DEPARTMENT OF THE ARMY
COLD REGIONS RESEARCH AND ENGINEERING LABORATORY, CORPS OF ENGINEERS
HANOVER, NEW HAMPSHIRE 03755-1290

CECRL-PP

10 February 1999

MEMORANDUM FOR SERDP Office, ATTN: Ms Catherine Vogel, Suite 303,
901 North Stuart Street, Arlington, VA 22203

SUBJECT: Project CU/1049/6

1. Enclosed is the end of the year report for SERDP Project CU/1049/6: Application of Neural Networks Coupled with Genetic Algorithms to Optimize Soil Cleanup Operations in Cold Climates.

2. This effort will be presented to the Joint Environmental Engineering and Construction Innovative Technology meeting in St. Louis, MO in March of this year. Efforts are on-going that will, hopefully, lead to a full-scale successful demonstration.

FOR THE DIRECTOR:

for
WILLIAM R. LOVEN
LTC, OD
Chief, Plans and Programs Division

Enclosure

A Coupled Processing System for Ground-Penetrating Radar Delineation: Neural Networks and Consecutive Layer Identification Algorithms

An End of Year report for Project CU/1049/6:

Application of Neural Networks Coupled with Genetic Algorithms to Optimize Soil Cleanup Operations in Cold Climates

Submitted to:

SERDP Program for Cleanup
901 North Stuart Street, Suite 303
Arlington, VA 22203

by

Dr. John M. Sullivan, Jr.

US Army Cold Regions Research and Engineering Laboratory
Hanover, NH 03755

December 1998

EXECUTIVE SUMMARY

This 98 end of year report focuses on integrating the separately developed GPR-related software modules into a larger, common software framework in order to provide a single, easily accessible, and user-friendly GPR data processing and analysis environment.

In particular, the Consecutive Layer Identification (CLI) approach as well as the Adaptive Transform (AT) have been re-coded and incorporated under a Windows 95/98-based Graphical User Interface (GUI). This report and the attached program description serve as a comprehensive account of our research towards streamlining all the software development under one common user interface.

Besides the CLI and AT modules, the report also discusses on-going efforts of integrating the neural network modules currently under development. Our ultimate goal is the field evaluation of GPR time-amplitude data either via AT/CLI or the neural network module under a single software platform.

ACKNOWLEDGMENTS

This work was funded in part by SERDP (Strategic Environmental Research and Development Program) Project 1049/6 cleanup thrust area and by CRREL (U.S. Army Cold Regions Research and Engineering Laboratory) SFRC Number DACA89-97-K0001.

This December 1998 document integrates all work performed since our 12/97 report on this project (CU/1049/6). This work will be presented as a Master Thesis (with credit and acknowledgements to SERDP) by Q. Lai under the Academic and Thesis direction of the project PI, Dr. John M. Sullivan, Jr.

Table of Contents

Table of Figures

1. Overview

1.1 Introduction

Ground Penetrating Radar (GPR) is an electromagnetic remote sensing technique which uses radio waves, typically in the 10 to 2500 MHz frequency range, to locate and map different features and structures below the ground surface (bgs). In general, a GPR system transmits a short electromagnetic pulse into the ground - the pulse is reflected, refracted or scattered by the targets that exhibit some difference in electrical properties (dielectric permittivity, conductivity, and magnetic permeability) and is then recorded by the receiving antennas. The greater is the difference in the dielectric permeability, the larger is the amplitude of the reflection pulse.

High radar frequencies are needed to achieve a good spatial resolution, but penetration depth of the electric field is inversely proportional to the frequency. Hence the choice of frequency range is a trade-off between resolution and penetration depth. Penetration depth also depends on the nature of the soil, which has different attenuation properties. For example, desert sand has an attenuation of about 1 dB/m for a 1 GHz frequency, clay has an attenuation of 100 dB/m at the same frequency.

The reflected wave is sampled and digitized by an A/D converter to form a vector. Typically 512 or 1024 points are taken through the region of interest. The recorded signal in the time domain is called an A-scan and in many GPR applications the A-scans are recorded consecutively along some spatial direction usually called radar or transect line.

Typical GPR system records 5 to 10 scans per meter and these GPR soundings are performed by dragging a GPR hardware package including transmitting and receiving antennas behind a vehicle.

The goal of the GPR data processing system (simply called system below) is to process and understand the GPR data in an automatic way such that a minimum amount of human interference is required.

The inputs to the system are field-collected experimental data (Radar or Transect Lines). They are the response of the underground structure to high frequency electromagnetic pulses. Those responses are sampled and quantified as discrete data points or Ascans. Each radar line contains several thousands of Ascans. And each Ascan may contain 512 or 1024 sample points with 8-bit or 16-bit resolution. The task of the software system is to evaluate those data and provide comprehensive information about the underground structure together with the parameters of interest.

1.2 Data processing methods

In our previous research two major methods had been studied. One of those was the Neural Network approach, the other was the Consecutive Layer Identification (CLI) approach.

The NN approach passes the data through a pre-processing stage to extract the key features, then uses two different neural network algorithms to recover the subsurface configuration and parameters, respectively. The result is a combination of two pieces of information, one is an indication showing the stratified configuration according to a set of pre-defined patterns, and the other one is the corresponding layer depth profiles. These results and system development were reported in detail in our 1997 end of year report.

By utilizing an analytical model that provides an explicit description of the interaction between the injected pulse and the material, the second approach (CLI) tries to identify the underlying system from the impulse response. Again, a pre-processing stage has to be carried out to determine the impulse response.

It should be noted that the same pre-processing technique is used in both approaches, which is named the Adaptive Transform. It deconvolves the input signal with the output signal to determine the impulse response and then passes this response to either the Neural Networks or the CLI program. It plays a key role in the overall performance.

