Advanced Petroleum Recovery Solution Inc.
Well-logging Digitization Software

GRADUATE PROJECT TECHNICAL REPORT

Submitted to the Faculty of
The Department of Computing and Mathematical Sciences
Texas A&M University-Corpus Christi
Corpus Christi, Texas

In Partial Fulfillment of the Requirements for the Degree of
Master of Science in Computer Science

by

WeiYu Chen
Summer 2004

Committee Members

Dr. Michelle Moore
Committee Chairperson

Dr. John Fernandez
Committee Member

Dr. Dulal Chandra Kar
Committee Member
ABSTRACT

The result of this project was the development of well logging digitization software for Advanced Petroleum Recovery Solution Inc. The software converts scanned well logs to vectors for input to well logging digital analysis software. It provides manual/auto tracking of well logs, rapid display of results, edit and image print capabilities and LAS standard format files. The user interface is understandable and usable.

The program is a 32 bit windows program that uses MFC (Microsoft Foundation Classes). Most of the Windows objects use MFC built-in windows classes, such as Work Place, Tool Bar, Menu, etc. Several image processing algorithms were applied. Tracing curves on paper log images and converting these curves to corresponding numeric values are the main functions of the software.
# TABLE OF CONTENTS

ABSTRACT .................................................................................................................. ii

TABLE OF CONTENTS ................................................................................................... iii

LIST OF FIGURES ........................................................................................................ v

LIST OF TABLES ........................................................................................................... vii

1. INTRODUCTION AND BACKGROUND .................................................................... 1
   1.1 Advanced Petroleum Recovery Solution Inc. ....................................................... 1
   1.2 The Purpose of the Project .................................................................................. 2
   1.3 Project Goals ..................................................................................................... 4

2. OIL WELL PAPER LOG DIGITIZING SYSTEM ......................................................... 5
   2.1 Digitizing Philosophy ....................................................................................... 5
   2.2 Features of the Software .................................................................................. 6
     2.2.1 Easy to Learn and Use ............................................................................... 6
     2.2.2 General Features ...................................................................................... 7
     2.2.3 "STEPS" on the Right Hand Side of Panels .............................................. 8

3. SYSTEM DESIGN .................................................................................................... 13
   3.1 Environment ..................................................................................................... 13
   3.2 Overview ......................................................................................................... 13
     3.2.1 Output Data Files ..................................................................................... 16
   3.3 Image Tracing .................................................................................................. 16
     3.3.1 Violent transform ...................................................................................... 18
     3.3.2 Hough Transform ..................................................................................... 19
3.3.3 Randomized Hough transform ................................................. 21
3.4 Summary .................................................................................. 22

4. EVALUATION AND RESULT ....................................................... 23
4.1 Meeting the Objectives .............................................................. 23
4.2 Simple Curve ........................................................................... 23
4.3 Different Type of Line .............................................................. 24
4.4 Cross Over ............................................................................... 26
4.5 Curve With Scales .................................................................... 27

5. FUTURE WORK ......................................................................... 30

6. CONCLUSION ............................................................................ 31

ACKNOWLEDGEMENTS ................................................................ 32

BIBLIOGRAPHY AND REFERENCES .............................................. 33
LIST OF FIGURES

Figure 2.1 Sample of User Interface: (a) menu (b) tool bar ........................................... 5
Figure 2.2 Sample of Paper Log .................................................................................. 6
Figure 2.3 (a) Tool Bar Tips (b) Status Bar Help (c) Different Cursor ......................... 7
Figure 2.4 Color Value on Tool Bar ............................................................................ 7
Figure 2.5 Sample of Instruction Dialog ..................................................................... 8
Figure 2.6 Interface 1 ................................................................................................. 8
Figure 2.7 Interface 2 ................................................................................................. 9
Figure 2.8 Scale Before(a) and After(b) Scale Detection Process ............................... 10
Figure 2.9 User Options ......................................................................................... 10
Figure 2.10 Interface 3 .............................................................................................. 11
Figure 2.11 Interface 4 .............................................................................................. 11
Figure 2.13 Interface 5 .............................................................................................. 12
Figure 3.2 Project Flow Chart .................................................................................. 14
Figure 3.3 Image Trace of a Car ................................................................................ 17
Figure 3.4 256 x 256 Highway ............................................................................... 17
Figure 3.5 Digital Line ............................................................................................. 18
Figure 3.6 Collection of Dots .................................................................................. 19
Figure 3.7 Example of Dots .................................................................................... 20
Figure 3.8 Hough Transform Result ......................................................................... 21
Figure 3.9 Random Selection ................................................................................... 21
Figure 3.10 Result of Randomized Hough Transform ................................................ 22
Figure 4.1 Effect of Velocity on Tracing. (a) correct (b) incorrect ............................... 24
LIST OF TABLES

