Using Image Processing Functions to Determine the Edges of the Dead Sea and Calculate the Declining Rate

Mua'ad M. Abu-Faraj[1] and Nazeeh A. Ghatasheh[2]

[1]Department of Computer Information Systems, The University of Jordan
Aqaba, 77110, Jordan

[2]Department of Business Information Technology, The University of Jordan
Aqaba, 77110, Jordan

Abstract

Satellite images provide the researchers with the ability to identify objects (such as roads, land use, elevation, waterways, etc.). Analyzing the satellite images of the Dead Sea in Jordan can help determining the edges of the Dead Sea and its declining rate. The analysis process can be done using evolving algorithms and software. This paper shows the outcome of applying three matlab image-processing functions on the Dead Sea images taken from Google Earth. The three functions are: threshold, edge-based and watershed segmentation. Determining the edge of the Dead Sea using image processing functions can be used to calculate, predict and forecast the declining rate of the Dead Sea. We also present a case study showing our analysis.

Keywords: *Image Processing, Threshold, Edge-Based, Watershed, Segmentation.*

1. Introduction

The Dead Sea has a cultural, economical, and historical importance locally and globally. Moreover it is characterized by being the lowest point on earth and the most saline natural water resource. Due to the significant water supply shortage going into the Dead Sea, it lost almost the third of its water quantity between 1960 and 2012. Serious decline of water level is considered to be an economical threat that the beneficial countries are trying to overcome. From this brief overview of the Dead Sea state, we can realize the importance of monitoring the water level in order to evaluate the outcomes of the remedy actions or put in hand the present situation [1].

Satellite images provide important information on earth surface, geographic area, weather and naturel phenomena. Satellites have been used over the past several decades to obtain a wide variety of information about the earth's surface, ranging from military applications to tracking global weather patterns, tectonic activity, surface vegetation, ocean currents and temperatures, polar ice fluctuations, pollution, and many other aspects [2]-[5]. The application of satellite image analysis to archeology has emerged alongside these other uses, but archaeologists are only now beginning to exploit more fully broad range of analytical tools available for assessing the satellite image data of the earth's surface and sub-surface [6].

Satellite imagery provides effective means of observing and quantifying the complexities of the surface of the earth. It allows us to see the world in a different way and provides a huge information source at our fingertips as we look to increase our knowledge about the environment. The technologies behind the application of this imagery are mature, and evolving rapidly.

2. Data and Methodology

The Satellite images are taken from satellite (for example NASAs) work with color values, the original image was segmented by threshold, edge-based segmentation, and watershed segmentation, the original image is shown in Fig.1.

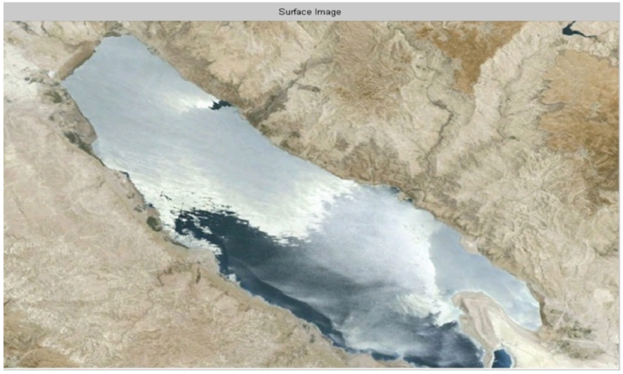


Fig. 1 The Original Image.

Matlab software was used to implement the algorithm, since Matlab is a high-performance language for education and research as it integrates computation, visualization and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation and also it has toolboxes for signal processing, neural network, image processing, database etc. Matlab Image Processing Toolbox is a collection of functions that extend the capability of the Matlab numeric-computing environment. The toolbox supports a wide range of image processing operations, such as Image analysis and enhancement. Region of interest operations, linear filtering and filter design [7]. All the functions and equations used in this study are from Matlab image processing toolbox.

3. Related Work

In [9], the author proposed an approach for image segmentation named Differential Immune Clone Clustering Algorithm (DICCA). That includes texture image segmentation and feature extraction, while watershed image segmentation used for its efficiency. The authors tested their approach on different types of images from which remote sensing images that are considered in our research. In order to use watershed, the author in [9]. followed few preprocessing steps that are simplifying the images and producing a gradient image. After the watershed preprocessing output was used as an input to the DICCA. Their findings on the images, that represent water/land combination, show that there is a need for visual measurement to describe the results. Also an interesting finding was the great processing time reduction on remote sensing images, and other types of images, by using watershed.

