NEW MEXICO DEPARTMENT OF TRANSPORTATION

RESEARCH BUREAU

Innovation in Transportation

GRAPHICAL USER-INTERFACE (GUI) FOR GPR-BASED ANOMALY-DETECTION ALGORITHM

Prepared by:
Hayat Consulting, Albuquerque, NM

Prepared for:
New Mexico Department of Transportation
Research Bureau
7500B Pan American Freeway NE
Albuquerque, NM 87109

In Cooperation with:
The US Department of Transportation
Federal Highway Administration

Report NM800108

NOVEMBER 2018

1. Report No. NMDOT- 80-0108	**2. Government Accession No.**	**3. Recipient Catalog No.**
4 Title and Subtitle Graphical User-Interface (GUI) for GPR-based anomaly-detection algorithm		**5 Report Date** November 2018 **6 Performing Organization Code**
7. **Author(s)** Hayat, Majeed M		**8 Performing Organization Report No.**
9 Performing Organization Name and Address Hayat Consulting, 9227 Calistoga Ave., Albuquerque, NM 87122		**10 Work Unit No. (TRAIS)** 11 **Contract or Grant No**. NMDOT- 80-0108
12 Sponsoring Agency Name and Address Research Bureau New Mexico Department of Transportation 7500 Pan American Freeway NE Albuquerque, NM 87109		**13 Type of Report and Period Covered** Final Report, 2018 **14 Sponsoring Agency Code**

15 Supplementary Notes

16 Abstract

A Graphical User-Interface (GUI) has been developed that enhances a newly-created algorithm that is capable of scanning GPR data and providing graphical output of GPR anomalies. The capabilities include the ability to select and interpret raw data extracted from RADAN.DZT files, and associated metadata, process them, and present detection results to the end user in graphics mode, Excel tabular format and in MATLAB mode. The methods for automating parameter selection to simplify the interface controls for the end user have been developed. The GUI code has been organized into readable functions for modularization purposes, which is then optimized for run-time performance. Refinements have been made based on Procuring Agency inspection and feedback. A user manual for the GUI provide training on using the GUI to Procuring Agency has been produced and appropriate training on using the GUI has been provided.

17 Key Words GPR, algorithm, detection, correlation, anomaly detection, lamination, distress, radar		**18 Distribution Statement** Available from NMDOT Research Bureau	
19 Security Classification (of this report) None	**20 Security Classification (of this page)** None	**21 No. of pages**	**22 Price**

Form DOT F 1700.7 (8-72)

PROJECT TITLE: GRAPHICAL USER-INTERFACE (GUI) FOR GPR-BASED ANOMALY-DETECTION ALGORITHM

Final Report

by
Majeed Hayat, Ph.D.
Hayat Consulting
9227 Calistoga Ave., NE, Albuquerque, NM 87122

Report No. NMDOT- 80-0108

November 2018

Research Bureau URL: http://www.NMDOT Research.com
Research Bureau Email: Research.Bureau@state.nm.us

A Report on Research Sponsored by
New Mexico Department of Transportation
Research Bureau

in Cooperation with

The U.S. Department of Transportation
Federal Highway Administration

http://www.Research.com
Research@state.nm.us

NMDOT Research Bureau
7500B Pan American Freeway NE
PO Box 94690
Albuquerque, NM 87199-4690
(505) 798-6730

PREFACE

The research reported herein develops a Graphical User-Interface that enhances a newly-created algorithm that is capable of scanning GPR data and outputting GPR anomalies.

NOTICE

DISCLAIMER

ABSTRACT

A Graphical User-Interface (GUI) has been developed that enhances a newly-created algorithm that is capable of scanning GPR data and providing graphical output of GPR anomalies. The capabilities include the ability to select and interpret raw data extracted from RADAN.DZT files, and associated metadata, process them, and present detection results to the end user in graphics mode, Excel tabular format and in MATLAB mode. The methods for automating parameter selection to simplify the interface controls for the end user have been developed. The GUI code has been organized into readable functions for modularization purposes, which is then optimized for run-time performance. Refinements have been made based on Procuring Agency inspection and feedback. A user manual for the GUI provide training on using the GUI to Procuring Agency has been produced and appropriate training on using the GUI has been provided.

TABLE OF CONTENTS

1. Introduction

Subsurface delamination, stripping, erosion, water accumulation and presence of air-voids are examples of problems that occur over time in pavements and can adversely affect their integrity. While there have been a number of studies on the analysis of pavement integrity using GPR data [1-4], they have traditionally focused on using the GPR data to solve the inverse problem of estimating the unknown structural and electromagnetic properties of the pavement layers, e.g., number of layers, thickness of each layer, the permittivity of each layer, etc. The solution to such inverse-problem is essentially performed in two steps. First, a parametric family of physics-based pavement models (over various pavement structures, e.g., number of layers and their thicknesses, as well as the electromagnetic properties of the layers for each possible structure) is assumed. Second, the model that best fits the data is identifying through extensive and exhaustive computations. Of course, certain data calibration and processing (e.g., surface reflection removal, noise reduction, etc.) needs to be performed *a prior* and along the way to make such solution possible. Solving such inverse-problem is a difficult task in general, especially in cases when little knowledge can be assumed about the pavement structure *a priori*. Regardless of the difficulty of the inverse problem, once (and if) it is successfully solved for each segment of the pavement, the examination of the estimated layers and their estimated electromagnetic properties can be used by an expert as a basis to determine if the pavement is sound or if its integrity has been compromised. We point out that the success of such anomaly detection approach depends on the success of two steps: (1) the reliable solution to the inverse problem for the pavement layers, and (2) the effective analysis of the estimated layer details for anomalies. The first step maps the GPR data to a high-dimensional (detailed) space of pavement layers and their electromagnetic properties, and the second step attempts to make an expert determination of pavement integrity based on the estimated detailed information. Typically, the solution of the inverse problem is time consuming, requiring analyst's intervention at various stages of the solution. Also, the estimated information from solving the inverse problem is very detailed, and anomaly detection requires an expert to extract relevant information from the high-dimensional information. To the best of our knowledge, no reliable approach for front-end subsurface anomaly detection is available that is simple enough to be automated fully for large stretches of pavement.

An automated solution to anomaly detection in pavements would be very valuable for pavement engineers since it can enable them to flag segments of pavements for coring and detailed diagnosis in an efficient and cost-effective manner. Since GPR scans are typically collected at a rate of several scans per meter, the application of the inverse-problem approach is practically impossible for long stretches of pavements. It is therefore critical to have an alternative method that can be automated and applied to GPR data over very large (tens of miles) stretches of pavement. The goal of this effort is to develop an algorithm, as well as an accompanying computer program, to process GPR data and automatically flag anomalous scans, which can be identified as candidates for distressed segments of the pavement.

1.1. Data-driven algorithm developed by Hayat in an earlier NMDOT research project

In an earlier research project with the NMDOT, Dr. Hayat has developed a new data-driven method for pavement anomaly detection using GPR data. Unlike the inverse-problem approach described above, the determination of anomalies is achieved by observing the GPR data directly, without the need to first mapping the data to a high-dimensional space representing the pavement's detailed physical model. The justification for this approach is as follows. Many types of anomalies, such as air-voids, layers' erosion or collapse, water pockets, abrupt changes in layers' depths, etc., can be detected directly from the traces of the reflected pulses without the need for having the full knowledge of the structure and composition of the underlying pavement. In other words, while the detailed information about the pavement's physical model along the stretch of a pavement can be sufficient for determining anomalies, it is not necessary. The value of such direct approach is that it requires minimal expert intervention, and it can be automated effectively and efficiently. The rationale of our approach is that GPR traces tend to maintain continuity, over time, in their characteristics over pavement segments that do not suffer from distress or other anomalies. For example, the number and location of peaks and valleys in each trace remain the same over a stretch of scans in a sound pavement segment. In contrast, any observation of discontinuities in the attributes of the traces can be utilized to directly flag subsurface anomalies. Hence, if we have a method to measure such "continuity" of the "trace attributes" over scans, then we may be able to detect anomalous scans that represent distressed.

In the earlier project have develop an algorithm to do just that. The input to the algorithm is the series of scans over a long stretch of pavement, and its output is the set of flagged scans. To identify anomalies, the algorithm computes and analyzes tracks of five metrics, which are grouped into two categories: (1) change in correlation in the trace from scan to scan (three metrics in this category); and (2) change in the presence of negative and positive peaks (two metrics in this category). As expected, certain thresholds need to be set for the detection process, and this is the only part where human interaction is required. The algorithm also identifies the locations at which the new peaks or valleys appear.

The algorithm has been implemented in Matlab, and its efficacy has been validated in two ways. First, we compare the outcomes of the algorithm, when applied to real GPR data, to results from the visual inspection of GPR data by an expert (using the propriety RADAN GPR visualization software). Second, we compare the results of the algorithm to the results obtained from coring. We have also used the algorithm to identify locations that need to be cored, and the results of such algorithm-guided coring were also analyzed. Overall, the results show that the proposed method is very promising in providing an automated solution to a front-end determination of anomalies in pavements. In its present form, however, the algorithm does not inform the user of the types of anomalies, not does it have the capability to distinguish between man-made anomalies, such as bridges and other structures, and natural defects in the pavement layers. Such capability can be investigated in futures works.

Finally, while the algorithm developed in the earlier project is targeted toward two-dimensional radar (each scan represents one longitudinal location of the pavement), we believe it can be extended to work in a three-dimensional GPR setting, where scans provide both transverse and

longitudinal information at various subsurface depths of the pavement. An outline of the steps toward such generalization is also included in this report.