Table 3-1 Functions Descriptions ................................................................. 15
Table 3-2 Image Tracing Methods ................................................................. 18
Table 3-3 Algorithm Comparison ................................................................. 22
1. INTRODUCTION AND BACKGROUND

Many new techniques are currently used for well logging. Processing of the information can be too large and complex a task to rely on only a human brain and human experience. That is why well designed software is so important to this field.

1.1 Advanced Petroleum Recovery Solution Inc.

Well-logging has developed rapidly since its introduction a half century ago. Even so, it is still a very new technique in the engineering field. Today, it has become the key process in many areas of the petroleum, coal and hydrogeology industries. Well log data is extremely important for analysis. Without it, expenses can become uncontrollable and huge amounts of resources can be wasted. Thanks to developments, in oil recovery technology, we have relatively inexpensive oil. However, these technologies require sufficient and accurate data. Most oil/gas wells were developed in the 80’s. Data for most of them is available only in paper log format. But digitalized data is crucial to modern well log analysis. In order to resolve this problem, appropriate software is needed.

Advanced Petroleum Recovery Solution Inc. (APRS) was set up in Victoria County, Texas in 2001. Its main services are well logging analysis software development and consulting. In order to serve more customers, APRS has decided to develop a huge petrophysical software package. It will include new well logging analysis application software (Digitizing software), 3D reservoir simulation software, and digitizing software. Digitizing software is what this project is all about. It converts paper logs to digital logs. Digital logs can then be used in the well logging analysis software.
1.2 The Purpose of the Project

APRS reexamines many old oil fields hoping to find huge bypassed reserves. The main reason the market exist is that there are a huge number of old oil fields that were developed before 1990, when well logging technology was not good enough to discover the bypassed reserves [Shell 1995]. Paper logging was used to record and analyze the oil wells. However, the accuracy rate was very low. With the great development of computer technology in general, many advanced computer technologies were introduced to the well log analysis field. This greatly improves the well log analysis accuracy rate. Current well log technologies mainly use digital data, not paper logs. Converting paper logs to digital data requires a lot of time, money, and human resources. Most paper logs only contain basic information, which is not enough for current well log analysis. Thus, most companies are not willing to re-examine old oil fields. Consequently thousands of tons of oil will not be found [Lai 1995].

This project greatly reduces digitizing time and, when combined with APRS, the new well log method makes re-examination of old oil fields possible. Digitizing software is a necessary tool for well log analysis software. This project creates the digital logs that well log analysis software needs.

Currently most companies scan the paper logs. They manually trace those lines and turn them into digitalized data. Even a single paper log includes at least 7 to 8 curves. The length of the paper log can be up to 150 to 200 inches. It can require as much as two hundred work hours to finish just one paper log. The accuracy is poor, and many human errors can occur.
Due to the huge resource cost, automatic methods must be developed for this task. This project allows paper logs to be traced by computer. Digital image enhancement and a line organization algorithm are used. Because of the aging problem, the quality of most paper logs is not good. Image noise and even damage exist. To gain sufficient accuracy, image enhancement was necessary. The first step was to remove image noise and damage. The other significant problem was caused by the size of the paper logs. The length of paper logs is usually over 150 inches. Thus scanning them correctly becomes very difficult. Unbalanced and unwanted stretching can occur easily during scanning. The image enhancement function took care of these problems. The next step was extracting them from background.

Image recognition techniques are used in most text processing fields. OCR, Optical Character Recognition, is used to convert the article image to an editable text file. This project shares some techniques with OCR, but still has many differences. There are several lines in a paper log and some of them will cross over many times. The size of each line is huge. These complications increase the difficulty of the reorganization. The software has to separate the lines to get the correct result.

