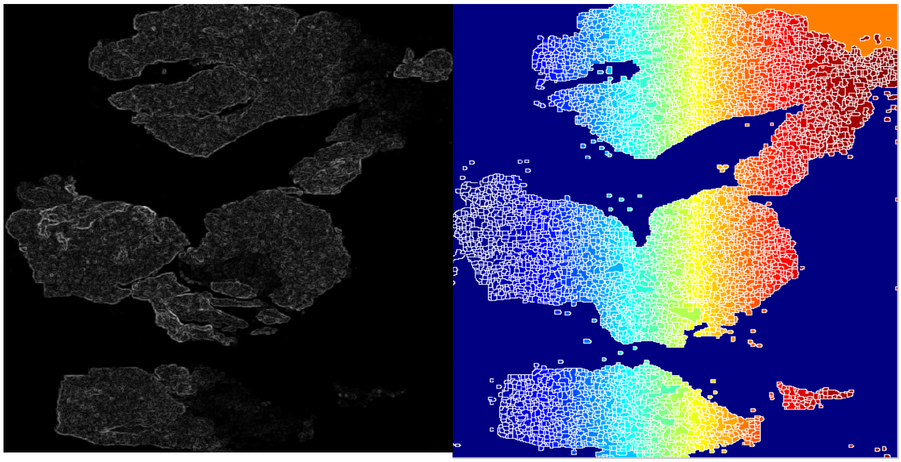


Figure 4. Over Segmentation of Watershed Algorithm

In *Figure 4*, we can see that watershed algorithm suffers a lot from over-segmentation. It cannot discriminate the over-density and under-density regions from normal organ regions. Besides, it would extract the whole organ out instead of the significant-different parts of an organ. It has no clue of the existence of the dark background. *Figure 5* shows the results of hierarchically applying the *watershed algorithm*. Based on reason just discussed, it's still unsatisfactory.

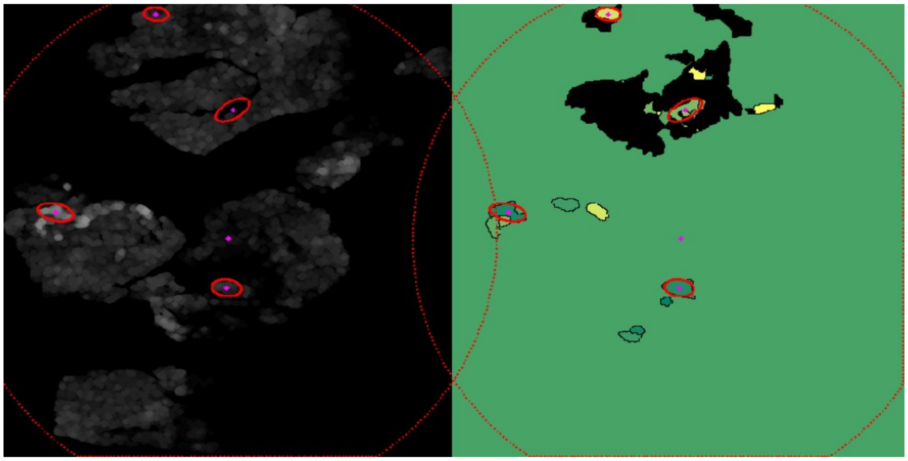


Figure 5. Hierarchical Watershed Algorithm

Results of Proposed Algorithm

The results are mainly evaluated by visual examination since hand-annotations are limited and incomplete. Some other objective measures are also available and listed in later part.