1.2. Goals of this project

The goal of this contract is to develop a Graphical User-Interface (GUI) that enhances the newly created algorithm, as described above, which is capable of scanning GPR data and providing graphical output of GPR anomalies automatically. This includes the ability to select and interpret raw data extracted from RADAN.DZT files, and associated metadata, process them, and present detection results to the end user in graphics mode, Excel tabular format and in MATLAB mode. The Contractor will develop methods for automating parameter selection to simplify the interface controls for the end user. The GUI code will be organized into readable functions for modularization purposes, which will then be optimized for run-time performance. Refinements will be made based on Procuring Agency inspection and feedback. The Contractor will develop a user manual for the GUI provide training on using the GUI to Procuring Agency staff as needed.

2. Task-by-task status of the completed project

All deliverables have been completed. Below is a list of each deliverable, task and subtask, as well as their completion status.

Deliverable 1: Code Base Finalization

Subtask 1.1 At the direction and assistance of the Procuring Agency's Technical Panel ("Panel"), the Contractor shall review current code and outputs to determine final structure of application. During this process the code will be assessed for speed and potential optimizations and will include the following:

- Thorough review of current code and identify functional blocks and relevant parameters for GUI interfacing
- Separate relevant display functions from the main code base into display methodologies that will be incorporated into the GUI plan;
- Logical functions will be written, the code modularized and parameterized;
- Validate streamlined code against original code.
- Extend and verify the code to allow selection of one of three typical options for the type of the low-pass filter when de-noising the raw data.
- Extend and verify the code to allow assigning arbitrary weights for each type of metric when visualizing the composite anomaly tracks.

- Extend and verify the code to display the horizontal access both in miles or scan number.
- Extend and verify the code to generate anomaly tracks as a 3D plot, where anomalies flagged by individual metrics are shown separately on a third axis.
- Extend and verify the code to allow arbitrary specification of the width and shape of the moving-average filter, which is applied to the anomaly tracks, in order to ease the visualization of problematic regions.

Status: Completed on July 18, 2018. The completed work was presented to the Panel.

Subtask 1.2 Contractor shall complete this task in tandem with Sub-Task 1.1. The Contractor shall define and implement parameter automation strategies, which includes the following:

- Identify target parameters for automation (thresholds for Radio Frequency interference, sensitivity of anomaly detection, bandwidth for low-pass filtering, etc.).
- Develop and implement code for automation of identified parameters.
- Incorporate automation code into the main code base.

Status: Completed on July 18, 2018. The completed work was presented to the Panel.

Subtask 1.3 The Contractor shall meet with the Panel as needed to provide work updates and receive direction.

Status: Completed on July 18, 2018. The completed work was presented to the Panel.

Deliverable 2: Develop and Test GUI

Subtask 2.1 At the direction and assistance of the Panel the Contractor shall build the GUI interface on top of the improved code, which includes the following:

- Create a mapping between GUI components and code base parameters and desired user interactions;
- Develop and deliver mock-up of the GUI through visuals and descriptions for review and approval of the Panel.

Status: Completed on August 1, 2018. The completed work was presented to the Panel.

- Design the GUI and develop base functionality;
- Test and validate all functionality and support user testing and acceptance;

- Demonstrate the GUI functionality to the Panel and collect feedback;

Status: Completed August 31, 2018. The completed work was presented to the Panel.

- Modify the GUI as needed based on feedback from the Panel;
- Meet with the Panel as needed to verify that updates are satisfactory.

Status: Completed on September 26, 2018. The completed work was presented to the Panel. The GUI''s Matlab code, v1.2.4, was delivered to Mr. Valdez and the Panel on September 26, 2018.

Subtask 2.2 The Contractor shall develop a **user manual** to be reviewed by the Panel and Procuring Agency staff will be trained on usage. The Contractor and the Panel will coordinate to determine the location and length of training course.

Status: Completed. User manual delivered to the Panel by email on September 26, 2019. The User Manual is attached to this final report. See Appendix. The Contactor guided the Panel through running the GUI on October 19, 2018. The Panel was asked if additional training was necessary. The Panel will inform the Contractor if additional training is required or if the Panel has additional questions.

Subtask 2.3 The Contractor shall submit a **final report** documenting the overall scope and results of the project. The report shall document all activities performed including a detailed description of the work, findings and recommendations. A draft of the Final Report shall be delivered to the Panel for comment and approval. A final report shall be provided to the Panel by contract termination.

Status: Completed. A draft of the Final Report was submitted on Oct. 22, 2018. The Panel inform the Contractor within two weeks of this submission if any revisions to the Final Report or the GUI are required.

3. Data-driven algorithm developed in the earlier research project

In the Appendix we give a detailed description of the algorithm that is to be enhanced, automated and coded to produce a GUI. Readers who are familiar with the algorithm and are interested in the GUI may skip this section and the Appendix.

4. Description of graphical user interface (GUI)

4.1. Software Description

The NMDOT – *Hayat Processor* is a MATLAB-based Graphical User-Interface (GUI) that advances a newly created algorithm that is capable of scanning GPR data and providing graphical output of GPR anomalies. This includes the ability to select and interpret raw data extracted from RADAN.DZT files (and associated metadata, including scan rate and scan numbers), process them, and present detection results concisely to the end user in graphics mode, Excel tabular format and in Matlab mode; the latter can be inspected and further analyzed by the user in a Matlab environment. It also includes refinements of the original code to include methods for automating parameter selection (such as surface reflection removal, metal-plate pulse isolation, bandwidth selection for noise removal, etc.) to simplify the interface controls for the end user. Additional refinements to improve the functionality of the code have also been made, as detailed in the stamen of work and subject to recommendations made based on NMDOT the technical panel. The prototype MATLAB code has been organized into readable functions for modularization purposes.

4.2. Installing and Running the Application

The main software application is written in MATLAB and as such requires MATLAB R2015a or later. All data export is written to Microsoft Excel files and as such requires that Excel be installed for export functionality to be enabled. The software is provided in a .zip file format. To install the software simply extract the NMDOT_v_x.x.zip file (where x.x is the version number) to a desired location. Upon successfully extracting the .zip file, the root folder will contain NMDOT.m and NMDOT.fig in addition to the "HayatProcessor" folder that contains all major processing functions. Within this folder is a further subfolder that contains the .DZT reader functions initially provided by GSSI (note that these have been modified for bug fixes during the development process by the authors). It is imperative that this file structure be in tact in order for the software to operate properly.

Once the software has been properly installed, simply double-click on the NMDOT.fig file. This will launch the main software graphical user interface within MATLAB. The next section describes software usage in detail.

4.3. Software Usage

Upon running the software, the main GUI window loads, as shown below in Fig. **1**. The GUI is logically organized into different panels that contain various input controls, display controls, and tables that display scan metadata and processing results. The individual panels are described in detail in the following subsections.

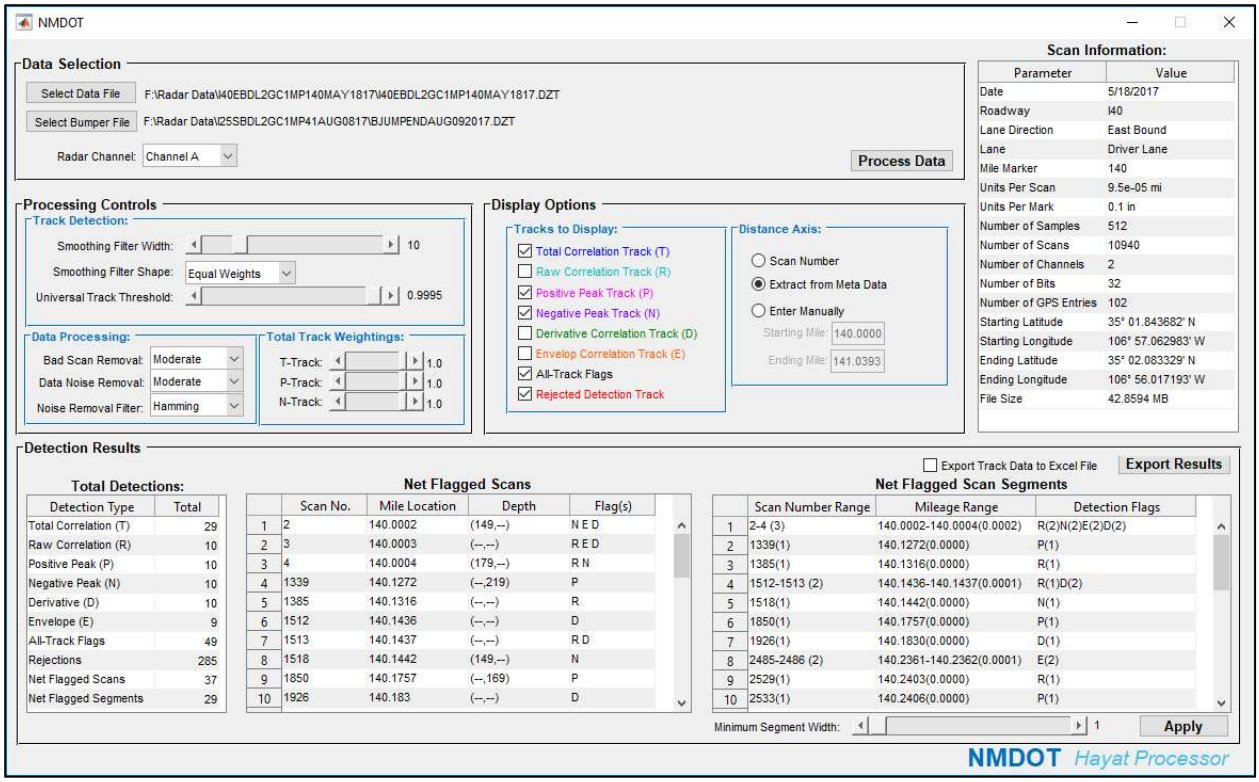

Fig. 1. Main Graphical User Interface for the NMDOT – Hayat Processor MATLAB application.