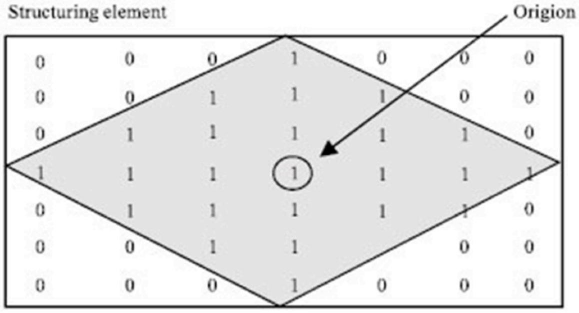
Automatic rule based spatiotemporal knowledge discovery is an interesting work by [10]. As they introduced Case-Based Reasoning (CBR) method which tries to simulates human reasoning. In terms of remote sensing images, their approach combines different measures for knowledge discovery, and could be applied to different environmental features. An example of application is to find out a model that represents a lake, then analyzing a large set of database data to find spatiotemporal changes relying on the lake model. So the changes over time to the lake could be discovered and reported.

4. Discussion and Results

The first step of our proposed software is to read the image by Matlab software and convert it into a matrix of values, each value represents two-dimensional light intensity function f (x, y), where x and y denote spatial coordinates and the value of f at any point (x, y). The image contains many objects of different sizes that are touching each other.
The next step aims to create the structure-element, which is needed to apply dilation and erosion operations. The structure-element is used also to probe the input image. The structure-element is a matrix that consists of 0s and 1s. It can have any arbitrary shape and size. Matlab functions are used to create the structure-element as given by the by Eq.(1):

$$origin = floor((size(nhood) + 1) / 2) \qquad (1)$$

where nhood is the neighborhood defining the structure-element . The structure-element is shown in Fig.2.



0	0	0	1	0	0	0
0	0	1	1	1	0	0
0	1	1	1	1	1	0
1	1	1	1	1	1	1
0	1	1	1	1	1	0
0	0	1	1		0	0
0	0	0	1	0	0	0

Fig. 2 Diamond Structure Element.

Dilation and erosion functions are used to accept structure-element objects, these objects are called STRELs. STREL function is used to create diamond shape structure-element. In [8], the author has proposed a technique known as structure-element decomposition, in which the structure-element is broken into smaller pieces in order to enhance the performance.

Enhancing the image contrast is the next step in our proposed software. The Matlab functions (from the image processing toolbox) imtophat and imbothat are used to perform the top-hat and bottom-hat transformations of the original image, respectively. These transformation can be performed using Eq.(2) and Eq.(3) [8]. The output of these transformations is shown in Fig.3 and Fig.4.

$$Ibot = imbothat(a\,f\,m;\, se) \qquad (2)$$
$$Itop = imtophat(a\,f\,m;\, se) \qquad (3)$$

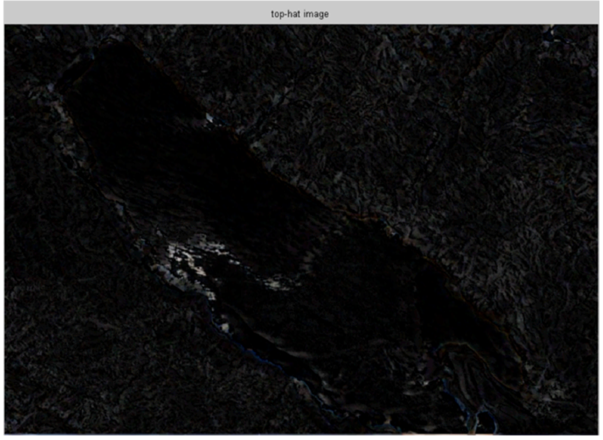


Fig. 3 Bottom-Hat Image.

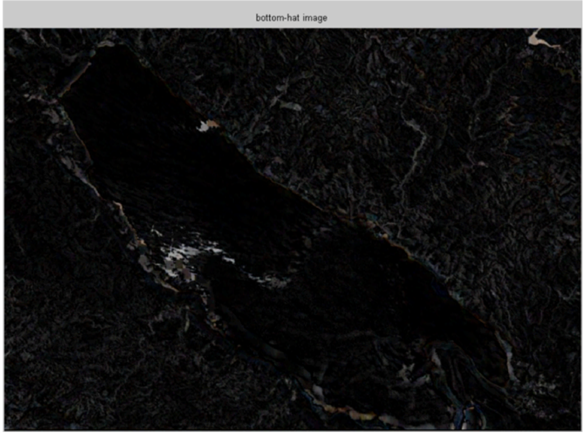


Fig. 4 Top-Hat Image.