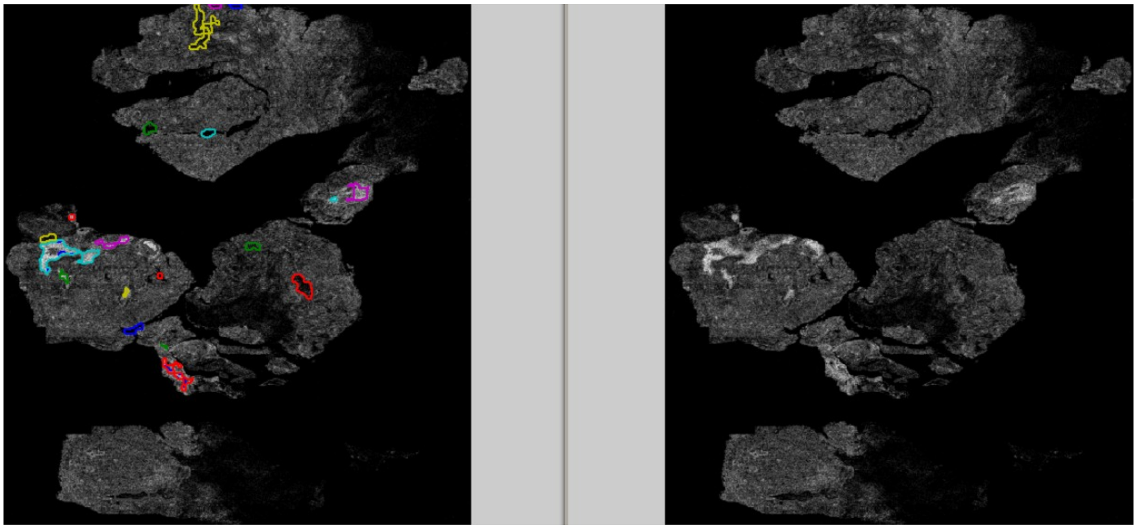


Figure 6. Sample Result 1 of proposed algorithm

Figure 6 shows great detection results of proposed region growing based algorithm. As shown, not only can it extract the much brighter regions (over-density), this approach also detected the darker holes inside the organs. Algorithm parameters are stable and easy to determine and explained in detail in source code. Better results can be achieved with simple parameter tuning.