Configuration for Processing

The first step in configuring the software is to select both a data file and a bumper file for processing. These options are contained within the **Data Selection** panel at top-left in the main GUI window, shown more closely in Fig. **2**. Clicking on the **Select Data File** button will open a file selection window. Navigate to a .DZT data scan file, select it, and then select **Open**. Repeat these steps for selecting a bumper file using the **Select Bumper File** button.

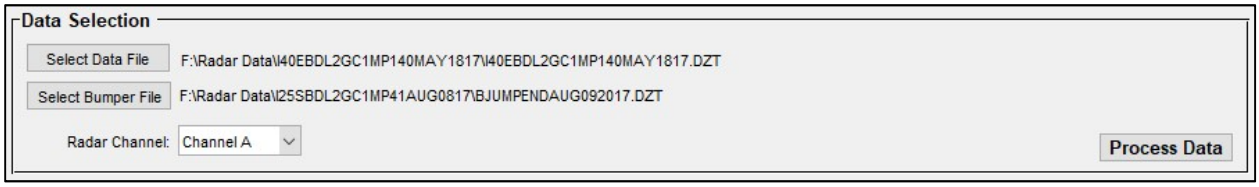

Fig. 2. Data selection panel used to specify the data file, bumper file, and radar channel for processing.

Upon successfully selecting a file, the **Scan Information** table will be populated with various metadata, as shown in Fig. **3**. In some cases not all metadata will be available. For example, the **Date**, **Roadway**, **Lane Direction**, **Lane**, and **Mile Marker** field values are extracted from the .DZT data filename if available. Otherwise, these fields will be populated with "**not available.**"

The **Units Per Scan** and **Units Per Mark** field values are obtained from a corresponding .DZX file, which is sometimes not available. In this case the value field will be displayed as "**not available.**" The **Number of Samples**, **Number of Scans**, **Number of Channels**, **Number of Bits** and **File Size** fields are always available from within the data file itself and will always be displayed. The **Number of GPS Entries**, **Starting Latitude**, **Starting Longitude**, **Ending Latitude**, and **Ending Longitude** fields are obtained from a corresponding .DZG file and may not be available.

Scan Information:

Parameter	Value
Date	5/18/2017
Roadway	I40
Lane Direction	East Bound
Lane	Driver Lane
Mile Marker	140
Units Per Scan	9.5e-05 mi
Units Per Mark	0.1 in
Number of Samples	512
Number of Scans	10940
Number of Channels	2
Number of Bits	32
Number of GPS Entries	102
Starting Latitude	35° 01.843682' N
Starting Longitude	106° 57.062983' W
Ending Latitude	35° 02.083329' N
Ending Longitude	106° 56.017193' W
File Size	42.8594 MB

Fig. 3. Upon selection of a data file the Scan Information panel will display all indicated meta data when available.

Also, after the data scan file has been selected, the Radar Channel dropdown box will be updated with all available channels that can be selected for processing, including the aggregate channel "**A+B.**"

Once a data and bumper file and radar channel have been selected, the **Processing Controls** panel can be used to change various processing parameters that will be used to process the data, which is shown in Fig. **4**. The values displayed in the figure are the default values for the application and should give reasonable results on average.

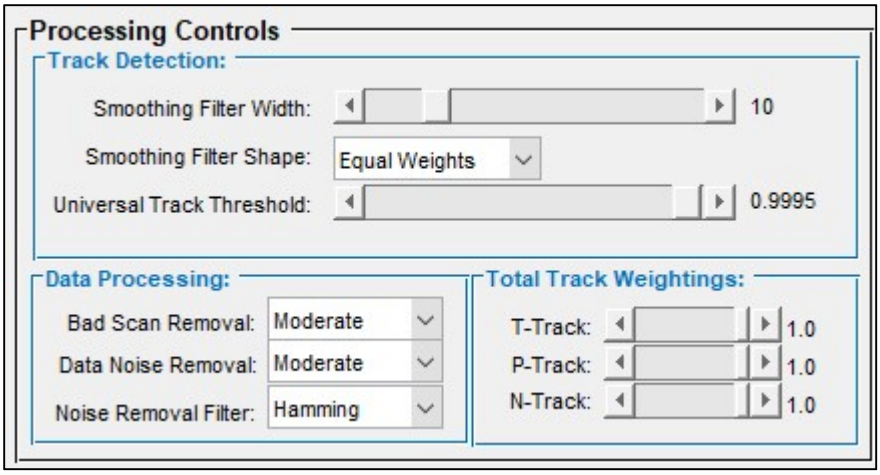

Fig. 4. Processing Controls panel used to specify various options used for processing a given data scan.

The **Track Detections** subpanel contains several parameters involved with the generated the various detection tracks. The **Smoothing Filter Width** controls the width of the smoothing filter that is applied to the tracks to smooth track noise. The **Smoothing Filter Shape** controls the types of coefficients used in the smoothing filter. All filter shapes are symmetric and three options are provided: **Equal Weights** (rectangular), **Gaussian** (bell curve), and **Bartlett** (triangular). The latter two simply allow for the central track value to be weighted more heavily than the surrounding values. The **Universal Track Threshold** specifies the probability value used to threshold the probability of detection. Decreasing this value will results in an increased number of detections. The more this value is increased the more likely false detections will be observed. Keeping this value at the default value of **0.9995** is recommended during an initial processing run.

The **Data Processing** subpanel provides controls for removing bad scans and reducing noise within the data. The **Bad Scan Removal** dropdown box allows the user to specify thresholds for the detection and removal of bad scans within the data with the options: **Mild**, **Moderate**, or **Aggressive**. The **Data Noise Removal** dropdown allows the user to select the amount of noise removal as: **None**, **Mild**, **Moderate**, or **Aggressive**. The corresponding **Noise Removal Filter** shape can also be selected from the following filter window types: **Ideal**, **Rectangular**, **Hamming**, **Hanning**, **Bartlett**, **Blackman**.

The **Total Track Weightings** subpanel allows the user to select the contributions of the various detection tracks that are mixed to produce the **All-Track Flags** detection Track. The **T-Track** slider provides a weighting for the **Total Correlation Track (T)**, the **P-Track** slider provides a weighting for the **Positive Peak Track (P)**, and the **P-Track** slider provides a weighting for the **Negative Peak Track (N)**. Each slider can be set in the range of [0.0, 1.0].

Once all data files are selected and processing parameters selected, simply press the **Process Data** button. During processing a progress bar will appear indicating the step and progress through the

processing procedure. The command window will display more detailed progress about the individual scans that have been processed. Once completed, detection tracks will be displayed in a MATLAB "Figure 1" window and information about any track detections will be displayed in the tables under the **Detection Results** panel. Each of these sections is described in detail in the next two sections.

Detection Results Displays

Once processing is complete, detection results are displayed in various ways inside of the application. Several different detection tracks are generated during the processing phase. These different detection tracks can be displayed in MATLAB's "Figure 1" window, as shown below in Fig. **5**. The various detection tracks are displayed both as a 2D plot (top) as well as a 3D plot (bottom). The **Display Options** panel, shown in Fig. **6**, provides the user with the ability to control which detection tracks are displayed in "Figure 1" under the **Tracks to Display** subpanel. Simply check the box next to each track to enable/disable it from being displayed. Notice also that the detection tracks are color-coded in both the **Tracks to Display** panel and in the figure.

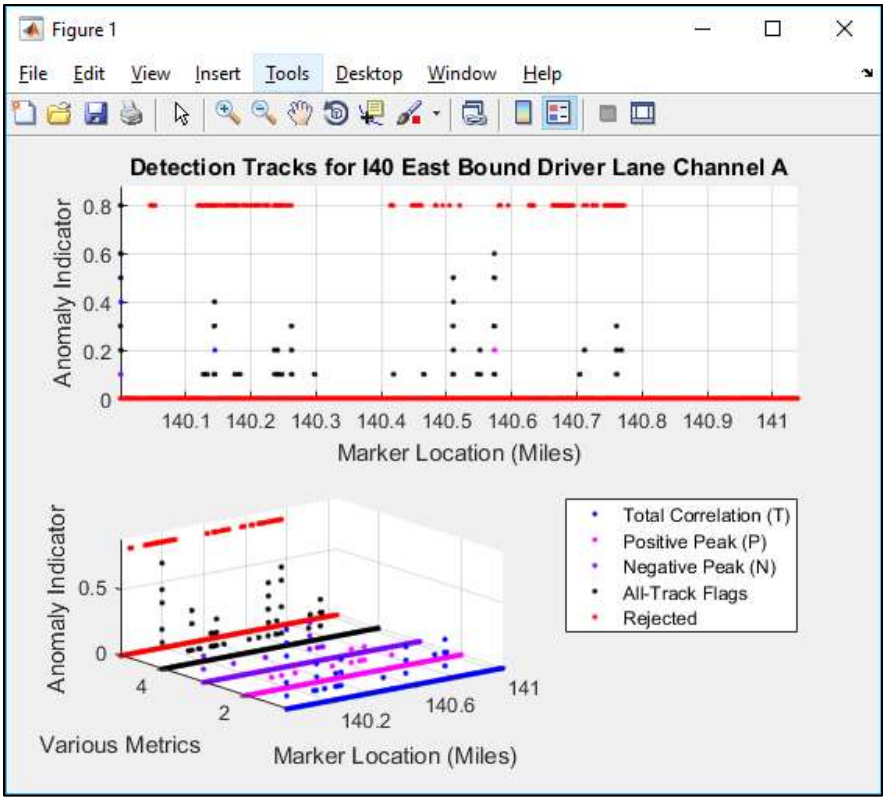

Fig. 5. After processing, "Figure 1" displays the different detection tracks according to the Display Options panel.

