GENERALIZED XYZ DATA EDITOR SOFTWARE FOR MARINE GEOPHYSICAL SURVEY

Yet-Chung Chang*, Ching-Hui Tsai**, and Shu-Kun Hsu**

Key words: geophysical, GUI, editor, software.

ABSTRACT

For most marine surveying data processing tasks, such as gravity, magnetic or echo-sounding, to find and eliminate the errors among the raw data is a tedious and time-consuming process. Usually, the Graphical User Interface (GUI) software provided with the data acquisition system is very useful to reduce the manpower requirements and speed up processing. However, each tool typically only displays one specific data type or format. In this paper, a GUI program is designed for general marine geophysical data editing. It allows users to define their own data formats, and thereafter it can recognize the data type automatically according to the type of file extension.

I. INTRODUCTION

For most marine geophysical data, including gravity, magnetic, or echo-sounding data, the core information collected from field are the geographical coordinates (X, Y) and the measured intensity or potential value Z. They are generally further interpolated into a data grid and are used to create contour maps or colored images for interpretation. Without efficient data debugging, erroneous data will be blended with correct information, confounding results. However, editing and removing incorrect data is a difficult and time consuming task, even though it is very important. It is especially time consuming when the volume of raw data is large. Therefore, there is the need for a software tool with a graphical interface to allow us to visually display data as color dots on a map. A large data set could then be displayed as an image and their spatial continuity could be easily checked.

Most advanced geophysical systems do not provide GUI editor in their main package, or only accessory software severed for specific data format, in accordance with their own system. In this paper, we propose a simple and easy to use GUI editor for general debugging of marine geophysical data. Most of the functions of similar modern software were implemented in the software, and the execution of the software is efficient.

In addition, a format analyzer is also designed in the program to adapt some different formats. The benefits of this design can be stated as: Firstly, it is no promise for the data must be in the sequence of (X, Y, Z). The sequence in ("Y", "X", Z), or the "Z" located first, are also possible. The format analyzer can adapt to those difference by the user definition. Secondly, the separator of the data columns might be a blank, an invisible "Tab" character or a comma. The analyzer can also adapt to those difference by user definition.

II. FORMAT ANALYZER

This is an interface for the system that converts the contents of a text file into an (X, Y, Z) data structure. The first step is to define the file format that needs be processed. As shown in Fig. 1, a file can be selected for analysis by clicking on the "Select File to Analyze" button. Then the first line of the file will be read and shown in all three textboxes in the frame. The user can use their mouse to select the correct characters for "X", "Y" and "Z"; and after the three selections are made, the "Save Result" button can be used to save the format recognition information as a random access file "C:\XYZdatfmt.RND" in your computer.

Once you select the substring within the text record, the program will remember where to find each column of XYZ data in files with the same file type extension, such as "mag" as shown in Fig. 1. In some cases, the column is not defined by the absolute position of the character, but by specific separators, such as "," or simply a blank space. Here, we also provide separator definitions. If the "Separator" and either a "Comma" or "Space" are chosen, then the program will use the defined separator to split subsequent columns thereafter.

The program recognizes the file type by the file extension; each different extension name stands for a specific data format. Once a new format is established, the format bank file is expanded. Every time the program initiates, the format bank file is loaded. Once a file is selected to be processed, the program selects a format type to analyze the data file. If the format of the file type has been defined, the format analyzer is initiated to guide the user to establish a format definition. If a file with

Paper submitted 12/05/07; revised 02/20/08; accepted 03/19/08. Author for correspondence: Yet-Chung Chang (e-mail: ycc@tsu.edu.tw).
**Department of Digital Entertainment and Game Design, Taiwan Shoufu University, Tainan City, Taiwan, R.O.C.*
***Department of Earth Sciences, National Central University, Taiwan, R.O.C.*