The software tries its best to avoid making mistakes, but human help is still necessary. After all, the paper image is quite complex. The software offers an easy to understand environment for the user to make adjustments. The user can click on the line that is wrong and drag it to correct place.

Finally, after the software collects all the digitizing data, it writes it into the standard industrial text format. The reason we need to use this format is so that the data can be used by different well log analysis software.
1.3 Project Goals

This project achieved the following goals:

- Automatic digitalization of paper logs.
- A friendly user interface.
- Suitability for both large and small oil companies.
2. OIL WELL PAPER LOG DIGITIZING SYSTEM

The project resulted in commercial software for personnel in the well logging field. The friendly User Interface (UI) was essential. For this purpose, Microsoft Foundation Classes (MFC) played an important role during the development.

2.1 Digitizing Philosophy

Digitizing software was designed to have the look and operational methods of a standard Windows program. Elements, such as pull down menus, common Windows terminology, linking of data between different graphics, etc, were used (Figure 2.1). A menu shows all the functions that the system provided to the user. The tool bar is a shortcut for the user to use those functions.

![menu](image1)
![tool bar](image2)

*Figure 2.1 Sample of User Interface: (a) menu (b) tool bar*

It is important that digitizing software be sufficiently well organized and user friendly. Figure 2.2 is a sample of a paper log.
2.2 Features of the Software

Specific features were incorporated into the project. The following sections describe some of the most important ones.

2.2.1 Easy to Learn and Use

The digitizing software was designed in a “step-by-step” fashion, which makes it easy to use and understand. The program also provides many user-friendly interfaces, such as dynamic user help, shown in Figure 2.3(a), (b) and different mouse cursors for distinct operations in Figure 2.3(c)
Furthermore, because some human effort is still needed for tracing, the program shows color values on the status bar (Figure 2.4). By showing this value, the user can make correct choices more accurately.

![Color Value on Tool Bar](image)

**Figure 2.4 Color Value on Tool Bar**

### 2.2.2 General Features

Program provides easy to use interface:

1. Point and click and step-by-step user interface with context sensitive help (Figure 2.5).

2. Base map, data traces, and histogram show the scanned images and the graph after software processing.

3. Data statistics table and data quality control table show digitalized numbers and results.

4. Data filter show result data after algorithm processing.
2.2.3 "STEPS" on the Right Hand Side of Panels

Five "steps" are on each panel on the right hand side:

- Project: provides an overview of the current project. Some display options are provided for the user to choose. From here, the user can get the whole picture of the project, or start a new project step by step as shown in Figure 2.6.
Import: Paper log in can be scanned and loaded in several popular image formats, including TIFF, JPEG, BMP, PNG, GIF and more. Various color depths and image resolutions can be accepted here (Figure 2.7). For accuracy, 24 bit, 200 pixels/inch and TIFF are recommended.

![NetMeeting - Not in a Call](image)

Figure 2.7 Interface 2

Digitize: Used to make tracing more effective. The scale of the image should be recognized and put in the record so that it can be referenced for tracing. By doing this, the interference that caused by scales can be reduced to minimum. (Figure 2.8 and Figure 2.9)
Figure 2.8 Scale Before(a) and After(b) Scale Detection Process

Figure 2.9 User Options

After recognizing all the scales, the program starts tracing the lines automatically.
The result is shown using different colors on the screen. Manual adjustment tools are provided to user. (Figure 2.10 and Figure 2.11)
Figure 2.10 Interface 3

Figure 2.11 Interface 4

- Output: Creates data output in text format. Two different output formats are provided. Users can check the results immediately (Figure 2.12).
Figure 2.12 Interface 5

The list of steps serves two purposes. First, the list is a guideline for the way in which a digitizing study is typically done. The arrow indicates where you are on the list. Clicking on any of the elements of the list will take you to that digitizing software panel. Thus, the list allows you to move anywhere in the program that you want to go.

If desired, the steps display can be toggled on and off by going to the View pull down menu and clicking on Show Steps. When the steps box is turned off, the steps will be listed as a pull down menu.
3. SYSTEM DESIGN

Because it is a Windows program, it is naturally an event driven model. Most actions and processes are invoked by events such as mouse click, key stroke or device interrupt. Under this concept, different categories of events will be defined beforehand, and a response will be generated.