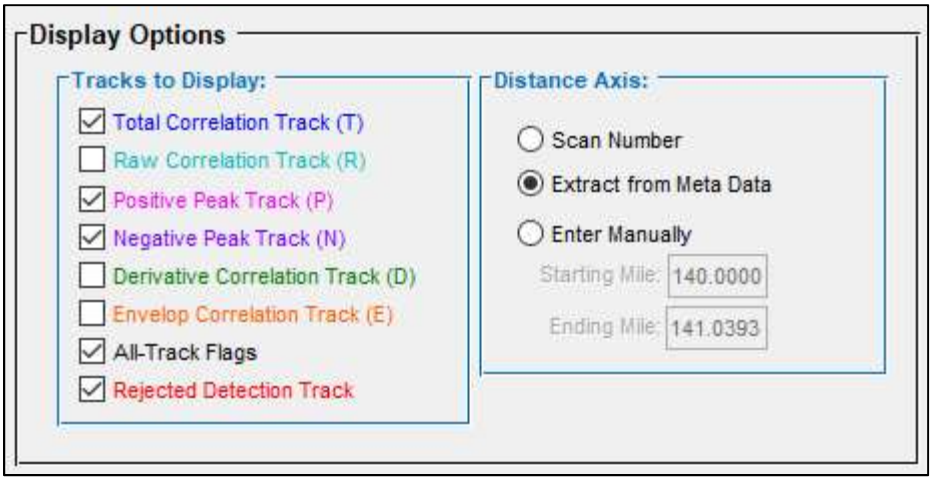

Fig. 6. The Display Options panel allows various detection tracks to be displayed and specification of distance axis units.

Also shown in the **Display Options** panel is the **Distance Axis** subpanel. This subpanel provides the user with the ability to specify the horizontal plot axis units for each detection track. There are three different options that can be selected according to each radio button. **Scan Number** simply sets the x-axis to an integer index. The **Extract from Meta Data** option will set the x-axis unit to miles when metadata is available (this option is the default when metadata is available). When this information cannot be determined from the metadata, this option will be grayed out and thus not be able to be selected. When this option is not available the **Scan Number** option will be selected by default. The third option, **Enter Manually**, allows to user to specify a **Starting Mile** and **Ending Mile** and the x-axis will be created accordingly.

A number of tables are populated with information upon completion of data processing. The number of detections found from each detection track are summarized and displayed in the **Total Detections** table, shown in Fig. **7**. Additionally, the **Net Flagged Scans** and **Net Flagged Segments** are also displayed. A net flag is created when one or more detections are found at a given scan line. The **Net Flagged Scans** refer to individual detections found at a single scan location. **Net Flagged Segments** refer to sets of adjacent scans that are grouped together as a single segment detection. Thus, the number of **Net Flagged Segments** will always be less than or equal to the number of **Net Flagged Segments**.

Total Detections:

Detection Type	Total
Total Correlation (T)	29
Raw Correlation (R)	10
Positive Peak (P)	10
Negative Peak (N)	10
Derivative (D)	10
Envelope (E)	9
All-Track Flags	49
Rejections	285
Net Flagged Scans	37
Net Flagged Segments	29

Fig. 7. The Total Detections table displays the number of detections from various detection tracks as well as aggregate net detections counts.

The **Net Flagged Scans** table lists all individual scans where one or more detections are found, as shown in Fig. **8**. For each flagged scan, the scan number and corresponding mile location are listed. The tracks where a detection was found are indicated in the Flag(s) column. The depth is associated with the positive and negative peaks. As such, the depth is only listed if the (P) or (N) flag is present. The depth corresponding to the (N) flag is always listed first and the depth corresponding to the (P) flag is listed second. Also, by default, if the mileage information is not available from metadata, the milepost locations will not be listed by default if the x-axis is specified in units of Scan Number. However, if the mileage is manually set for the x-axis the Mile Location column will update accordingly (and can be updated any time after processing to reflect any changes to the manually entered mileage range).

Net Flagged Scans

	Scan No.	Mile Location	Depth	Flag(s)
1	2	140.0002	(149,--)	N E D
2	3	140.0003	(--,--)	R E D
3	4	140.0004	(179,--)	R N
4	1339	140.1272	(--,219)	P
5	1385	140.1316	(--,--)	R
6	1512	140.1436	(--,--)	D
7	1513	140.1437	(--,--)	R D
8	1518	140.1442	(149,--)	N
9	1850	140.1757	(--,169)	P
10	1926	140.183	(--,--)	D

Fig. 8. The Net Flagged Scans table display all individual flagged scans. For each scan, the scan number, mile location (if available), the depth information (if a positive or negative peak), and flags contributing to the detection are shown.

The **Net Flagged Scan Segments** table, shown in Fig. **9**, is similar to the Net Flagged Scans table, except the net flagged scans are condensed into groups of adjacent detections. Here, the scan number range of the segment is listed in the Scan Number Range column, along with the number of net flagged scans that are contained within the segment (contained within parenthesis). The corresponding Mileage Range (if available) is also listed for the segment. The distance in miles of the segment is listed in parenthesis. Finally, all detection flags found within this segment (along with the counts of each flag type contained within parenthesis) are listed in the Detection Flags column. Below this table is a slider bar entitled **Minimum Segment Width**. This slider acts as a filter and will only list segments that are greater than or equal to the minimum number of segments specified. For example, if this slider is set to a value of 2, rows {2, 3, 5, 6, 7, 9, 10} would not be displayed in the table of Fig. **9** since these rows have segments that are of length 1. Each time the slider is changed the user must press the **Apply** button in order for the table to update.

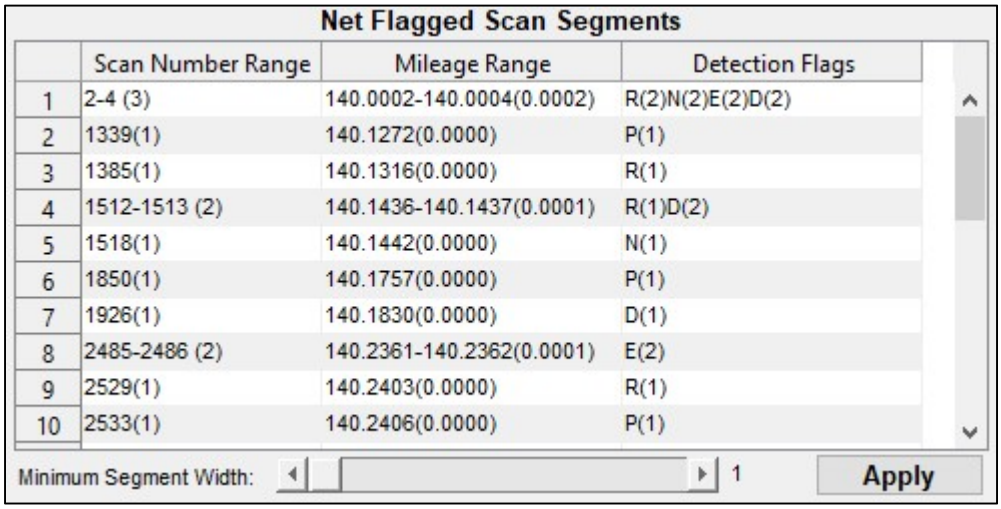

Fig. 9. The Net Flagged Scan Segments table displays net detections segments that constitute sequential flagged detections. The segment results can be filtered by specifying the minimum segment width.

4.4. Exporting Results

The results of a given data processing can be exported to an Excel file using the **Export Results** button, shown in Fig. **10**, located above the **Net Flagged Scan Segments** table. Upon pressing the **Export Results** button the user will be prompted with a **Save File Dialog** to select the name and location of the export file. By default, the first sheet of the excel file contains all processing parameters and scan file metadata information used to generate the processing results. The second sheet contains the output of the track display plots of MATLAB's "Figure 1" along with the table results from the **Total Detections**, **Net Flagged Scans**, and **Net Flagged Scan Segments** tables. To the left of the **Export Results** button in a checkbox labeled **Export Track Data to Excel File**. If checked, upon export the data from each individual detection track, both the x and y-axes, will be outputted to sheet 3 of the Excel file. Please note that this option takes a considerable amount of time if selected and is disabled by default.

Fig. 10. The processing results from a given scan data file can be exported to an Excel file using the Export Results button.

It is also worth noting that the contents of MATLAB's "Figure 1" can be directly copied to the clipboard from the figure window itself. To do so, select the **Copy Figure** option from the Edit menu inside of the figure window, as shown in Fig. **11**. The **Copy Options…** menu item can be used to control how the figure is copied to the clipboard if desired.