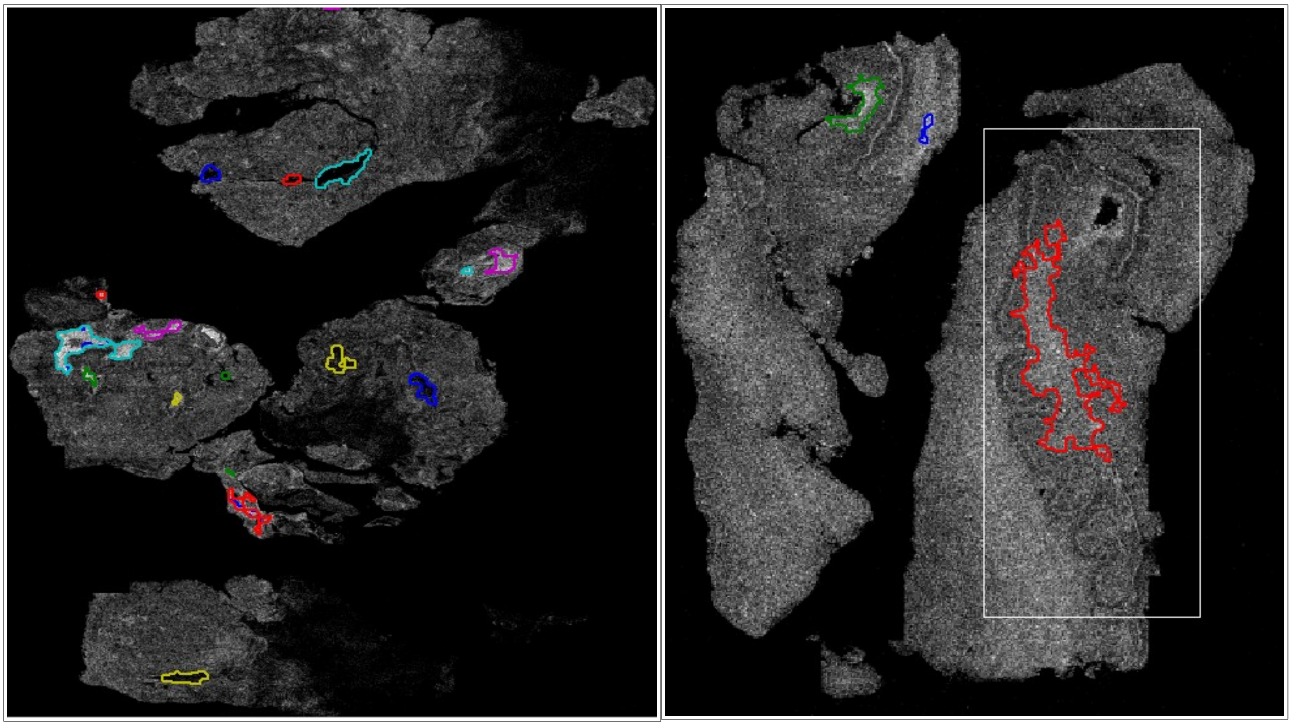


Figure 7. Sample Result 2 of proposed algorithm

In Figure 7, the left part is the result of detecting both under-density and over-density at the same time. In the right half, we can see that the amount hand-annotated regions by my classmate Dejun is very small. For each annotated picture, only a few markups are available. My algorithm not only detected the region annotated by hand, but also found other significantly brighter regions as the blue and green boundaries show. Moreover, the regions are more accurate in my algorithm, which reflects their actual shape, while the hand-annotations are only rectangles.

For users who prefer the results to be presented in forms of rectangular minimum bounding box (MMB), the output can be converted to MMB very easily.

Optional Evaluation Standards

1) Global segmentation evaluation index, Liu's F-factor [10]

$$F(I) = \sqrt{R} \times \sum e_i^2 / \sqrt{A_i}$$

2) recall rate compared with hand-annotated regions

Conclusion

The experimental results show that the proposed region growing based algorithm works great in detecting significant-different regions with efficient computational power. Different from density-based clustering algorithm, my approach construct an M*N intensity image after partitioning the original data into M*N cells. M and N is generally in the order of 10^2. Therefore, the algorithm is very fast to run and can be above 10^3 speedups. Meanwhile, the regions extracted can be any shape and reflects the real shape of the significant-different regions, rather than rough specific shapes such as rectangular bounding box or circular regions.

References

[1] Weishan Dong , Xin Zhang , Li Li , Changhua Sun , Lei Shi , Wei Sun, Detecting Irregularly Shaped Significant Spatial and Spatio-Temporal Clusters

[2] http://en.wikipedia.org/wiki/Region_growing

[3] STING : A Statistical Information Grid Approach to Spatial Data Mining

[4] WaveCluster: A Multi-Resolution Clustering Approach for Very Large Spatial Databases

[5] Jie Wen, Xiaofeng Meng, Xing Hao, Jianliang Xu, An efficient approach for continuous density queries

[6] Daniel B. Neill, Andrew W. Moore , Rapid Detection of Significant Spatial Clusters

[7] Daniel B. Neill, Andrew W. Moore, Francisco Pereira, and Tom Mitchell , Detecting Significant Multidimensional Spatial Clusters

[8] Serge Beucher , watershed, hierarchical segmentation and waterfall algorithm

[9] Om Prakash Verma, Madasu Hanmandlu, Seba Susan, Muralidhar Kulkarni and Puneet Kumar Jain, A Simple Single Seeded Region Growing Algorithm for Color Image Segmentation using Adaptive Thresholding

[10] Jianqing Liu , Yee-Hong Yang,” Multiresolution Color Image Segmentation” IEEE Transaction on pattern analysis and machine intelligence Vol no. 7, JULY 1994

[11]Otsu, N., "A Threshold Selection Method from Gray-Level Histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 9, No. 1, 1979, pp. 62-66.