3.1 Environment

The Oil Well Paper Log Digitizing System was designed to run on Microsoft Windows™ family Operating System, Windows 95/98/NT/2000/ME/XP/2003. The PC should be 586 or higher, with at least 100 MB hard disk free, and at least 64 MB of RAM. This project was built in Visual C++ 6.0 and with Microsoft Foundation Classes. The Oil Well Paper Log Digitizing System is used by the APRS.

3.2 Overview

Microsoft Visual C++ and Microsoft Foundation Classes (MFC) were the core tools for developing the digitizing software. By using the Microsoft Visual C++ development environment, large scale project development became manageable. Microsoft Foundation Classes allowed the project to have standard Windows™ graphic user interface (GUI) and operations. All the windows elements were derived from the MFC build-in object. Figure 3.1 shows the basic class categories of MFC [Microsoft 2003].
The functions in digitizing software are event-driven, so the graphical user interface is quite important to the system and the user. Figure 3.2 illustrates the key components of the digitizing software.

Figure 3.2 Project Flow Chart
The functional descriptions of major components are shown in Table 3-1.

**Table 3-1 Functions Descriptions**

<table>
<thead>
<tr>
<th>Function Name</th>
<th>Describe</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>Start the program.</td>
</tr>
<tr>
<td>Load</td>
<td>Load a scanned paper log in BMP format.</td>
</tr>
<tr>
<td>Convert</td>
<td>Convert image to specific format.</td>
</tr>
<tr>
<td>Pre-process</td>
<td>Erase unwanted dots, enhance the sharpness, recognize and separate the frame lines and curves.</td>
</tr>
<tr>
<td>Trace</td>
<td>Trace the curves and turn them into numbers.</td>
</tr>
</tbody>
</table>

The software includes four sub functions: Data List, Data View, Statistic Chart and Data Change.

- **Data List**: This item displays partial/all data in a table. Users can select how much data, or what part of the data they need to display on the screen.
- **Data View**: This item displays partial/all data in a plot. Each data item in a plot has a particular meaning for analysis.
- **Statistic Chart**: This item provides two or more types of data intersection charts, such as Neutron/Density, Sonic Transit Time/Density and Neutron/Sonic Transit Time.
- Data Change: This item edits and saves data in table/plot/intersection chart. After the program finishes the processing or the user adjusts the data, the user can save the changes.

There is only one function that is involved with output:

Output: Data will be output in text format or in industry standard format. Two different formats make it flexible for the user. Industry standard formats can be used by other software. Text format makes it easy to convert to any other format or machine.

### 3.2.1 Output Data Files

The worldwide acceptance of the Log ASCII Standard (LAS) proved the need for such a format. The LAS format is intended to provide an easy method to read and distribute well log data. The format is based on standard ASCII code so that it may be easily worked with using common simple computer tools such as a text editor or a word processor. LAS data files are commonly distributed on floppy disks although it is becoming increasingly common to see log data distributed in the LAS format due to its ease of use and openness. The format was originally developed by the Canadian Well logging Society in the late 1980's [Heslop 2000].

### 3.3 Image Tracing

After the system scans paper logs and turns them into digital images, they are enhanced for recognition. Because the goal was to extract digital data from the image, recognizing lines and curves became important. Figure 3.3 shows the idea of image
tracing [Chung 2002]. A complex image processing algorithm was not necessary for the project. The project dealt only with lines on the paper log. Still, several techniques were considered and applied.

![Figure 3.3 Image Trace of a Car](image1)

Figure 3.3 Image Trace of a Car

Figure 3.4 is a 256 * 256 image of a highway [Yung 1995]. Humans can easily recognize it. For a computer, some algorithm has to be applied.

![Figure 3.4 256 x 256 Highway](image2)
Table 3-2 shows the methods for image tracing.

