

## Performance of Automatic Seismic Image Analysis Tool

**K. Rajesh Babu**

KLEF Deemed to be University, Vaddeswaram, Guntur, 522302, India

### **Abstract.**

We have developed a program in C# that scans 2D seismic images in search of geological formations; the primary reason behind this endeavor was the significant potential for time and cost savings it offers in the process of finding oil. Even in this day and age, image screening is still carried out by hand, despite the fact that the quantity of geological data being collected is expanding at an unprecedented rate. Because the screening of seismic images by using computer-assisted image analysis is a relatively unknown area of research, we make use of well-established theories about image processing and analysis. In the context of picture pre-processing, this refers to operations such as thresholding at specific intensity levels, frequency filtering, mathematical morphology, and so on. Skeletonization, distance transformation, property measurement, and a host of other techniques are utilized during the analysis and classification processes. The results are highly encouraging taking into account the fact that this is just an early stage prototype. The developed software tool is successful at finding and correctly classifying items, with the exception of certain issues that emerge when various image objects are strongly coupled to one another. Processing time for a "standard sized" single image can range anywhere from 0.2 seconds to 10 seconds, depending on how much data the image has. It should be noted that this is for an AMD Athlon X2 running at 2 GHz, and the code is not optimized in any way.

**Keywords:** Seismic images; 2D; Morphology; and classification.

### **1. Introduction**

#### *Seismic data format*

The subsurface is mapped to digital 3D volumes by sound waves - if onshore by special vibrator trucks attached with an array of receiving geophones, if offshore by boats with an air gun generating vibrations. The transmitted waves are reflected back when hitting

density boundaries of, e.g., changes in rock types or fluid contents. Using this method the signal's time of travel is closely related to subsurface depth (Satterfield/Whiteley) [1].

Upon interpreting, the recorded volume is often sliced vertically, horizontally or along a "horizon". The latter represents a line or surface corresponding to a density border. Though, for this project we will solely work with horizontal slices which are more commonly known as "time slices". The data is colored red and blue where high amplitude reflections occur.

### ***Computer aided search***

Due to modern techniques and depleting oil supplies the oil industry encounters a very rapid growth in the amount of seismic data to analyze. But the increase in the number of employed doesn't by far reflect this increase in collected data [2].

As for now, the images are screened manually. In order to streamline the process it does seem natural to consider computer based image analysis.

### ***Image interpretation***

As mentioned in Section 2 red and blue parts of an image are strong reflections. In general, the more reflections an image shows, the more interesting it is. The reflections build up shapes that represent different geological objects, and we learned that our image set has three different kinds of objects (Figure 3):

- Channels,
- Deposits,
- Structural reflections.

As it becomes increasingly difficult to find profitable deposits of oil to explore, the accuracy of the information regarding promising geological sites becomes more and more important [3]. Huge amounts of seismic data are collected every week but the manpower and time needed to analyze all the data manually imposes a huge cost. This means that the need for a computerized automated analysis tool is in great demand.

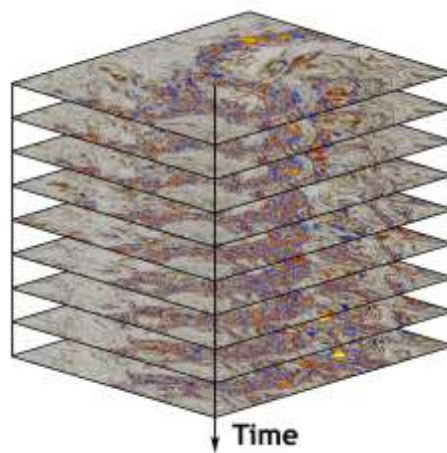
The aim of this work is to create a prototype program that can automatically find and classify interesting objects in seismic images. The prototype program should work as a first step towards a fully functional automated analysis tool.

Our prototype program is developed to work on a special type of images, namely colour coded amplitude spectrum images. It can easily be modified by changing a few parameters to

handle different images. Automatic parameter adjustment is very complex, however, and lies outside our project scope [4].