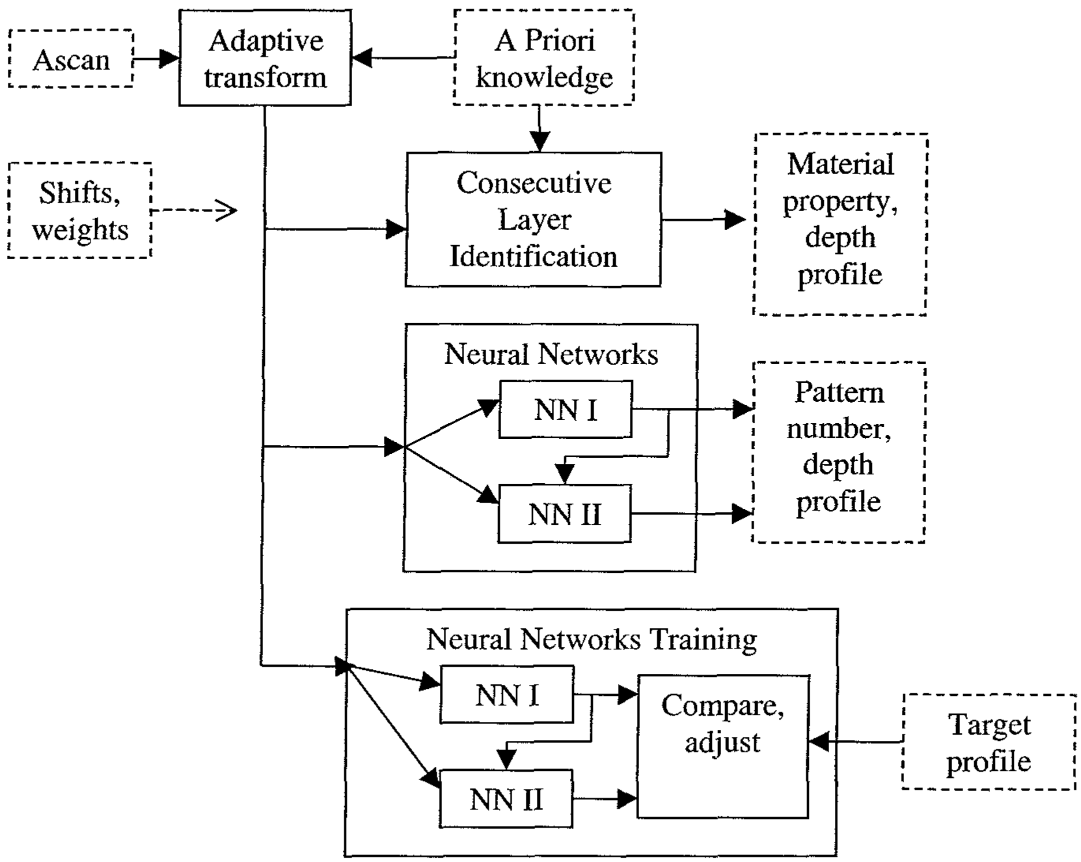

Figure 1. GPR Data Processing System.

1.3 System functionality

This GPR Delineation System is developed under Microsoft Visual C++ 5.0. It uses Microsoft Foundation Class (MFC) to provide a user interface that is similar to most Windows 95 applications.

The major components of the system include an Adaptive Transform unit, a Neural Network processing unit, a CLI unit and a Neural Network training unit. See Figure 1 for the relations among those units.

Users select a radar line to be processed, then choose an input pulse that matches most closely the input pulse when the data were collected. Users have the option to choose the entire line or just a portion of it for processing. The actual processing can be done by either the Neural Network or the CLI approach. It is also possible to determine whether the Ascans will be processed after averaging, sampling, or individually. Moreover, the users can train the Neural Network with user-defined profiles.

In the last version of this system, the result of the CLI approach is real time displayed on the screen. And the result of the Neural Network approach is stored in a file.

2. System Details

2.1 User interface

The interface is coded with MFC to provide a familiar and friendly user environment. A variety of dialog boxes are provided, including the standard file operation dialog boxes, dialog boxes for options and dialog boxes for specifying parameters. More details are given below.

File operations: users can invoke the 'File|Open' dialog box to open a radar line (data file), the parameters are shown and can be changed if necessary. In the future the software will be modified in order to save the results from different approaches into different files, and allow the printing of the results.

Options: Here the operator can use this dialog box to specify an Input Pulse to be used by the Adaptive Transform. One can configure a broad range of parameters, including frequency, amplitude, phase and even the pulse type. Apart from these choices, one can also construct their own Input Pulse, i.e., from experimental data, and import it as a file. Since using the appropriate options is critical for the performance. Default values are provided for the first-time use.

Run: This option allows whether the Neural Network or the CLI approach will be chosen. In addition, a range/number of Ascans can be selected for processing, and whether averaging or sampling will be applied to the Ascans to speed up the process.

Training the Neural Network: Whether a Neural Network will succeed depends not only on the structure, but also on the training. Training has to be sufficient in order to cover the entire sample space, and it must involve a certain degree of redundancy. For this reason, the users are provided with a choice to train the Neural Network by themselves. They can either build the Neural Network from scratch, or use the existing weight matrices as a base line. In the current version, we can only input an artificial configuration and use that set to train the Neural Network. Training by incorporating the field collected Borehole data is our currently ongoing research.

2.2 The Adaptive Transform Unit

The Adaptive Transform unit serves as the pre-processing stage for both the Neural Network and CLI approaches. Its task is to determine the impulse response in an adaptive/recursive process.

Let us consider the diagram given in Figure 2, where the input to the system is an Input Pulse produced by a particular antenna (transmitter). The system is the underground geophysical structure that affects and modifies the excitation pulse. The output is the Ascan that is recorded. If we assume the system to be linear and time invariant, we obtain the convolution $A(t) = I(t) * h(t)$. Thus, by deconvolving I(t) from A(t), the impulse response of the system can be found. This is the basic idea of the Adaptive Transform.