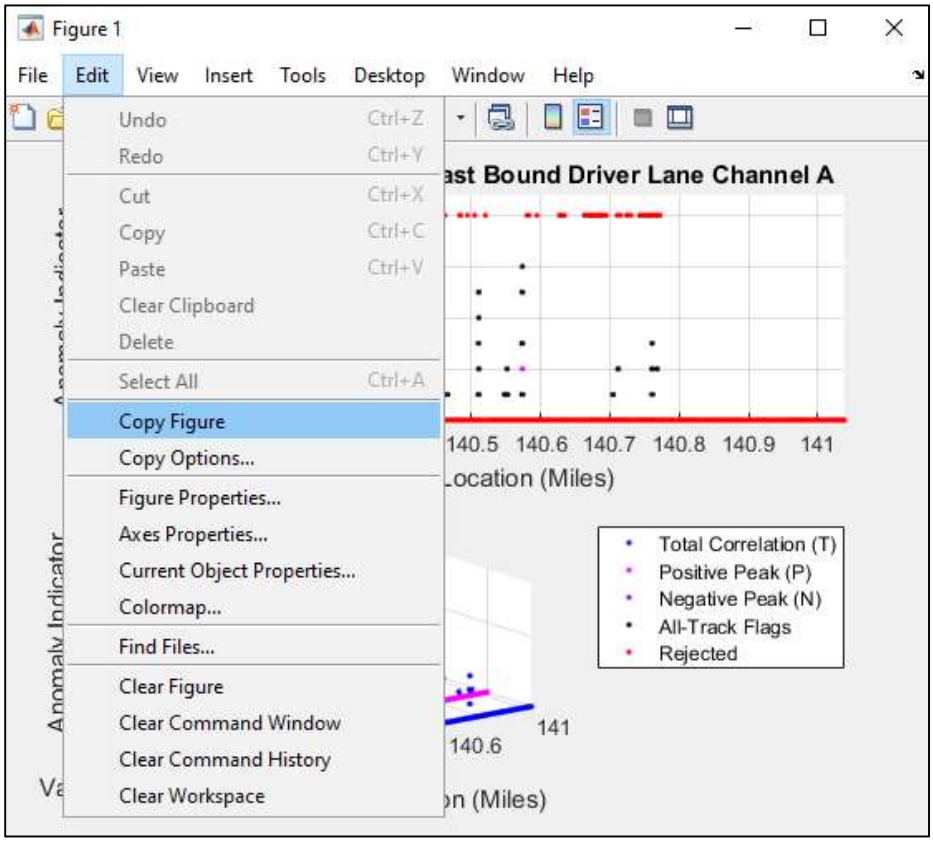

Fig. 11. The "Figure 1" window can be exported to the clipboard and pasted into another application by simply selecting Edit->Copy Figure from the figure window's main menu.

5. Appendix: Description of the data-driven algorithm developed in the earlier research project

5.1. Part 1: Planning and information gathering

5.1.1. Underlying assumptions and the rationale for the ability to perform anomaly detection

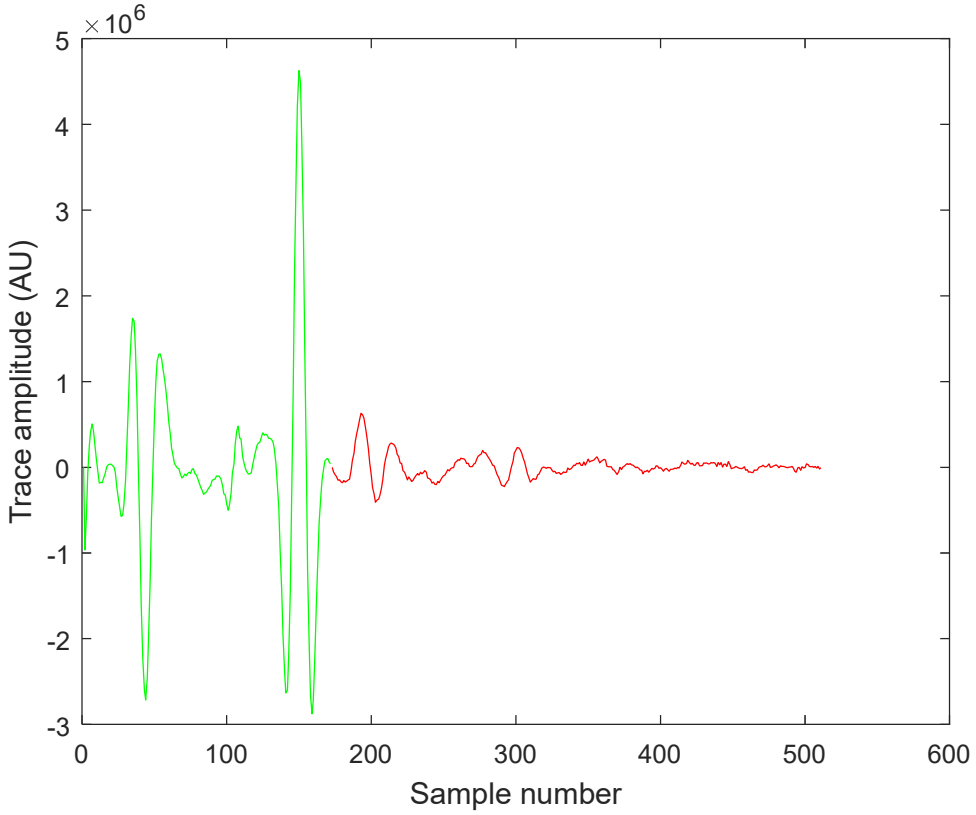

Fig. A.1. Representative time-domain trace of a GPR return. The green portion of the curve represents the surface (asphalt) reflection, and the red portion represents reflections from subsurface layers. Sample numbers can be identified with either time to a particular reflection point or depth beneath the surface.

Figure A.1 below shows a typical time-domain trace of the electromagnetic-pulse return from a GPR system. The green portion of the curve represents the surface (asphalt) reflection, and the red portion represents reflections from subsurface layers. The two peaks in the green portion represent the reflection off the surface of the asphalt (the first green positive peak from the right) and the direct-coupled wave (the first negative peak from the left), which represents the portion of the transmitted pulse that is directly detected by the antenna before reaching the surface of the pavement. All the analysis performed in this effort ignores the pre-surface reflections, since we assume that they are of little use to subsurface anomaly detection.

Large negative peaks from the subsurface portion of the trace can generally be indicative of distress layers since they typically represent the transition from a solid layer (high index of refraction, or wave impedance) to air (unity index of refraction) [1]. Similarly, a positive peak can indicate

presence of water. In addition, sudden changes in the location of peaks can also be indicative of layer a disturbance in the pavement layers, as in a sudden change in its depth, which could represent the collapse of layers. These and other changes in trace attributes are often looked for when a visual inspection is performed on GPR data for the identification of subsurface pavement distress.

The key assumption that the proposed algorithm is built upon is that *any trace along a sound segment of pavement maintains continuity in its attributes*. Such attributes include the number of major positive and negative peaks, the locations of peaks, and regularity in the shape of the peaks. In particular, the sudden appearance of a positive or negative peak could be indicative of distress. The same can be assumed about the disappearance of peaks. More generally, a sudden change in the shape of the trace can also be indicative of distress.

With the above assumptions in mind, the *key idea of the proposed anomaly detection algorithm is to detect discontinuities in the attributes of the traces from scan to scan*. To do so, we will need to have reliable metrics for the attributes of the traces, such as peak presence and changes in the shape of the trace from scan to scan, produce tracks of these metrics as successive scans are processed, and finally examine these tracks of the metrics to detect discontinuities in their values. The detection of discontinuities will be based upon statistical analysis of the metric tracks. Finally, a detection of a discontinuity in a track will constitute the detection of an anomalous scan.

5.1.2 Overview of the proposed anomaly detection algorithm

The detection algorithm, which is detailed in the next section, consists of two main steps:

Step 1: Process the longitudinal (along track) data, namely series of time-domain traces as the vehicle moves forward, and compute five metrics for each scan. As described above, the idea is to have tracks of the changes in the attributes of the traces, as captured by the five metrics. Once the entire data set for a pavement segment is processed, the algorithm produces five tracks: each track is an array, indexed by the mile post (or scan number). For each track, the amplitude of each metric at a particular scan, represents the strength of a feature that represents anomaly. The tracks are grouped into two categories based on the metrics they use. Each track uses one metric only. An overview of the types of metrics is described below.

Uniformity (correlation) metrics: These are based on change in the shape of the trace from scan to scan. There are three of these tracks. Scan-to-scan uniformity in the trace is measured by means of computing the maximum cross correlation between successive scans.

Peak-detection metrics: Here, we utilize metrics that respond to the sudden appearance of a positive or negative peak in the traces from scan to scan. There are two of these tracks. Change in the presence of peaks is measured by correlating a reference pulse (obtained from a metal-plate measurement) with each trace. The process also identifies the depth at which the strongest emergences of positive and negative peaks occur.

Step 2: Examine the scan-by-scan evolution of each one of the five tracks, and it identify scans where substantial change has occurred. The identification is based on selecting the highest values (or change therein) of the metrics at each scan, based on a statistical comparison test subject to a prescribed threshold. The threshold is set manually for each metric.

5.2. Part 2: Development of Algorithm

In this section we describe the details of the algorithm, including the pre-processing of GPR data. The method for extracting the raw data from the RADAN visualization software is described in the Appendix.

5.2.1. GPR data pre-processing

Before the GPR data is applied to the algorithm (described in the next subsection), certain pre-processing and signal conditioning is required.

5.2.2.1. Noise removal

As seen from Fig. A.1, the data includes temporal noise. To improve the performance of the algorithm, the noise must be reduced. In this effort we have used a simple lowpass filter. The filter is assumed to be constant (unity) in its passband and zero outside the passband. Figure A.2 shows the spectrum of a noisy trace (similar to the graph shown in Fig. (A.1)), represented by its 512-point discrete Fourier transform (DFT). To identify the cutoff frequency for the lowpass filter, we eyeball the spectrum and look for the region of high frequencies where the spectrum becomes more-or-less flat, which is typical noise characteristic. Two examples of such cutoff frequencies are shown in Fig. A.2, representing relaxed and aggressive noise removal. Figure A.3 shows the effect of aggressive lowpass filtering on the trace. The figure shows the reduction in fluctuations as a result of filtering. Other filtering techniques can also be used if additional noise removal is desired.