Step 4 involves exaggerating the gaps between objects. The top-hat image contains the peaks of objects that fit the structure-element. This can be done using the imbothat function, which shows the gaps between the objects. To maximize the contrast between the objects and the gaps that separate them from each other, we added the top-hat image to the original image and then subtract the bottom-hat image from the result. We used the toolbox image arithmetic functions, imadd and imsubtract, to perform these operations as shown in Eq.(4).

$$Ienhance = imsubtract(\ imadd(\ Itop;\ a\,f\,m\ );\ Ibot) \quad (4)$$

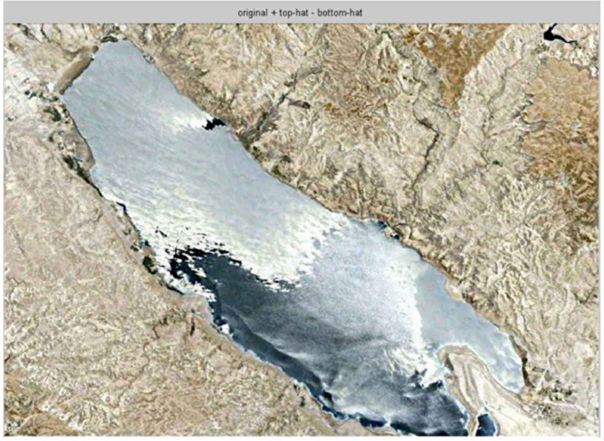


Fig. 5 Original, Top-Hat and Bottom-Hat Image.

The following step is converting the objects of interest. This can be done using the watershed transformation which that detects intensity valleys in the image. We also need to enhance the image by highlighting the intensity valleys using the imcomplement function in Matlab image processing toolbox as shown in Fig.6.

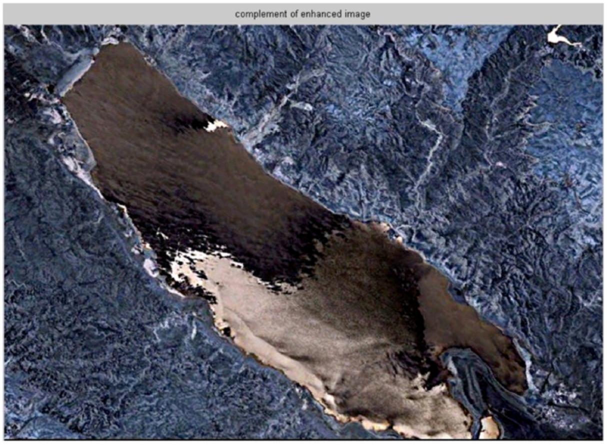


Fig. 6 Enhanced Image.

Then we need to detect all the intensity valleys deeper than a particular threshold with the imextendedmin functions in the Matlab image processing toolbox. The output of this function is a binary image. The location rather than the size of the regions in the imextendedmin image is important.

The imimposemin function modifies the image to contain only those valleys found by the imextendedmin function. The imimposemin function also changes a valleys pixel values to zero (deepest possible valley for unit 8 images), all regions containing an imposed minimal are detected by the watershed transform as shown in Fig.7 and Fig.8. Then we need to apply watershed segmentation. Accomplishing watershed segmentation of the imposed minimal image can be done as given in Eq.(5).

$$war = watershed(Iimpose) \tag{5}$$

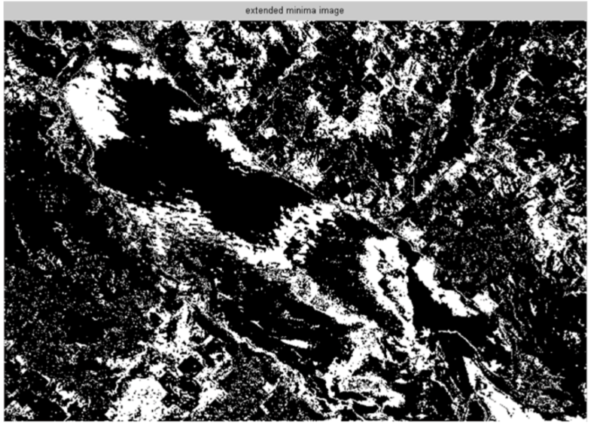


Fig. 7 Extended Minimal Image.