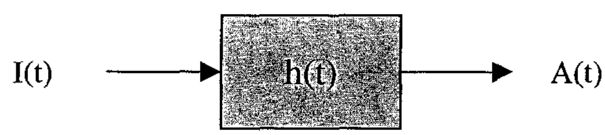

Figure 2. System representation of GPR signal processing.

The generic approach of the Adaptive Transform is that it first locates the signal with maximum reflection, subtracts it from the Ascan, then locates the second one, subtracts it as well, and so on. As a result, a series of time-stamped (time-shifted) impulses will be found and used as the impulse response of the system. After some ordering and/or justifying, those shifted impulses will be used as the input vector for the Neural Networks and the CLI subroutine.

It should be noted that only attenuation and delay are considered, distortion and dispersion are not taken into account.

2.3 The Neural Network Processing Unit

The complete Neural Network approach has been pursued in parallel with the CLI investigations. The advances of the NN system have been reported in our previous annual report (1997 end of year report). The NN effort is still continuing and various input/output formatting has been slightly modified in order to be incorporated into a common user interface along with the CLI and AT approaches. A brief description of this Neural Network approach including some of its pros and cons are summarized below.

There are two different Neural Networks in this unit, both use the *back propagation* algorithm. The first one, named NN I, is used for pattern recognition. It takes the rearranged impulse responses as input and categorizes it into one of the pre-defined geological patterns. Conventionally, we represent each reflected pulse by a pair of shifts and weights. The first half of the input vector consists of the weights in sequential time increments, the second half are the same weights but arranged according to magnitude such that larger weights go first. Each output node corresponds to one of the patterns, the maximum output is selected as the decision of NN I.

The second Neural Network, named NN II, uses the decision of NN I together with the shifts to determine the depth profiles. Since different patterns have different numbers of layers, there are different weight matrices for different patterns. Each output node corresponds to one specific layer of that pattern, and its output, after appropriate scaling, identifies the depth of that layer.

2.4 The CLI Unit

The analytical model currently under development within this approach makes the assumption that the input pulse is a plane wave. The underground structure is simplified into a one-dimensional stratified configuration. In this algorithm, the property (dielectric constant) of the first layer must be known as prior knowledge. This constraint is not viewed as significant since the top layer is the easiest layer to measure for its dielectric constant.

The CLI algorithm works in a recursive way to investigate individual layers one after another, from top to bottom. The amplitudes of the reflected pulses are first used to determine the reflection coefficients at each interface. That, in conjunction with prior knowledge, can be used to decipher the dielectric constants for all the layers immediately. Furthermore, the shifts are translated into travel time of the pulse; hence the depths of each layer can be recovered.

One of the difficulties encountered is that, due to proximity, an interface may have more than one reflection being recorded. As a result, the CLI algorithm has to keep track of the transmissions and reflections in order to distinguish and cancel the occurrence of multiple reflections caused by the same layer. Because of this, the accuracy of the CLI algorithm highly depends on the accuracy of the Adaptive Transform.

2.5 The Neural Network Training Unit

Training of the Neural Network is such an important issue that it needs to be discussed in detail. The training sources are discussed first, then the specific details regarding the actual implementation aspects are presented.

There are two sources that can be used as training samples. The first source is a set of synthetic Ascans generated according to the profile provided by user. The second source is the information from the Borehole data and the corresponding radar lines.

The first situation is well established and the training process has been developed and tested. It can provide a substantial degree of information to aid in choosing a particular type and configuration of the Neural Network. This synthetic generation of the GPR Ascans has been tested extensively and reported in our '97 end of year report. We had developed a numerical model of the governing physics associated with the GPR. Following verification of the model the system was used to create a variety of situations that were representative of the physical sites in question. These synthetic GPR Ascans could then be used to train the Neural Network since our knowledge of the actual signal and subsurface were known explicitely.

The second type of source uses the experimental data collected in the field. It is therefore more significant than the first type in a practical situation. However, a time-consuming amount of preliminary work is required for this actual training strategy. First, the correspondence of the radar line with the Boreholes has to be determined. Then relevant information has to be extracted from the Borehole data and mapped into one of several patterns. After that, the Ascans near the Borehole must be extracted from the radar line and pre-processed. Only then can the actually training begin. There is another difficulty using Borehole data for the training purpose. A large portion of Borehole records yields incomplete or skeptical information, which reduces the usefulness of the Borehole data. It is this latter constraint that we are addressing currently. Most of the other restrictions have been resolved successfully.

Like the processing unit, the actual training also has two parts. The first part is the training of NN I, which is somewhat straightforward. The second part is the training of NN II. In this part, different settings of the Neural Network have to be used for different patterns. In order to ensure optimum performance, the training targets (depths) have to be scaled according to the bounds.

3. CURRENT AND FUTURE WORK

3.1 Current Work

3.1.1 Training with Borehole Data

The most recent research has been attributed to the training with the second type of sources (see 2.5 for detail). Currently, two patterns have been selected and equipped with corresponding Ascans for the training purpose. Appendix A provides details about these two patterns. Specific information regarding the training sample sources, the table below summarizes all relevant parameters.

Borehole ID	Radar Line	Pattern #	Metermark	Selected Ascans
AP-6209	CR93-11	1	120	#1100-#1299
AP-6211	CR93-11	1	270	#2600-#2799
AP-6228	CR93-27	2	440	#2100-#2299
AP-6492	CR93-27	2	700	#3400-#3599

Table 1. Samples used in Training with Borehole data

Several issues are being carefully considered when conducting this mode of training. The first issue is space consistency between borehole data and radar lines. Each borehole record comes with a 'metermark', and in the header of each radar line file information about the space resolution (scans/meter) of the Ascans is found. Those two pieces of information can be employed to select the Ascans. However, the problem is that sometimes the radar line appears incomplete. For example, borehole records associated with CR93-11 have metermarks up to 1400m, which corresponds to Ascan #14000. Unfortunately the total number of Ascans in the file is only 8905. These type of inconsistencies could have serious effect on the training. We are retrieving original records to resolve such inconsistencies.