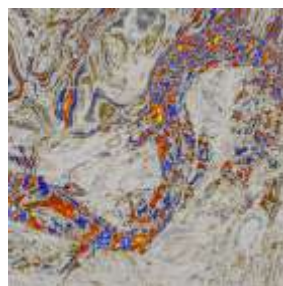
We were given nine time slice images as our development dataset. They are all from the same place, only separated by some milliseconds in depth.

In the amplitude spectrum images the reflections from softer sediments are clearly differentiable from the reflections of solid rock, making interesting areas easily detectable for human eyes. However, present in all these images are virtual artefacts that make the task of classification and differentiation much harder.

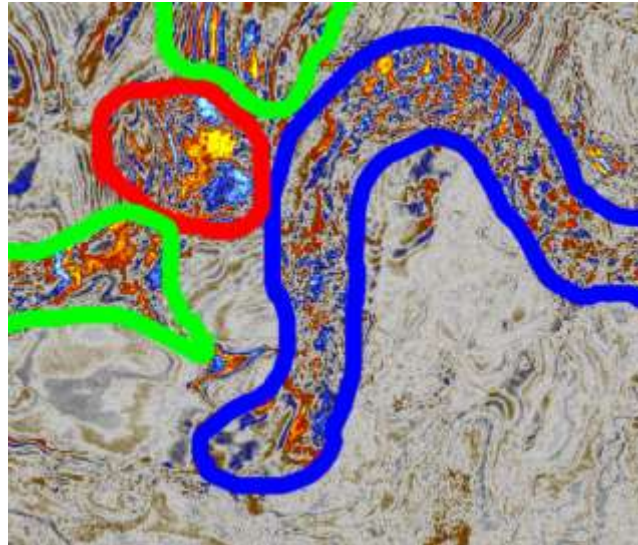


**Figure 1** 3D volume illustration with stacked time slices.

The time resolution of the original data set is 2 msec.



**Figure 2** Time slice where a channel can be seen.



**Figure 3** Outlined objects.

## 2. Methods

Provide sufficient detail methods to allow the work to be reproduced. Methods already published should be indicated by a reference: only relevant modifications should be described. This project is not based on or built on any previous work in this area, so we build our tool using well-known image analysis algorithms programmed in high level languages. In general the following is our method of development:

1. Use MATLAB and its Image Processing Toolbox to try different image analysis methods. The Image Processing Toolbox offers a comprehensive set of image analysis functions, and allows for a very quick startup phase in the development process.
2. When we feel confident that we have found a working method, the MATLAB code is migrated to C# code. For image processing and analysis we use EmguCV which is a .NET wrapper to the Intel OpenCV image processing library (for C/C++). While searching for suitable image processing libraries we found that OpenCV is, arguably, a de facto standard in image processing and computer vision implementations. EmguCV/OpenCV does actually not do everything MATLAB does, which forces us to implement many algorithms ourselves [5].

The choice of using C# (instead of e.g. C) is solely because Schlumberger's software platform Petrel nicely allows for .NET plugins or apps. While our tool is not a Petrel

application, it had been considered in an early stage. Furthermore, making it C# will facilitate any attempts to integrate it within Petrel in the future [6].

3. Final evaluation of results and computation times. Even though speed is not a part of our goals, the problem formulation has an inherent speed concern - the tool needs to outperform human screening.

Threshold is the most common way to select parts of an image. In a normal 8-bit grayscale image, each pixel has a brightness value that ranges between 0 and 255. When we use a threshold, we simply select those pixels that have a brightness value larger than a certain threshold [7]. Then we create a new binary image where the selected pixels are set to white and the rest set to black. An example of this technique with different threshold values is shown in Figure.