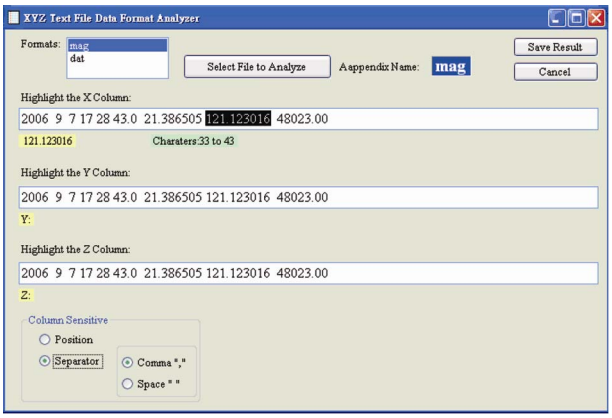


Fig. 1. Format Analyzer for data format definition.

the same extension but a different format is encountered, an error will occur during the data analysis. When this occurs, the program will stop reading and prompt the user to redefine the file format using the format analyzer.

III. DATA PREPARATION

When a format has been established, the data file will be loaded as a byte array, and not as lines of text, or a list of individually defined mathematical variables. This enhances the file system's operation, by speeding up the data loading. In addition, it reduces the number of background encodings, translations, and format checks that must be performed by the computer to read a string or mathematical variables from the disk to the computer's memory. Avoiding these processes is important, because they will frequently interrupt the hard disk's operation, and therefore severely slow down the process. On the other hand, reading a byte array requires less checking processes by the system, so, it is much faster. The trade-off is that we need to write more code to rearrange or decompose the byte array into the mathematical data arrays. However, after the data stream is actually in the memory (RAM), movement of data is many times faster.

After the byte arrays are ready, they are encoded into a single string at first, and then split into lines of text arrays according to the line breaking character(s), like "Carriage Return (ASCII=13)" or "Line Feed (ASCII=10)." Here, the line breaking character(s) are recognized automatically. Generally, there are only two possibilities. One for a Unix System is a "Line Feed (ASCII=10)" only. The other for a DOS System is a combination of "Carriage Return (ASCII=13)" and "Line Feed (ASCII=10)". When processing is complete, the qualified data is saved into exactly the same format as they are read, including the line breaks, except that any bad records (or lines) are skipped.

When the lines of textual data are ready, the format information we created in the format analyzer is employed to further establish the mathematical X, Y, and Z arrays that form the basis for any task to be processed. In addition, a Boolean

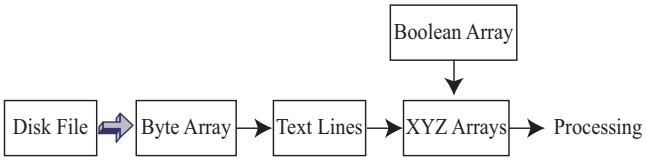


Fig. 2. Data preparing procedures.

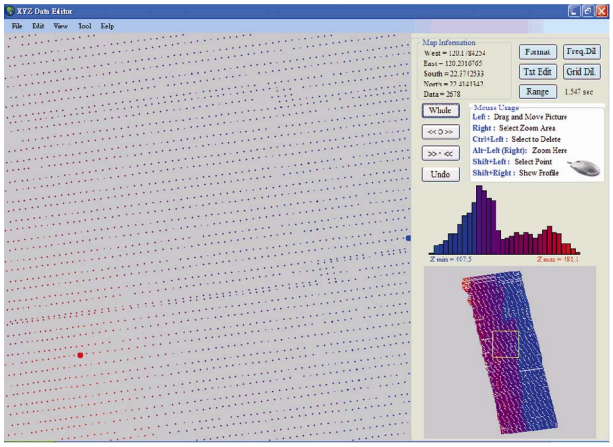


Fig. 3. A portion of multi-beam echo-sounding raw data.

array with the same dimensions of X, Y, and Z is also created to denote whether each datum is normal or abnormal. When the editing processes delete any datum, the corresponding Boolean value will be changed and will not be saved back to the hard disk at the end of processing. Figure 2 shows the main procedures of data reading and preparing. The 3-D gray arrow denotes a process of mechanical disk reading; the other plain black arrows are purely electronic operations within the memory, which are very fast in nature.

IV. MAP EDITING

At this stage, the XYZ mathematical data should be plotted as a map, and all the data points are shown as colored dots. The extreme values of data are searched and found first. The X and Y extremes are used to define the boundaries of the map; the Z extremes are used to define the color palette of the data points. Figure 3 shows an example of a multi-beam echo-sounding data set loaded and zoomed into its center portion. The map information, function buttons, usage of the mouse, Z distribution histogram and a simplified navigation map are shown on the right corner of the main data map.