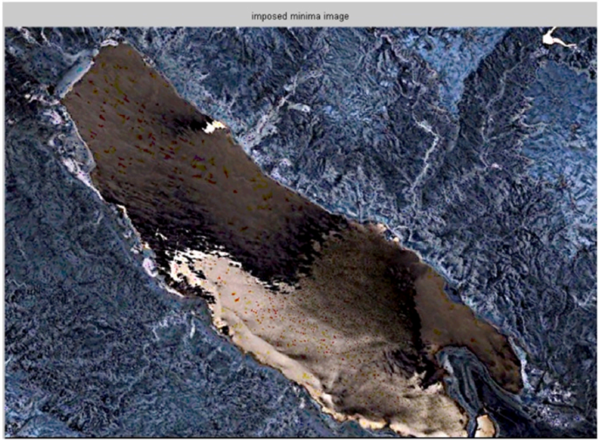


Fig. 8 Imposed Minimal Image.

The watershed function in Eq.(5) returns a label matrix containing nonnegative numbers that correspond to watershed regions. Pixels that do not fall into any watershed region are given a pixel value of 0, a good way to visualize a label matrix is to convert it to a color image, using the label2rgb function. In the color version of the image, each labeled region is displayed in a different color, and the pixels that separate the regions are displayed in white as shown in Fig.9.

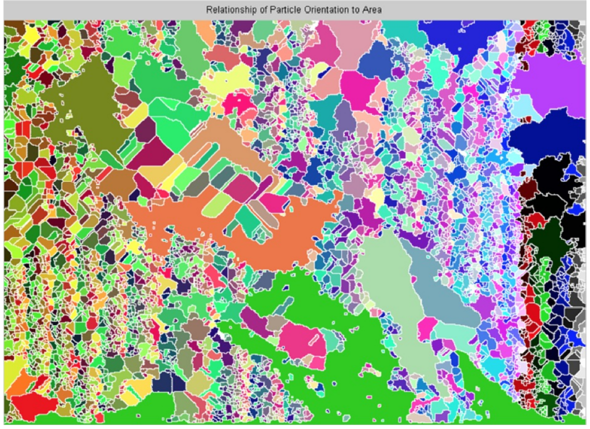


Fig. 9 Watershed Segmentation Image.

Table 1 shows the results generated by the evaluation tests. We can see the consuming time for every method. The result shows that threshold is the fastest, but the output image from threshold is not segmented clearly than other method, edge-base and watershed segmentation consuming long time than threshold but output image is improved, also threshold only segment gray-level images, it is considered to be the simplest and it did not segment image clearly. It only segments gray-level images and can be determined to segment objects and background. Edge segmentation is one of the earliest segmentation approaches and still remains very important.

Table 1: Results

method try	**thresholoding**	**edge-base**	**watershed**
1	**0.112**	**5.914**	**15.6512**
2	**0.15**	**5.215**	**16.6543**
3	**0.15**	**5.1489**	**15.8743**
4	**0.16**	**5.221**	**15.1234**
5	**0.32**	**5.1201**	**18.6543**
6	**0.16**	**5.943**	**17.6541**
7	**0.32**	**6.102**	**19.5741**
8	**0.312**	**5.0593**	**15.5421**
9	**0.341**	**5.4321**	**15.3421**
10	**0.16**	**5.4444**	**16.5432**
11	**0.13**	**5.4851**	**15.5641**
12	**0.17**	**5.0778**	**15.4321**
13	**0.14**	**5.4444**	**19.5664**
14	**0.13**	**5.49111**	**19.5432**
15	**0.13**	**5.0123**	**19.3421**
16	**0.3**	**5.1239**	**17.4343**
17	**0.32**	**5.0132**	**19.5432**
18	**0.13**	**5.0876**	**19.5431**
19	**0.112**	**5.0123**	**16.7653**
20	**0.113**	**5.1456**	**15.6564**
21	**0.13**	**5.5432**	**18.4512**
22	**0.14**	**5.132**	**19.6752**
23	**0.14**	**5.5317**	**19.5654**
24	**0.14**	**5.5356**	**18.7666**

5. Conclusions and Future Work

For the purpose of separating objects from their background in gray-level images, it is clear according to our results, that threshold is the best. But when it comes to effectiveness edge-based then watershed are more suitable respectively. As also working on colored images and focusing more on improved results excludes threshold from being used, as our results explain. Furthermore depending on our experiments we nominate edge-based algorithms for calculating the declining rate of the Dead Sea, without omitting the possibility of using watershed. Edge-based segmentations rely on edges found in an image by edge detecting operators these edges mark image locations of discontinuities in gray-level color.