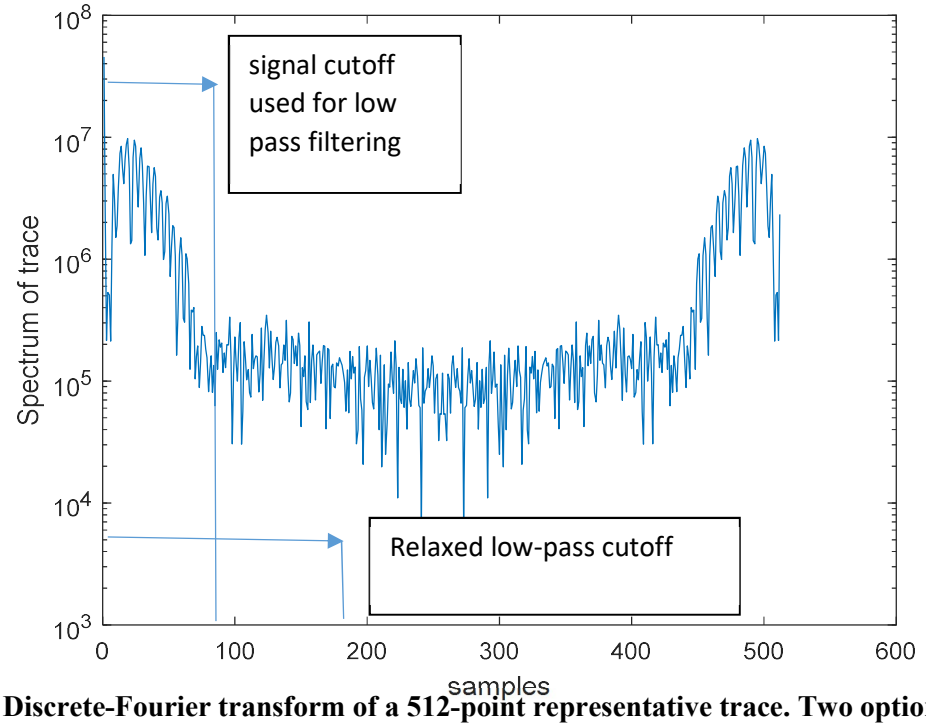

Fig. A.2. Discrete-Fourier transform of a 512-point representative trace. Two options for the selection of the cutoff frequency are shown, representing relaxed and aggressive filtering,

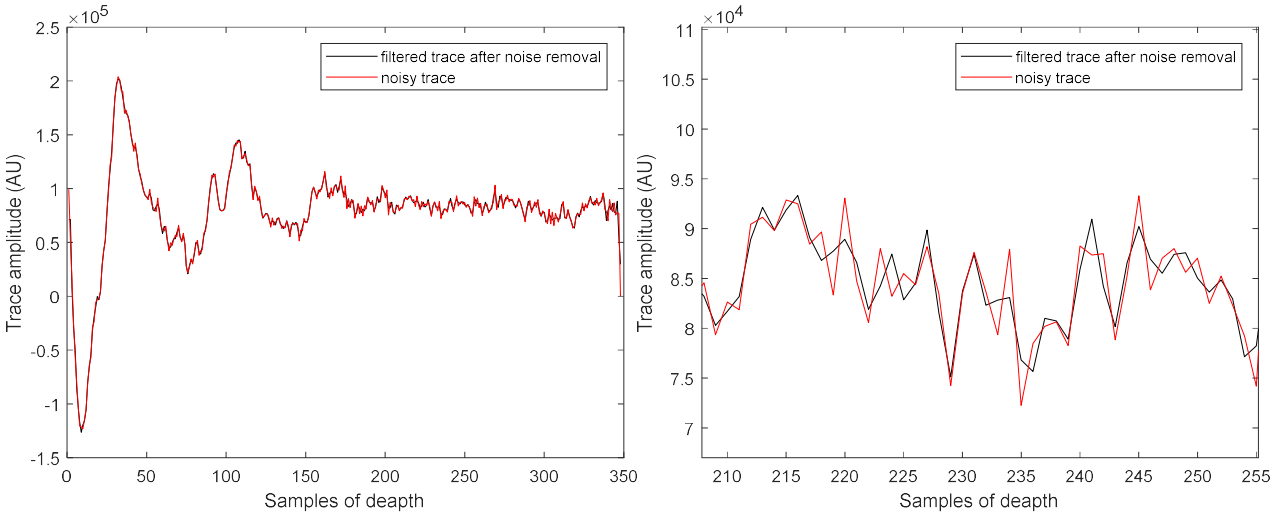

Fig. A.3. Left: Effect of aggressive lowpass filtering of the trace shown in Fig. A.1. Red and black curves represent the unfiltered and filtered traces, respectively. Right: A zoomed in traces.

5.2.2.2. Surface-reflection removal

Due to their large amplitudes relative to the amplitude of subsurface layers, the influence of the surface reflections is expected to be high on the metrics that will be used to measure changes in the trace attributes. For this reason, the surface reflections are removed entirely from the data prior to processing by means of truncation. This is done simply by eliminating the samples in the trace that are deemed to have resulted from the surface reflection and direct coupling. The initial approach has been to identify the surface reflection by inspection. For example, the first 150 samples are typically due to surface reflection and direct coupling. The red portion of the curve in Fig. A.1 shows the trace after the removal, resulting in a shorter array (due to sample deletions). This method can be easily automated by identifying the location of the largest peak, which can be estimated by performing a cross-correlation method between the trace and a reference pulse. It is worth mentioning that other methods do exist for surface reflection removal. These include the subtraction of amplitude and phase modified reference pulse from the trace [1]. The amplitude and phase adjustments are done to ensure that the subtraction has minimal effect on the smaller reflections from subsurface layers. The method is known to have short comings, however, since amplitude and phase matching can be very difficult to achieve accurately. Since our intent was to merely minimize the effect of the surface reflection on the metrics that represent change in the trace in the subsurface portion of the traces, the simple removal of the early portion of the trace by simple truncation (without altering the subsequent samples) was adopted with success.

5.2.2.3. Removal of scans infected with heavy RF interference

We have observed that in some occasions, certain portions of the scans are suffering from heavy RF interference, which is manifested in the form of highly fluctuating amplitudes at certain RF frequencies. The interference can be due to the presence of cellular-data towers, for example. These scans would almost certainly give rise to false anomaly detections, and we prefer to identify and remove them prior to performing anomaly detection. The method for identifying these scans is described next.

For any scan, we first compute the gradient of its trace (as a function of the samples), take the absolute value of the gradient values, and then sum up all the (positive) values. This procedure gives us a metric on how severe the fluctuating amplitudes are in a trace. We then compute the histogram of the values of the absolute value of the gradient, and identify and exclude the outliers based on a user-specified threshold. More specifically, we define an outlier as a scan the falls r times one standard deviation above the mean. We name r the *scan-rejection parameter*. A typical range of values for r is between 1.5 and 4. Clearly, the higher the value of r the bigger the number of the outlier is.

5.2.3. Mathematical definition of metric-based tracks that indicate distress-related anomalies in sequences of scans

As indicated in Section 5.1.2, there are five metrics that are grouped into two groups.

5.2.3.1. Three uniformity (correlation) tracks (Group 1)

These metrics measure changes in the cross-correlation in the traces from scan to scan. There are three metrics in this category. The cross correlation is measured using multiplication and integration.

Before we define the three metrics, we describe the cross-correlation operation mathematically. Let $f(n)$ and $g(n)$, n=0, …, N-1, be real-valued functions. The cross-correlation function between the f and g is defined as

$$C(f, g; m) = \sum_{k=0}^{N-1} f(k)g(k+m), \quad (1)$$

where m is a positive or negative integer shift variable. It is assumed that the values of $f(n)$ and $g(n)$ outside the range n=0, …, N-1 are zero; hence, $C(f, g; m)$ is generally nonzero for m = -N+1, …, 0, …, N-1. Next, we define the peak correlation, $r(f, g)$, between the functions $f(n)$ and $g(n)$ as

$$r(f,g) = \max_{-N+1 \le m \le N-1} C(f,g;m). \quad (2)$$

Note that for any function g, $r(g, g) = C(g, g; 0)$, which simply states that the peak value of the correlation function is achieved at zero shift. Moreover, if $g(n) = f(n-q)$, namely a shifted version of another function $f(n)$, then a simple calculation shows that $r(f, g) = C(f, g; q)$, which simply means that the maximum correlation is achieved when the two functions are fully lined up, i.e., when the shift q is fully compensated for in the function g.

Next, to have a universal metric that represents similarity between two functions f and g, we must normalize $r(f, g)$ so that the metric is invariant under scaling of the function f. To do so, we recall Schwarz inequality, which states $|r(f, g)| \le \sqrt{r(f, f)\, r(g, g)}$, and define the normalized metric, $\rho(f, g)$, as follows:

$$\rho(f, g) = r(f, g) / r(f, f). \quad (3)$$

Note that unlike its variant, $r(f, g)/\sqrt{r(f, f)\, r(g, g)}$, which is guaranteed to be between -1 and 1 by Schwarz inequality, the metric $\rho(f, g)$ is not necessarily between -1 and 1, and it can increase if the function g is scaled. This is a desirable feature for pavement anomaly detection because we want to be able to flag a trace that is a scaled version of the previously scanned trace.

We now define the three uniformity (correlation) metrics. Let $f_i(n)$ and $f_{i+1}(n)$, represent two consecutive processed traces, namely after lowpass filtering and surface-reflection removal, associated with the ith and i+1st scans. The *raw uniformity (correlation) metric* at the ith scan is defined as

$$\rho_1(f_i) = \rho(f_i, f_{i+1}). \quad (4)$$

Note that if $f_i = f_{i+1}$ then $\rho_1(f_i) = 1$, and if $f_i = -f_{i+1}$ then $\rho_1(f_i) = -1$. Generally, $\rho_1(f_i)$ can be above 1 or below -1. However, for the type of data encountered in this project, $\rho_1(f_i)$ will fluctuate around unity (often times, slightly), as shown in the next paragraph when we introduce the concept of a

metric track. This is because changes in the traces are small from scan to scan. An above-unity value can indicate, for example, the accentuation of some peaks from one trace to the next; and similarly, below-unity values may indicate the attenuation of some peaks.