The color palette here is from blue to red corresponding to the smallest to the largest Z-value continuously. Additionally, the maximum datum is plotted as a big red solid sphere, and the minimum datum as a blue one. Here, the contrast of colors is the key factor to evaluate the spatial data continuity, which is the main criterion for us to decide whether a datum is normal or not. Data with spatial continuity will draw little attention on the map; even though they are plotted as big dots. Otherwise,

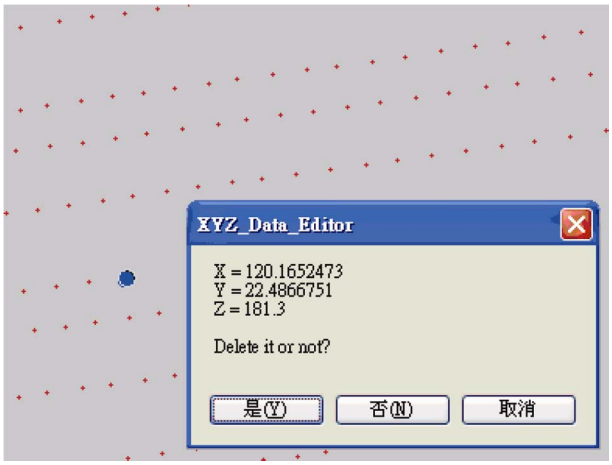
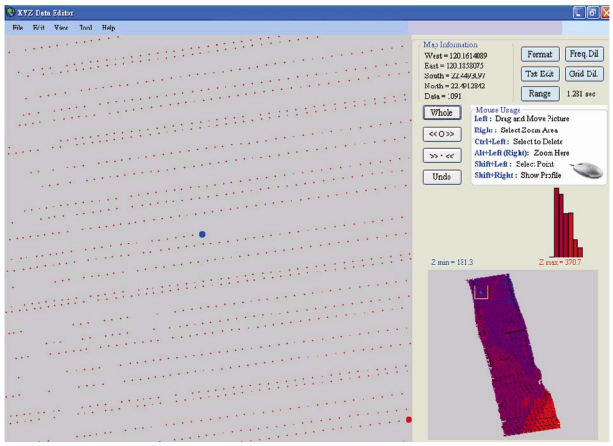


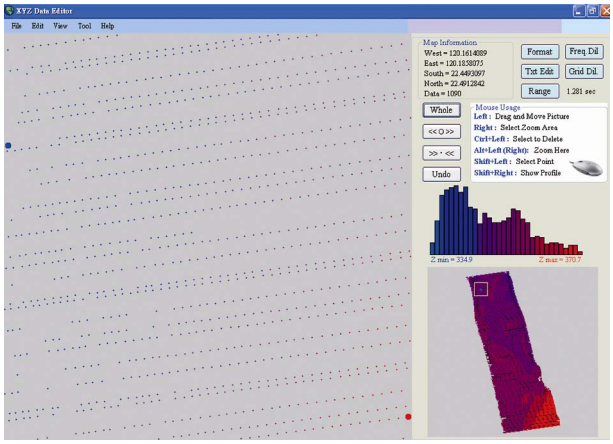
Fig. 4. Zoom in map and retrieve the single point data information.

they would be prominent and should be taken care of first. Figure 4 shows a zoomed area of a multi-beam data set. The detailed information of the selected point could be withdrawn and scrutinized by selecting it with the computer's mouse. If the point is abnormal, it can be deleted prudently.

Another useful information display is the histogram of value vs. data abundance distribution on the panel. Each time the map is re-plotted, the histogram will be re-built based on current existent data within the main window. If abnormal data exists, that is too big or too small; the histogram is shown as an unbalanced shape, like the one shown in Fig. 5(a). In this case, an incorrect datum with too small value exists and distorts the histogram by making most of the data to be relatively too large and displayed as red dots. After we delete the abnormal points, the histogram will become smoother and more reasonable, as shown in Fig. 5(b).