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Violent transform</td>
<td>Take every dot and calculate.</td>
</tr>
<tr>
<td>Hough Transform</td>
<td>Use space parameter.</td>
</tr>
<tr>
<td>Randomized Hough transform</td>
<td>Vote for the most likely dot.</td>
</tr>
</tbody>
</table>

### 3.3.1 Violent transform

A digital line contains many black dots to form the shape. A width of line can mean several dots. Which dot did we want? Generally, the dot in the middle should be the one [Knuth 1985]. To explain it, a sample is provided below (Figure 3.5 [Yung 1995]). The edge A and edge B form a line. To get the best result of digitizing, we take the values which fall right between them, which B presents. Therefore, one has to find the edge of the line.

![Digital Line Diagram](image_url)
Every digital image is formed by a matrix of "dots". Dots with dark color attributes become part of a probable line. Using the violent transform, one can group those dots. By using this group, one collects adjacent dots and divide the total length of them by 2. Thus, a middle dot can be obtained [Yung 1995]. If we keep using this method for each group of dots, a line results. Figure 3.6 shows the result that can be obtained using violent transform [Yung 1995].

![Figure 3.6 Collection of Dots](image)

### 3.3.2 Hough Transform

The Hough Transform was introduced by Hough. Duda and Hart used this concept to develop a serial method to detect circles and lines [Duda 2000]. Basically, the Hough Transform maps \( x - y \) space to \( r - \theta \) Parameter Space, or Normal Distance – Normal angle space [Chung 1999]. The program goes through every dot on the image. Different angle values are used to form many virtual straight lines. The program then checks if there are enough dots in the lines. The count is called a "vote". When a line gets enough dots, or votes, it is considered a straight line. The program applies this procedure
to all dots and then tries all angles, until all straight lines are found. The downside is that this process is very time consuming.

In Figure 3.7, it is easy for a human to see that the line should be at a 45 degree angle, from up left to down right [Yung 1995].

![Figure 3.7 Example of Dots](image)

But for a computer, it can be difficult. In this algorithm, we use the formula:

\[ r = x_2 \cos \theta + y_2 \sin \theta \]

Let \( \theta = 45^\circ \)

(B, 3), (C, 2), (D, 1) are collinear. All of them are \( r = 3 \frac{\sqrt{2}}{2} \). For (D, 3), \( r = 3\sqrt{2} \). Thus it is clear that (D, 3) is not what is wanted. Why do we know \( \theta = 45^\circ \) is appropriate? We divide 0 to \( \pi \) into \( n \) parts. We test every one with every dot. Then we can get an acceptable degree. Figure 3.8 shows the result. It is much more accurate than the result achieved with the previous method [Yung 1995].

20
3.3.3 Randomized Hough transform

Fischler and Bolles introduced the RANSAC concept to deal with line tracing [Fischler 1981]. Later, Xu, Oja and Kultanan introduced a more efficient algorithm based on it [Xu 1995].

The main idea of this algorithm is to randomly select three dots on the line (Figure 3.9[Chung 2002]). Using these three dots, one can form a possible correct line position and repeat the actions again and again.

By using this algorithm, the unwanted noise can be removed. Figure 3.10 shows this advantage [Chung 2002]. Even though there is an unwanted bowl on the floor, it can easily be removed. This is especially useful for paper logs in bad condition [BEG 1995].
3.4 Summary

These algorithms are the basis for image tracing. More techniques can be developed based on them. There are obvious advantages and disadvantages to them (Table 3-3).

<table>
<thead>
<tr>
<th>Name</th>
<th>Advantage</th>
<th>Disadvantage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Violent transform</td>
<td>Easy to implement</td>
<td>Poor accuracy</td>
</tr>
<tr>
<td>Hough Transform</td>
<td>Good efficiency</td>
<td>Needs good input</td>
</tr>
<tr>
<td>Randomized Hough</td>
<td>Fit for all input in all condition</td>
<td>Complex</td>
</tr>
</tbody>
</table>

Because the project dealt with many different shaped lines and curves, the Hough Transform and the Randomized Hough Transform were both needed. The randomized Hough transform was good in many ways, but if the curve has a spike, the spike can accidentally be removed.
4. EVALUATION AND RESULT

Because this project was about curve tracing, our evaluation was rather straightforward. The white box technique was used here. We collected some of the most popular curve properties that occur in a general paper log and tested them. Several example images are used here. Each image presents one of the properties of the curve.

4.1 Meeting the Objectives

This program met all the objectives that were proposed.