Next, a track, using the metric ρ_1, can be defined as follows:

$$T_{u,1}(f_0, \ldots, f_{N-1}) = [\rho_1(f_0), \ldots, \rho_1(f_{N-1})], \quad (5)$$

which we name the *raw uniformity (correlation) track*. It simply represents the progression of the metric as more and more scans are processed. An example of the track $T_{u,1}$ is shown in Fig. A.4.

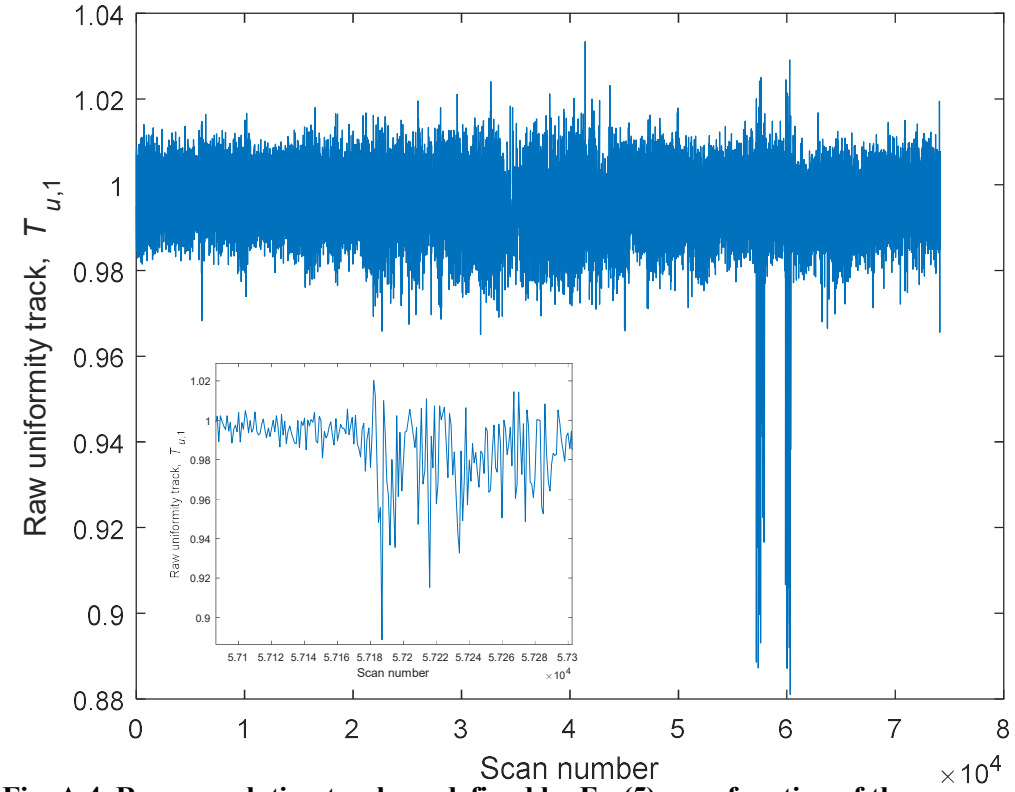

Fig. A.4. Raw correlation track, as defined by Eq.(5), as a function of the scan number. Relatively large deviations from 1 are indicative of change in the shape of traces corresponding to successive scan at that location. The inset is a zoomed-in version of a portion of the track, showing the scan to scan fluctuations.

We now describe the second uniformity (correlation) track. From our experience it appears that the upper and lower envelopes of the track $T_{u,1}$ can provide refined information about trace variability from scan to scan, which can be used in the anomaly detection stage. An envelope tends to slightly spread the isolated peaks, thereby strengthening their presence. It also allows us to examine the effects of above-unity deviations and below-unity deviations in the raw uniformity (correlation) track separately. There are many techniques for envelope detection, and in this effort

we have found the ("*N*-point") spline technique effective for our application. The method basically uses a spline fit over local maxima separated by at least *M* points; we took *M* as 10 in our calculations. If we denote the upper and lower envelopes of the track $T_{u,1}$ as $UT_{u,1}$ and $LT_{u,1}$, respectively, then we can define the second pair of uniformity (correlation) tracks as

$$T_{u,2,\text{upper}} = UT_{u,2} \text{ and } T_{u,2,\text{lower}} = LT_{u,1}, \quad (6)$$

which we call the *upper and lower uniformity (correlation) tracks*, respectively. For convenience, we refer to these two tracks collectively as an *envelope uniformity (correlation) track*. Figure A.5

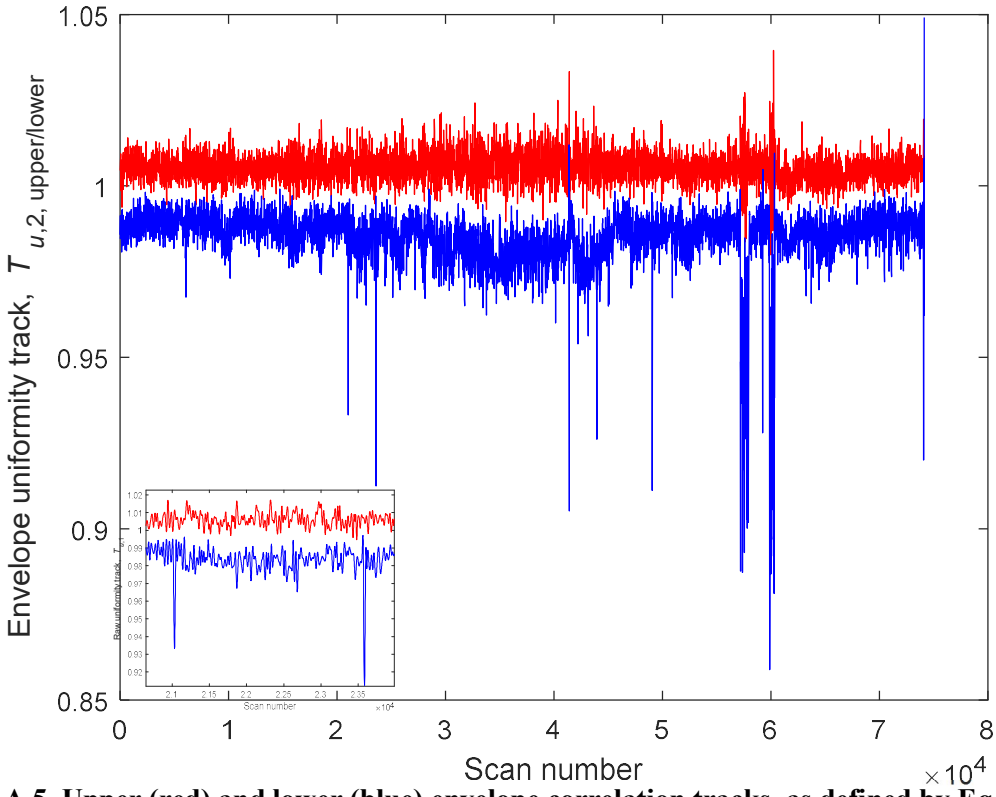

Fig. A.5. Upper (red) and lower (blue) envelope correlation tracks, as defined by Eq.(6), as a function of the scan number. The inset is a zoomed-in of a portion of the track, showing the scan to scan fluctuations.

shows representative lower and upper envelope tracks.

The third and last uniformity (correlation) track is the discrete derivative of the raw uniformity (correlation) track, $T_{u,1}$, denoted by $T_{u,3}$; namely,

$$T_{u,3} = (T_{u,1})_1 - T_{u,1}\ , \quad (7)$$

where the notation $(T_{u,1})_1$ means the vector $T_{u,1}$ is shifted to the right by a single time step. We call $T_{u,3}$ the *derivative uniformity* (correlation) *track*. Figure A.6 shows a representative derivative uniformity (correlation) track.

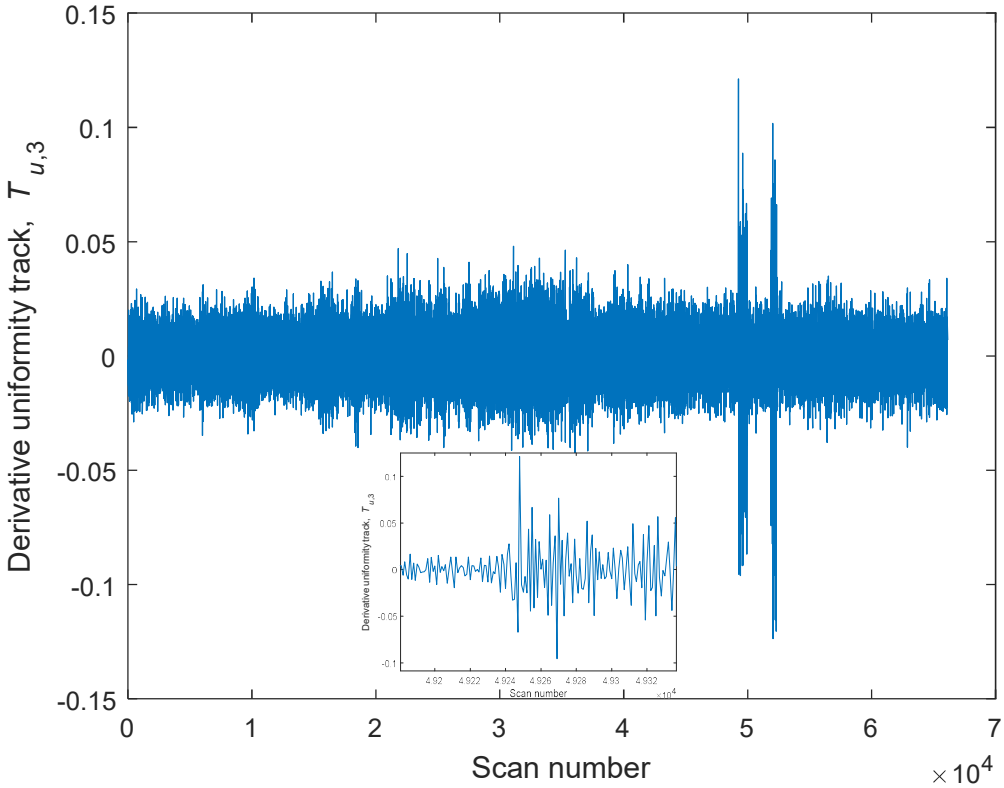

Fig. A.6. Derivative correlation track, as defined by Eq.(7), as a function of the scan number. The inset is a zoomed-in of a portion of the track, showing the scan to scan fluctuations.