(a)



(b)

Fig. 5. (a) Maps and histograms with abnormal small value datum. (b) Maps and histograms after deleting abnormal small value datum.

V. TIME SERIES EDITING INTERFACE

The data can also be extracted and shown as a time series by selecting it with the computer's mouse. In many situations, a time profile can show not only the data anomaly, but also its cause. For example, the profile in Fig. 6 is a multi-beam data series; the abrupt peak in the center should be an error. On the other hand, the periodic changes elsewhere are the data coming form different swath, which should be taken as normal. The profile interface itself is an independent whole data viewer, which can be scrolled to display all of the data-set along the time axis. The user can also select and delete incorrect data from this interface and the result is synchronized with the main map immediately. The profile of the data is also displayed on the navigation map in the lower-right corner of the main window.

VI. DATA FILTERING ALGORITHMS

When erroneous data are numerous and swarm together, another deleting function could be used, to delete blocks of data. This is performed by simultaneously pressing the computer's "Ctrl" key and dragging the computer's mouse to form a rectangle, and then all points within that rectangle are deleted. This method is only appropriate if the data can be confidently established to be erroneous by its color alone, and offers the advantage of being easier and faster than getting the numerical information for the data.

In some instances, data filtering by certain thresholds is required. For example, the continental shelf has a limited depth about 130 m, based on generally accepted notions of oceanography. So we can use a simple threshold, say 300 m, to exclude any erroneous data obtained from the shelf area automatically without checking them on the map. This can be done by using the "Range Cutter" interface in this program as shown in Fig. 7. It also works for the X and Y range. So, the raw data outside the observed area can be excluded in a similar way as incorrectly measured values.

For many kinds of marine geophysical surveys, the equipment will produce more data than we actually need. For

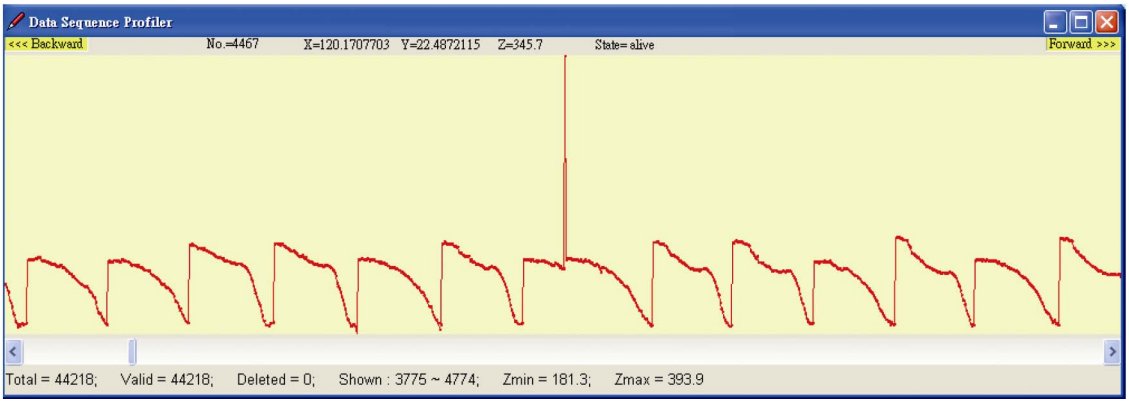


Fig. 6. A sample of multi-beam data in time series shows an abrupt change.

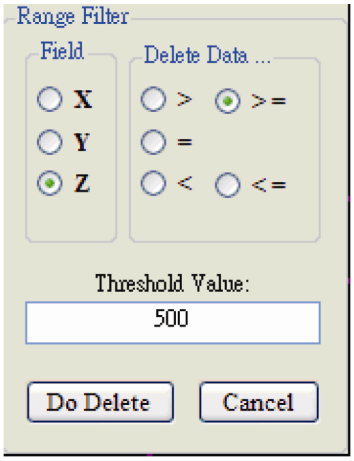


Fig. 7. Range Filter tool for checking out erroneous data according to the user defined X, Y or Z threshold.