The second issue is the travel time consistency. All the borehole data and radar lines are collected in Alaska, some were acquired in the summer and autumn, some were gathered in winter and spring. Given that the borehole data and the associated radar lines were collected in different seasons, then the actual subsurface layer configuration and depth profiles could possibly exhibit significant differences.

Besides the two issues raised above, there is one additional problem that has to be solved. Past data processing work is based on a 7-pattern configuration as summarized in our '97 end of year report. However, not all the 7 patterns can be found in the borehole data, and some borehole data do not belong to any of the initial 7 patterns. Another set of patterns should be derived, which may not be fully compatible with the current set.

3.1.2 Different Approaches of Implementing the Neural Network Scheme

Currently there are three strategies of implementing the Neural Network approach. Each methodology has trade-off considerations. We are advancing all 3 modes from a research perspective. However, which method will yield the greatest return in the field has not been determined as yet.

First of all, a complete set of weight matrices was generated that can be loaded and used immediately. However, the codes used for training the Neural Networks require a significant amount of time to converge to a stable set of weights.

The second way is to use the synthetic data to retrain the Neural Network. Some modifications of the Neural Network code are also possible. The codes responsible for training had been already developed and integrated into the system. However, this method is of relevance for functionality testing only.

The third way is to use the borehole data to retrain the Neural Network. The advantages and disadvantages are outlined in 3.1.1.

3.2 Future Improvement

3.2.1 More Robust Adaptive Transform

The current implementation of the Adaptive Transform is memoryless. In other words, results from processing the previous Ascans do not affect the processing of the present Ascan. Consequently, the results of two adjacent Ascans could be dramatically different, although they may only be 0.1m or 0.2m apart.

One way to remedy this is problem is by using the results from the previous Ascans to gain some knowledge about the current Ascan. Adjustment could be made if necessary. Thus, the processing result will appear more continuous and the algorithm can be better immunized to noise. This history record development is being pursued.

3.2.2 Advance architectures in Neural Network

Thus far, only the Back Propagation (BP) type of Neural Network is used. BP Network is not the optimum choice for certain applications. There is a need to try other types of Neural Networks and compare their performance with the BP Network in our application.

3.2.3 New Approach Using the Inverse Formulation

As shown previously in this report, the identification performance highly depends on the Adaptive Transform, irrespective of the approach chosen. There is a need to try alternatives.

Under this scenario, an analytical inverse formulation has been studied. This method does not use the Adaptive Transform for pre-processing. And it does not rely on the simple model that was used in the CLI approach, (which is also used implicitly in the Neural Network approach). It is based on a more generalized model, and key operations are matrix operations. There are still more aspects to be explored, i.e., speed, noise tolerance, etc. But judging by the knowledge obtained so far, it would be a good candidate for our application.

Appendix A: Pattern configuration for Borehole data

General information: Pattern configuration

Note: Items in parenthesis may or may not be present

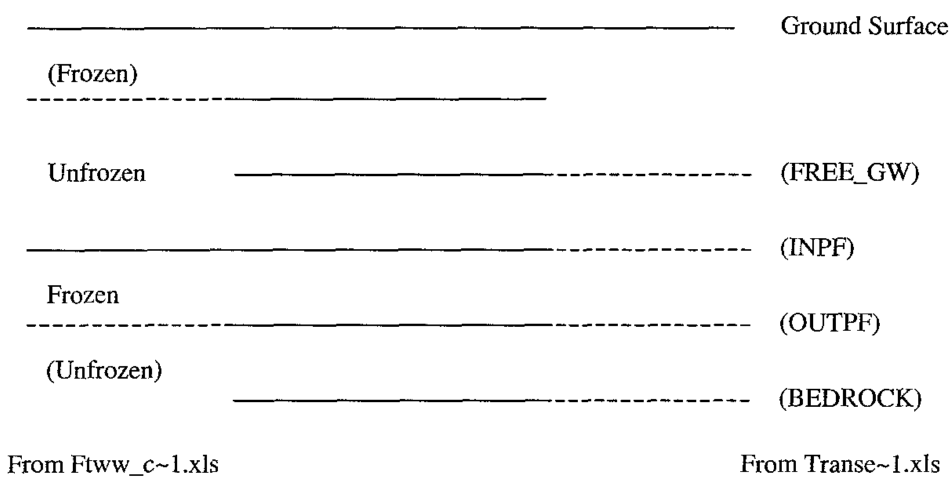

Figure A.1 General pattern configuration

Example: Pattern 1

Borehole AP-6209, correspond to CR93-11 metermark 120, all depths are in ft.
Note: The difference between the two data records is caused by the fact that they were collected in different seasons.

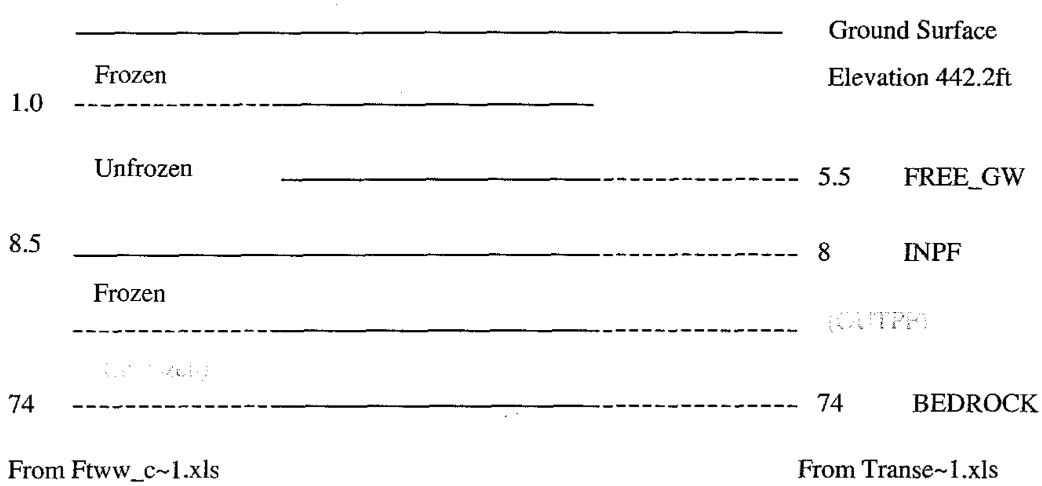

Figure A.2 Pattern configuration example 1

Dry sand	Saturated sand	Permafrost	Bedrock
0 - 5.5 ft	5.5 - 8 ft	8 – 74 ft	74 – ft

TableA.1 Parameters for pattern configuration example 1

Example: Pattern 2

Borehole AP-6492, correspond to CR93-27 metermark 700, all depths are in ft.