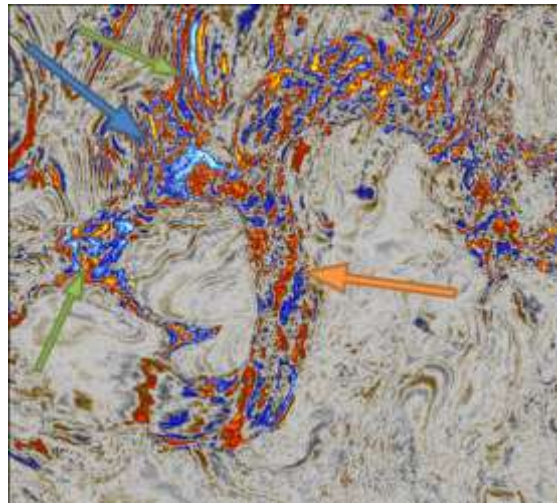
By using color space transformations this can be extended to a more general concept. By converting the image to a HLS (Hue-Lightness-Saturation) color space, the saturation and hue will also be represented by a grayscale image. So by applying threshold to for instance the saturation channel, we can select saturated areas [8].

### 3. Results and Discussion

In this section we consider the output of two different types of images. The first we “call main images” because the program development is almost exclusively based on this type. The other image type is discussed in “Other images” and we use it to demonstrate some flexibility of the tool. Last, a discussion about speed is included.

To evaluate results of the program we start with one of the nine images from Schlumberger. Because of the separation problem we use two versions of this image. The first version is the unmodified image In the other version we have by hand disconnected connected. We see that in the modified image, objects are classified correctly. There are a channel object and a deposit object.

The channel and the deposit are connected to each other in the original image, where the connection region is equally wide as the channel. This is a segmentation problem that requires a much more complex solution than we have implemented. Even though the deposit and the channel are segmented as one object, the factorization step finds the right channel path.



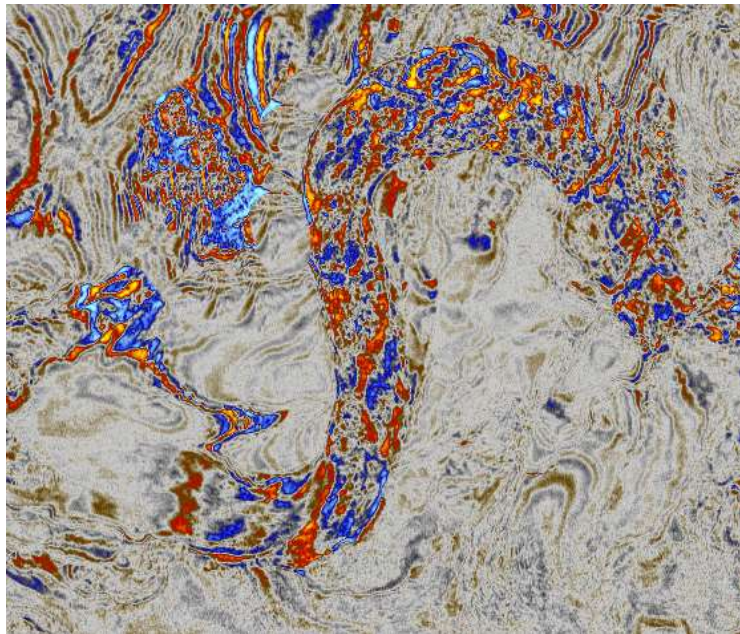
**Figure 4** Illustration of separation problem.

The channel (orange arrow) is strongly connected to the deposit (blue arrow) which is further connected to the structural reflections (green arrows).