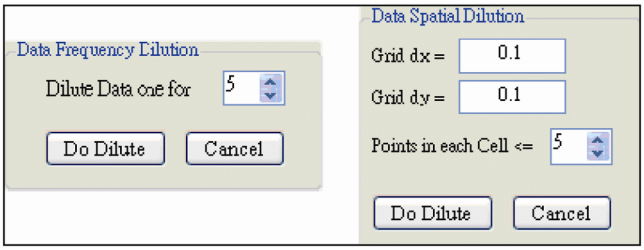


Fig. 8. Data Dilution operation interfaces.

example, gravity changes very little from mile to mile, while a ship is cruising, the equipment might produce thousands of similar data. Although the redundant data will not invalidate the results of the survey, minimizing them will reduce the data storage space and speed up the data processing. This program has two ways to eliminate redundant data. The first, is according to the data frequency as "one of every n sequential data will be saved, others skipped". The second is based on their spatial distribution, by minimizing the volume of data from each locality. Specifically, the program first accepts the user defined "dx" and "dy" intervals and the maximum number of data in each cell; then sets the data one-by-one into the nearest grid cell. If a cell is filled up to the maximum number, subsequent data will be ignored. The two user interfaces described above are shown in Fig. 8.

VII. TEXT EDITING OF THE DATA

During the data editing process, switching between the text and the image data viewer is usually demanded. In this program, several tools are designed for viewing and editing textual data. Under the item "View" in the function table, the file information and numerical X, Y, Z data are displayed, but they are set to read-only, and can't be edited. If editing is required, from the "Tool" menu user can open a true text editor for the current file that is being processed. In addition, from the "Help" menu, user can visit the website that offers support for this software and get more information and help.

VIII. DISCUSSION & CONCLUSION

The software can help scientists to easily remove erroneous data by using a graphical interface. In addition, the specially-designed Format Analyzer can make the software adapt to many different data formats. Once a new format is defined by the user, the program can thereafter automatically recognize the format according to its filename extension. To some extent, the program has the ability to "learn" and "remember" all pre-specified format. This is very useful for workers who need to process data from many different sources.

For most situations, the data processors in certain geophysical laboratory need only simple functions to check out the unwanted data, and they usually had software for regular data processing already, which is packaged with their surveying system. The simple and free software introduced here can easily accomplish the necessary data editing for most geophysical laboratories without introducing any redundant and expensive software serving only for data editing.

The software in this version is quite simple and straightforward. Although it was designed firstly for marine geophysical data, it has no limitation of applying it to any sur-

veying data on land or on air. A major limitation is its lack of ability to read binary format data. The author had successfully designed similar editor software for specified multi-beam binary data [5]. It is not a problem to design binary reading module, but it is difficult to design an interactive Format Analyzer for binary data.

Due to advances in computer technology, geophysical data processing has been increasingly relying on computer software. One of the dramatic changes was when window based software replaced non-window based software, around the middle of 1990's. Before that, most data processing programs were designed and maintained by scientists. The user interface and executing efficiency of those programs wasn't optimal, but via them the scientists had full control over their research. After that, commercial software with a fancy GUI and high execution efficiency became the core of most data processing tasks. The tasks became easier and faster, but the flexibility and control of the research decreased.

The proposed software, XYZ Data Editor, can help scientists to easily remove erroneous data by using a graphical interface. A major difference of this software compared to commercially available software, is that it provides a Format Analyzer, which allows user-defined formats, and removes the need for format translation in most cases. It has been applied to several kinds of geophysical data including: gravity, magnetic, and multi-beam echo-sounding data.

However, the main purpose of this work might not be a true challenge to the existing commercial software in their functions. In the other hand, its contributions might be:

1. Providing an alternative for most researchers who have not enough resources or not willing to pay the price of specific software for their simple demands.
2. Providing a transparent data processing tool by opening the source codes of this software.
3. Providing a chance for researchers who want to design or implement their own editing criteria or algorithms in the software.

REFERENCES

1. Deitel, H. M., *Visual Basic. NET For Experienced Programmers*, Prentice Hall, New Jersey (2003).
2. http://www.microsoft.com/taiwan/vstudio/express/
3. http://ycc.tsu.edu.tw/Research/XYZ/xyz_data_editor.htm
4. Skibo, C., Young, M., and Johnson, B., *Working with Microsoft Visual Studio 2005*, Microsoft Press, Washington (2006).