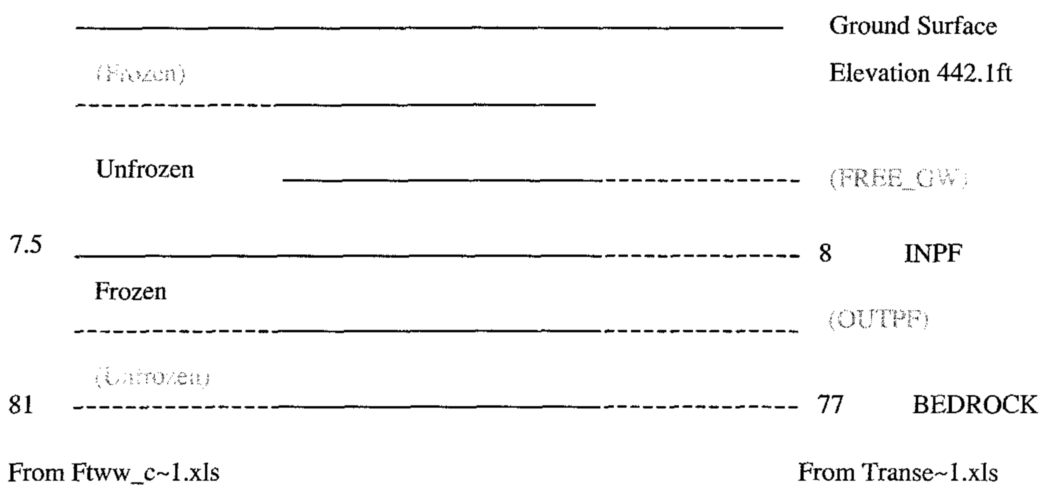

Figure A.3 Pattern configuration example 2

Dry sand	Permafrost	Bedrock
0 - 8 ft	8 – 77 ft	77 – ft

TableA.2 Parameters for pattern configuration example 2

Appendix B: User interface and system functionality

This appendix provides a brief introduction of the user interface and system functionality.

B.1 Overview

This program uses the Microsoft Foundation Class (MFC) to provide a user environment that is compatible with most Windows 95/98 applications. The main window is shown in Figure B.1. As usual, the software user can either invoke the menu or the tool bar to call the functions provided.

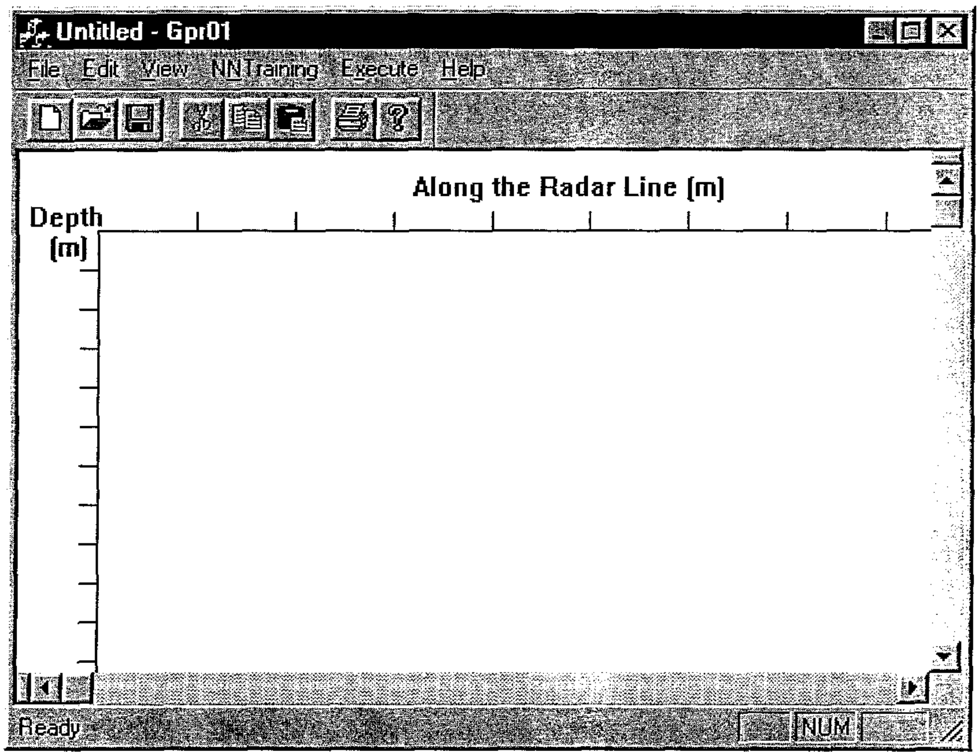

Figure B.1. Main window of the GPR data processing system

According to the nature of this system, several operations are of special interest. Those include, *File Open, Options*, *Processing the data*, and *Training the Neural Network*. Before going into a detailed discussion, it will be helpful to first understand how the system works.

1) To process GPR data, the first step is the selecting and opening of a GPR data file, i.e. the **.dzt** files. Since the processing highly depends on the Ground Penetrating Radar configurations at the time the data was collected, the data file header is shown right after the user open a file. The parameters specified in the header can be changed if necessary.