Starting from this research results, future work may focus on improving the quality and efficiency of a reasonable analysis algorithm. Considering other types of remote sensing images then deciding which one is more reliable, or even so if a hybrid image may lead to better results. Another interesting work is to expand the analysis region to include the water resources feeding the Dead Sea, either by consulting other researches in this matter or proposing a new approach..

References

[1] A. McPhail, S. Lintner. (2013, August 26). Red Sea Dead Sea Water Conveyance Study Program Overview [Online]. Available: http://siteresources.worldbank.org.

[2] O. Muresan, et al., “Satellite Image Processing Applications in MedioGRID,“ in the 5th Int. Symposium on Parallel and Distributed Computing IS- PDC ’06, 2006, pp. 253-262.

[3]D. Gorgan, et al., “Grid based satellite image processing platform for Earth observation application development,” in IEEE Int. Workshop on Intelligent Data Acquisition and Advanced Computing Systems: IDAACS ‘09, 2009, pp. 247-252.

[4] Z. Wang and X. Pei, “Application of Principal Component Extraction technique in processing cloud images from Chinese FY-1 satellite,” in International Conference Image Analysis and Signal Processing: IASP 2009 ‘09, 2009, pp. 18-21.

[5] S. Bharathi, et al., “Performance analysis of segmentation techniques for land cover types using remote sensing images,” in 2012 Annual IEEE India Conference (INDICON), 2012, pp. 775-780.

[6] A. M. Montufo, “The Use of Satellite Imagery and Digital Image Processing in Landscape Archaeology. A Case Study from the Island of Mallorca, Spain,” in an Int. Journal Geoarchaeology, Vol. 12, No. 1, 7185, 1997.

[7] S. Beucher, et al., “Road segmentation by watershed algorithms,” in Proceedings of the Pro- art vision group Prometheus workshop, Sophia- Antipolice, France, 1990, pp. 212-246.

[8] R. Gonzalez and R. Woods, Digital Image Processing. Prentice Hall, 2007.

[9] M.Wenping, et al.,” Image segmentation based on differential immune clone clustering algorithm, “ in International Journal of Intelligent Computing and Cybernetics, vol. 6, Iss: 1pp. 83-102, 2013.

[10]W. Boulila, I.R. Farah, B. Solaiman, H. Ben Ghzala, ” Interesting spatiotemporal rules discovery: application to remotely sensed image databases, “ in VINE , vol. 41, Iss: 2, pp. 167 - 191, 2011.

Mua’ad M. Abu-Faraj received the B.Eng. degree in Computer Engineering from Mu’tah University, Mu’tah, Jordan, in 2004, the M.Sc. degree in Computer and Network Engineering from Sheffield Hallam University, Sheffield, UK, in 2005, and the M.Sc. and Ph.D. degrees in Computer Science and Engineering from the University of Connecticut, Storrs, Connecticut, USA, in 2012. He is, at present, assistant professor at the University of Jordan, Aqaba, Jordan. He is currently serving as reviewer for the IEEE Micro, IEEE Transactions on Computers, Journal of Supercomputing, and International Journal of Computers and Their Applications (IJCA). His research interests include computer architecture, reconfigurable hardware, image processing, cryptography, and wireless networking. Dr. Abu-Faraj is a member of the IEEE, ISCA (International Society of Computers and their Applications), and JEA (Jordan Engineers Association).

Nazeeh A. Ghatasheh received the B.Sc. degree in computer information systems from the University of Jordan, Amman, Jordan, in 2004, the M.sc. Degree in Electronic Business Management and the Ph.D. Degree in Electronic Business from the University of Salento, Lecce, Italy, in 2008 and 2011 respectively. He conducted research activities in the aerospace field related applications, information and communication technologies in the telecommunication industry, and corporate knowledge management. His research interests include image processing and its applications, knowledge representation and management, corporate learning, and e-Business. Dr. Ghatasheh, at present, is an assistant professor at the University of Jordan, Aqaba, Jordan, since 2011..