- Provides friendly user interface.
- Traces the curve in a short time.
- Gives accurate results.
- Extracts scales before tracing.
- Saves digitizing results as LAS file.

4.2 Simple Curve

We initially tested the software with a simple single curve. One thing worth mentioning was that when the velocity of the curve reached some limit, tracing often failed to have the correct results (Figure 4.1). One of the reasons is that the dot underneath may get closer than the real next dot. To solve this problem, one can calculate the direction of the line and give the dots on the line direction side more chance to be found. The result was good.
4.3 Different Type of Line

Because most oil well logs are black and white, they use different types of lines to indicate different earth attributes. As shown in Figure 4.2, dotted lines made the process of tracing more difficult.

Figure 4.1 Effect of Velocity on Tracing. (a) correct (b) incorrect

Figure 4.2 Example of Dotted Line
Because those empty spaces had to be ignored, using a “for loop” to try next several possible dots became necessary (Figure 4.3).

![TESTPNG02](image)

**Figure 4.3 Tracing Result of Dotted Line**

Again, the type of dotted lines can be various. To test if our algorithm was robust enough, we used a more broken line to try (Figure 4.4).

![TESTPNG03](image)

**Figure 4.4 Example of Other Non-Solid Line**

We checked the result and found it still worked quite well. Then we knew our algorithm worked for different type of line (Figure 4.5).
4.4 Cross Over

Next, the most serious problem on a paper log was that sometimes two curves cross over each other (Figure 4.6). It caused some troubles.
Here we tested the more difficult one, the dotted line. The result was good as well (Figure 4.7).

![TESTPNG04](image)

**Figure 4.7 Tracing Result of Two Cross Over Curves**

### 4.5 Curve With Scales

In a real oil well paper log, there are scales along with the curves. As mentioned above, scales can introduce a lot of interference during the tracing process. The following test put the curve and scales together (Figure 4.8).
The program recognized the scale first, and showed them with different colors. The user can choose to hide the recognized scales or not (Figure 4.9).
The test showed that the program could handle a real paper log without problems (Figure 4.10).

Figure 4.10 Tracing Result of Curve with Scales
5. FUTURE WORK

Although the Oil Well Paper Log Digitizing System has achieved most of its goals, there is still some space to improve.

Fix the stretched image: Because paper logs are always pretty long, scanning them correctly becomes difficult work. In many situations, the log would be stretched in some part and become incorrect input for digitizing. The program could provide a tool to fix it.

Make it more efficient: Image processing always takes a lot of resources, and it becomes quite slow when the image gets very large. Program efficiency becomes important here. More sophisticated algorithms and parallel techniques could be used.

Parameter adjustment: Paper log tracing is a very complex task to do. Human effort is still needed; for the user's convenience, an interface for various parameter adjustments should be provided. It would give the user more control when confronted with different kinds of paper logs and provide more accurate results without increasing effort.
6. CONCLUSION

The result of this project was a powerful, fast, and easy-to-use program that
provides the petroleum industry a much better tool to digitize old oil well paper logs. This
project achieved its main goals, which were to simplify the tracing process; significantly
shorten the time of paper log digitizing; and decrease human-relative errors. Furthermore,
this program provides a friendly user interface, letting users control the tracing
processing. Lastly, it provides the user a function to save the digitizing result in standard
industry format for analysis program to do the further work.
ACKNOWLEDGEMENTS

I am grateful for the enlightenment given by Dr. Moore who encouraged me to conduct experiments and gave me lots of help on this project. I would like to take this opportunity to express my gratitude and deep regard for her guidance and mentoring. Sincere appreciation is extended to Dr. Fernandez for his help on editing my project and his personal concern for my career. I am also grateful to Dr. Kar for being one of my committee members.
BIBLIOGRAPHY AND REFERENCES


[Chung 2002] Chung, Ku-Lung. Progressive Image Transmission and Computer Vision, Ton Wha, 2002. (Figure 3.3, Figure 3.9, Figure 3.10)


[Yung 1995] Yung, Wu-Zi. Digital Image Process, Chum Wha, 1995. (Figure 3.4, Figure 3.5, Figure 3.6, Figure 3.7, Figure 3.8)