2) After a data file is opened, the user should check the options, i.e. the parameters of the *Initial Pulse,* which is crucial to the overall system performance. The user can either choose a pre-defined initial pulse or import a user-defined pulse from a separate file. However, it is advised that the default initial pulse be used at first.

3) The system is now ready to process the data file. Both the Neural Network approach and the Consecutive Layer Identification (CLI) approach are implemented. Thus the user can choose either one or both to process the data. Moreover, the operator can also specify other parameters that are relevant to the processing, e.g. the range of Ascans to be evaluated. The result of the CLI approach will be shown on the screen.

B.2 Opening a file

When the menu item File|Open is selected, or the Open button on the toolbar is clicked, a dialog box will appear and prompt the user to select and open a file, as shown below.

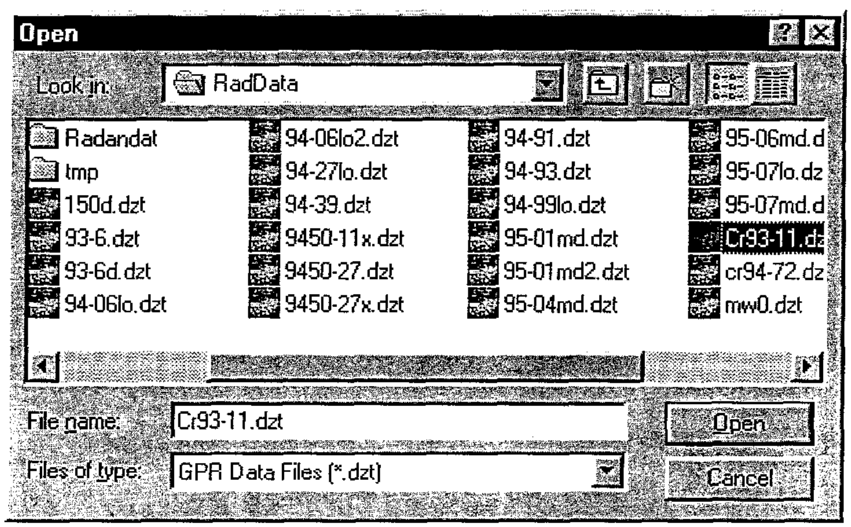

Figure B.2. The File Open dialog box

As discussed before, the information contained in the file header is shown to the user and can be changed if necessary. This is done via the dialog box shown in Figure B.3.

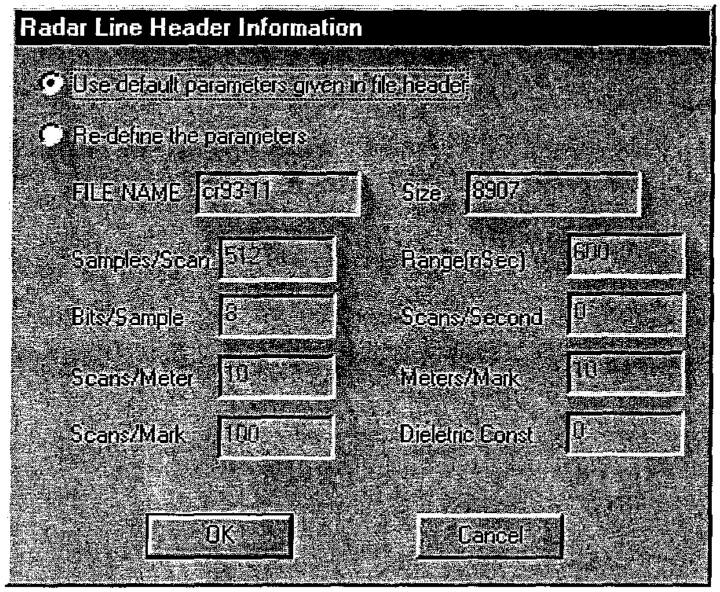

Figure B.3. GPR data file header information

By default, the user cannot change those parameters. Should it become necessary to change the parameters and one clicks on "Re-define the parameters", a warning message will be shown to alert the operator and to require further confirmation, as shown in Figure B.4.

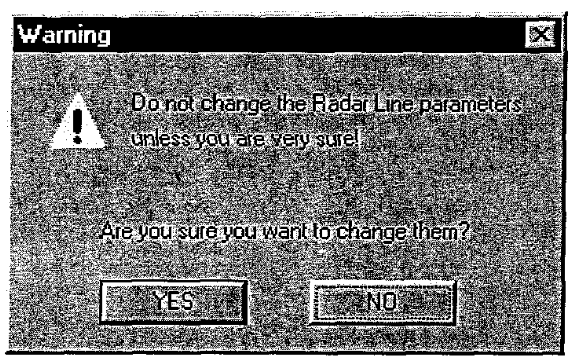

Figure B.4. Warning message for changing the radar line parameters.

B.3 Specifying Options for Initial Pulse

The options presented here allow the specification of an initial pulse that will be used by the Adaptive Transform to decompose the Ascans. Since both the Neural Network and the CLI approach will use the result of the Adaptive Transform, these parameters are crucial for a good identification performance. However, there is no direct way to determine the initial pulse from the data file itself. Therefore, the user has to select an appropriate initial pulse. This can be done in one of two ways. The first one is to choose a few parameters for two kinds of pre-defined pulses, as shown below. The second way is to generate the pulse by other means and import it as a file.