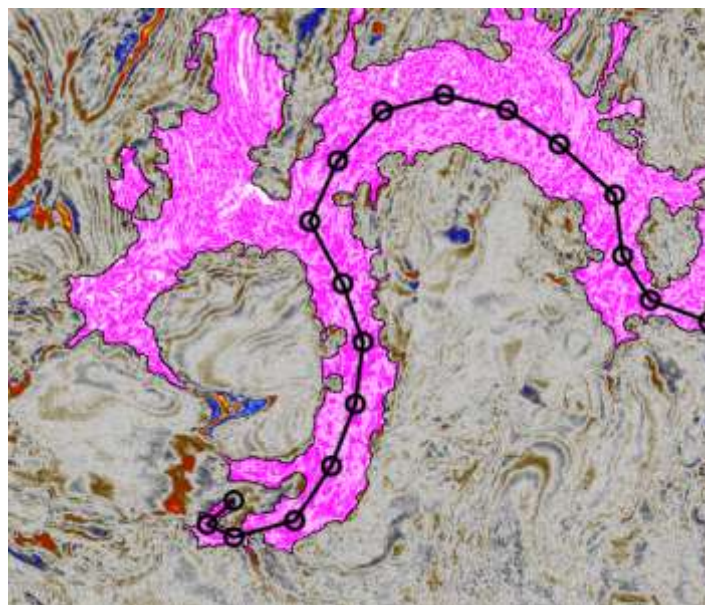
Here we try another image with a visible channel. The common feature between this image and the main image type is that objects of interest have distinctly different color intensity compared to background objects. On the other hand notable differences are color space, contrast and size. By changing the parameters of the algorithm, this image can also be nicely segmented and classified for objects.

In the long run, beyond our prototype program, speed is of utmost importance. Due to time constraints we haven't optimized our code with respect to speed, but we will here present some figures to show that doing automatic segmentation and classification can become a huge benefit in future.





**Figure 5** Modified image different object types have been separated



**Figure 5** Object 1 Class: channel, Elongated- , Size: 43 · 103, Length: 813. One image object. The vector rejects 2 Class: deposit, Elongatedness.

#### 4. Conclusions

We have met the goals we set at the beginning of this project. We have created a prototype program that can segment, detect and, to some degree, classify interesting objects in seismic images. The program is tuned to work on colour coded acoustic amplitude reflection images but can be adapted to different images by changing a few parameters.

Two major concerns are still present that we have not been able to solve in the scope of this project. These are separation of geological objects that should not be connected and an advanced classification tool. The program is by no means a complete analysis tool and there is a lot of room for further development. We have given a few suggestions to this end in this report.

#### Acknowledgements

We thankfully acknowledge management of KLEF (Deemed to be University) to provide every source and required facilities for completion of this work.

#### References

- [1] Dorothy Satterfield, Martin Whiteley. Seismic Interpretation and Subsurface Mapping. [http://www.derby.ac.uk/files/seismic\\_interpretation.ppt](http://www.derby.ac.uk/files/seismic_interpretation.ppt) (January 12, 2012)
- [2] Gonzales, Rafael C, Woods, Richard E. 2008. Digital image processing. 3<sup>rd</sup> Edition. New Jersey: Pearson Education Inc.
- [3] Jolliffe, I.T. 2002 Principal component analysis. 2nd Edition. New York: Springer-verlag New York Inc.
- [4] Sonka, Milan, Hlavac, Vaclav och Boyle, Roger. 2008. Image processing, Analysis, and Machine Vision. 3rd Edition. Stamford: CENGAGE Learning
- [5] Hosseini-Fard, Esmail, et al. "Automatic seismic image segmentation by introducing a novel strategy in histogram of oriented gradients." Journal of Petroleum Science and Engineering 209 (2018): 109971.



- [6] Di, Haibin, Zhen Wang, and Ghassan AlRegib. "Real-time seismic-image interpretation via deconvolutional neural network." SEG International Exposition and Annual Meeting. SEG, 2018.
- [7] Valet, Lionel, et al. "A fuzzy linguistic-based software tool for seismic image interpretation." IEEE Transactions on Instrumentation and Measurement 52.3 (2003): 675-680.
- [8] Woods, Joshua E., et al. "Automated crack detection and damage index calculation for RC structures using image analysis and fractal dimension." Journal of Structural Engineering 147.4 (2015): 04021019.