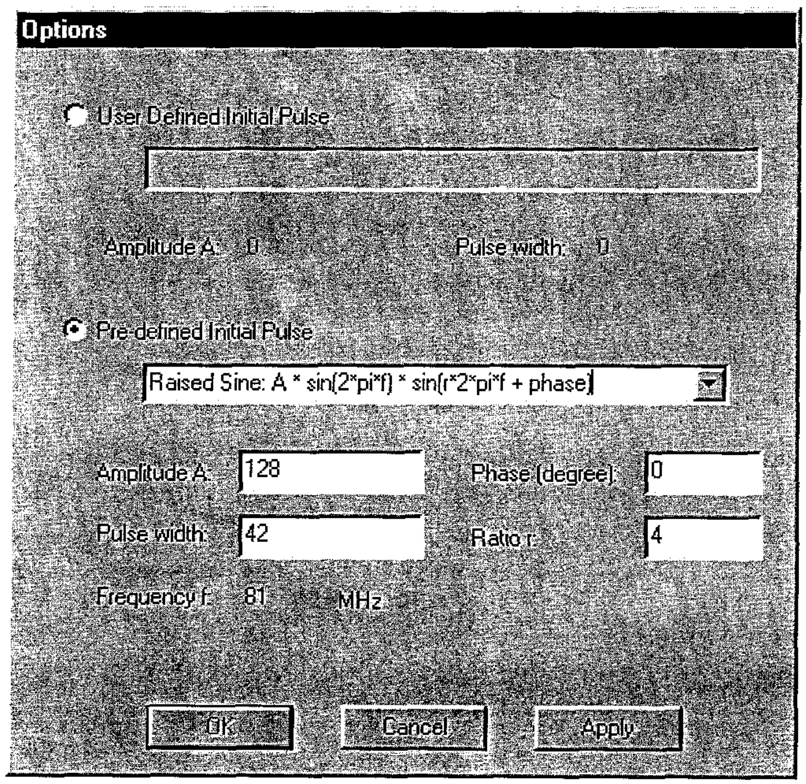

Figure B.5. Options for Initial Pulse.

A number of radar lines had been studied in detail in the previous research. As a result, there are default (suggested) parameters for those radar lines. The user is advised to use the default choices.

B.4 Processing the data

There are four options available to process the data, the user can choose either the Neural Network or the CLI approach to process the data, or use both approaches to process at the same time, either with or without collaboration. The first three options have been implemented, and the last one will be implemented in the near future. The result of the CLI approach will be displayed real time on the screen, while the result of the Neural Network approach will be saved to a file. Below is the dialog box used to specify different parameters.

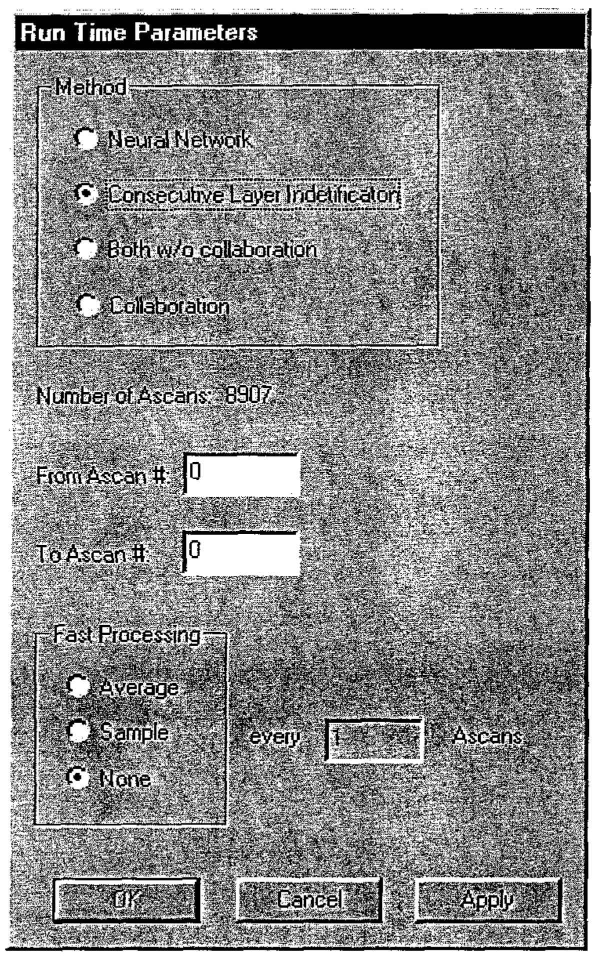

Figure B.6. Options used for processing the data

The total number of Ascans in the current radar line is given for the user's convenience. The user can specify a range of Ascans to be processed, of course that should not exceed the total number. Several fast-processing options are also provided for experimental purpose. The user can choose to process every single Ascan, one out of every 10 Ascans, or the average of 20 Ascans. For an entire new radar line, some general ideas would be helpful for choosing the various options mentioned in the previous sections. And that can be achieved by using these fast-processing options. The user can keep trying until an optimum set of parameters is found. Below is a screen shot of the result obtained by the CLI approach, processing one out of every 10 Ascans for a total of 120 Ascans.

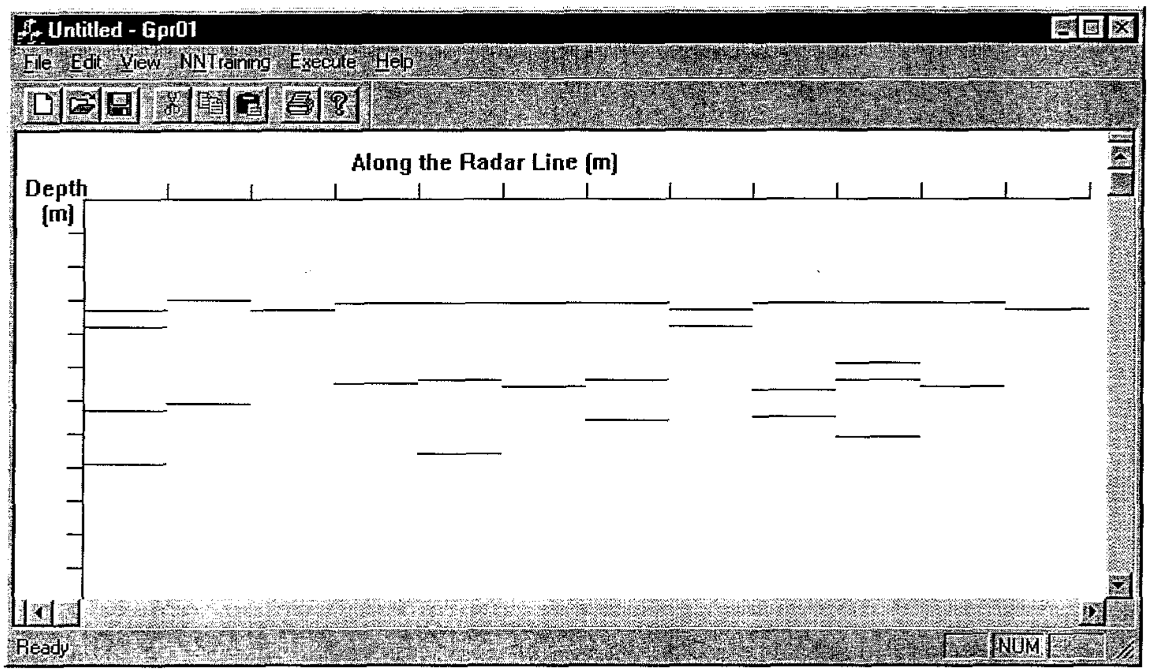

Figure B.7. Result of the CLI approach

B.5 Training the Neural Network

As discussed before, the Neural Network approach consists of two parts: one used to determine the pattern, and the second used to recover the depths. To train the neural networks, the user has to provide both the pattern number and the depth profile. As shown in Figure B.8, the user can choose from one of the six possible patterns, the material property is shown for reference, and the user can specify the minimum and maximum thickness of each layer. The program will randomly generate 500 samples within the range given by the user and uses those samples to train the Neural Network.

Currently, another training scheme has been developed and is being incorporated into this framework. This new training scheme uses the field-collected borehole data together with the corresponding Ascans as the training samples. For this scheme, it is important to choose an appropriate initial pulse before the actual training takes place.

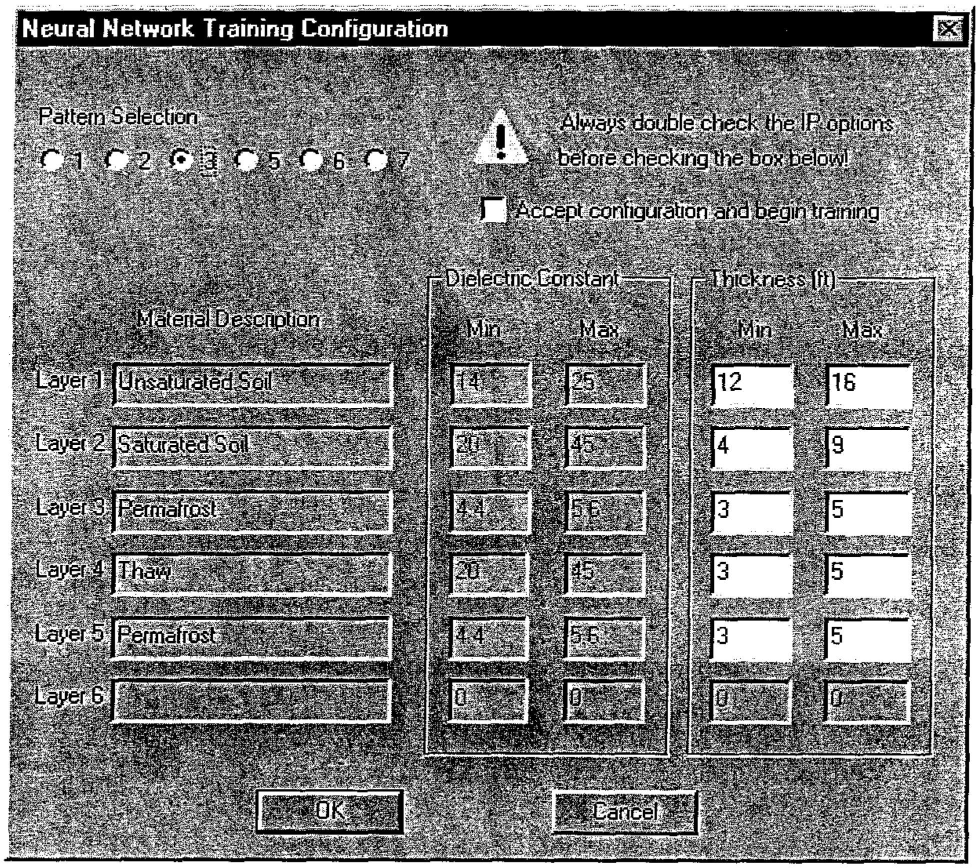

Figure B.8. Neural Network training configuration

Since the training process usually takes a certain period of time, a progress bar is shown to indicate the current progress of the training, as shown in Figure B.9.

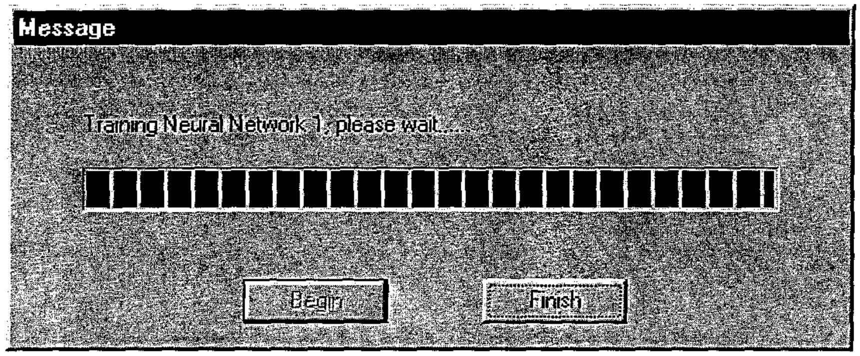

Figure B.9. Message during training