5.2.3.2. Two tracks based on positive and negative peaks (Group 2)

The goal here is to look for the sudden presence of a positive peak or a negative peak in the trace, where a "peak" is defined by the shape of a reference pulse, e.g., the pulse from a metal plate reflection. To do so, we divide each trace into bins of length 20 samples, as shown in the middle figure in Fig. A.7. We then look for the presence of a negative (positive) peak in each bin, as depicted in the left figure. The detection of a negative (positive) peak is accomplished by cross-correlating a reference pulse (left), obtained from the metal plate reflection, say, with each trace (see middle figure). The peak of the cross-correlation function over each bin is indicative of the presence of a negative peak in that bin (see right figure in Fig. A.7). We announce a "negative-peak detection" if any of the local maxima situation changes drastically from scan to scan, as described mathematically next.

Recall the metric $\rho(f, g) = r(f, g) / r(f, f)$, which was defined in (3) in Section 2.3.1. Here we take the function f as a reference negative pulse, as shown in the left figure in Fig. A.7, and take

the function g as the ith trace, namely $g = f_i$. Next, for the kth bin we define the local maximum negative-peak indicator as

$$L_{i,k} = - \max_m \rho(f, f_i), \tag{8}$$

where the maximum is taken over all samples m inside the ith bin. Next, we define the negative-peak indicator in the ith trace as

$$L_i = \min_i L_{i,k} , \tag{9}$$

where the maximum is over all bins. Next, we define the *negative-peak track* as

$$T_{NP}(f_0 , \ldots, f_{N-1}) = [L_0, \ldots, L_{N-1}]. \tag{10}$$

In a similar fashion, we can define the positive-peak track. To this end, we define

$$U_{i,k} = \max_m \rho(- f, f_i), \tag{11}$$

where, as before, the maximum is taken over all samples m inside the ith bin, and define the positive-peak indicator in the ith trace as

$$U_i = \max_i U_{i,k} , \tag{12}$$

where the maximum is over all bins.

Finally, we define the *positive-peak track* as

$$T_{PP}(f_0 , \ldots, f_{N-1}) = [U_0, \ldots, U_{N-1}]. \tag{13}$$

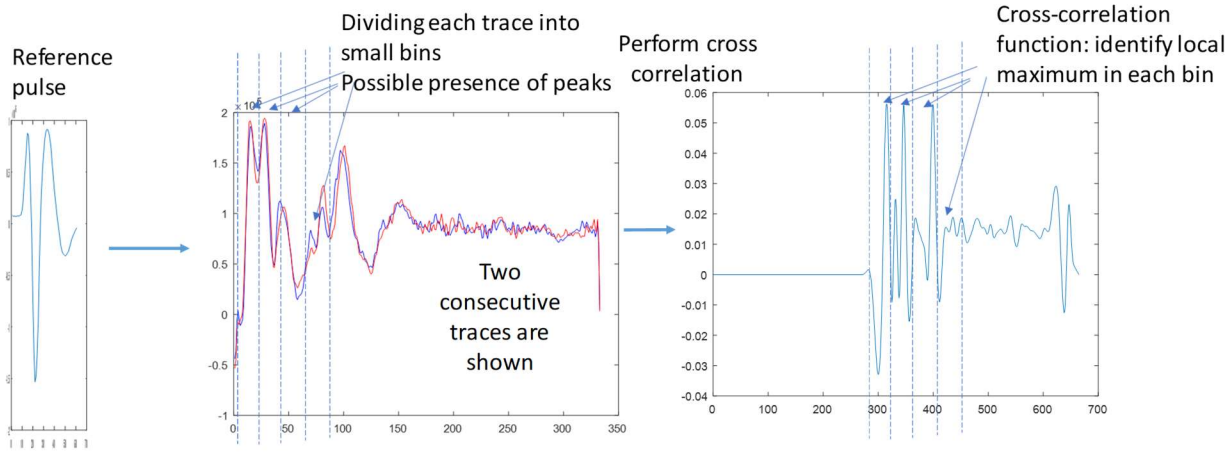

Fig. A.7. Process for identification of a negative-peak presence in a trace using a reference pulse from a metal-plate reflection (left). Each trace (middle) is subdivided into small bins and further cross-correlated with the reference pulse. The maximum of the cross-correlation function (right) in each bin is identified with the likelihood of the presence or absence of a pulse in that bin. We say a peak emerged in a trace if one of the local maxima experiences a statistically significant increase compared to the previous trace.

Figure A.8 shows examples of the tracks T_{NP} and T_{PP}.

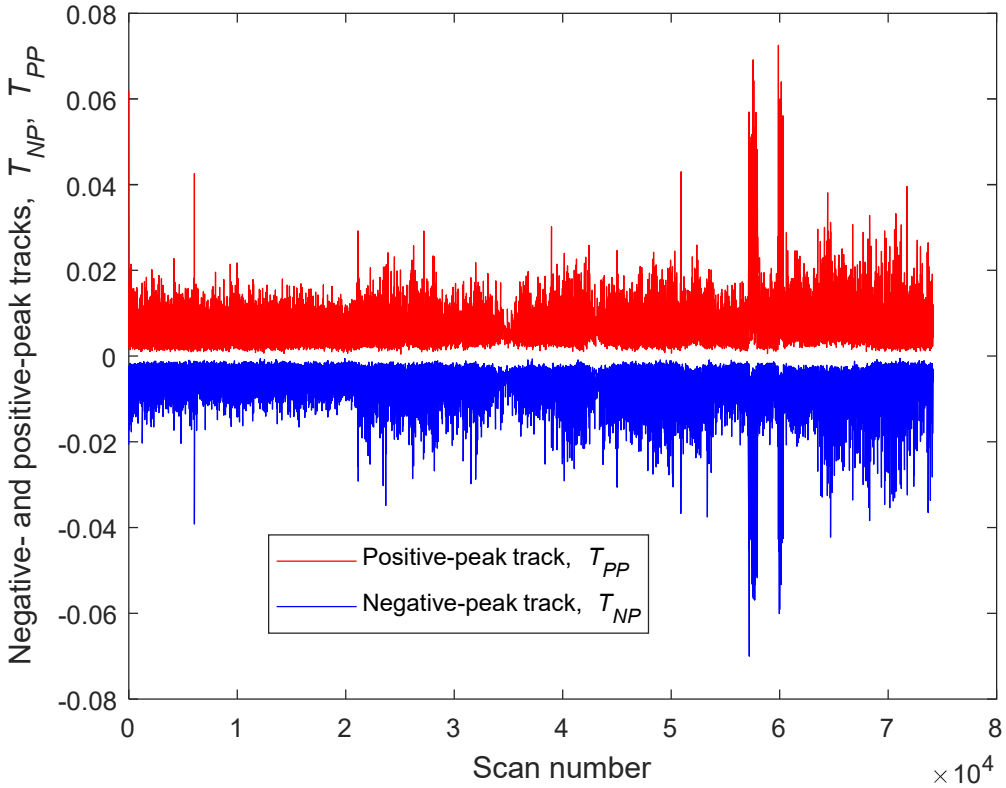

Fig.A. 8. Negative- and positive-peak tracks, as defined in Eqs. (10) and (13), as a function of the scan number.

To summarize, in this subsection we have developed five tracks that will be used in the next subsection to identify scans that potentially have anomalous traces. The tracks are (i) the raw uniformity (correlation) track, $T_{u,1}$; (ii) the pair of upper- and lower-envelope tracks, $T_{u,2,\text{ upper}}$ and $T_{u,2,\text{ lower}}$; (iii) the derivative uniformity (correlation) track, $T_{u,3}$; (iv) the negative peak track, T_{NP}; and (v) the positive-peak track T_{PP}. Each one of these five tracks is a function of the scan number.

5.2.4. Statistical anomaly detection based upon metric-based tracks

The main idea of our anomaly detection process of scans is to isolate scans that have large metrics (or change) associated with them. This is done for every track independently, and then the isolated scans from various tracks are consolidated. To perform the separation of the scans, we follow a simple statistical method to identify outliers. For each track, we compute the probability mass function (pmf), extracted from a histogram of all the track values for a given track, and use a prescribed threshold probability to identify the highest and lowest percentiles. We use the symbol p for a pmf. For example, $p_{u,1}$ is the pmf associated with the track $T_{u,1}$, and so on. In principle, the thresholds can be different for various tracks; however, in the calculations we use a single threshold

for simplicity. (In fact, we have found no good reason to assign different threshold values for different tracks.) The details are described below.

For the raw uniformity (correlation) track pmf, $p_{u,1}$, and a threshold probability η, we first calculate the critical raw-uniformity (correlation) metric values, $a_{u,1}$ and $b_{u,1}$ so that the probability that a uniformity (correlation) track value exceeds $a_{u,1}$ or falls below $b_{u,1}$ is η. Then, we define the raw-track anomalies as the set of scans whose metrics either exceed $a_{u,1}$ or fall below $b_{u,1}$. The anomalies based on the derivative track are performed in the exact same manner.

For the envelope uniformity (correlation) track, the process is slightly different. For the upper-envelope pmf $p_{u,2,\text{upper}}$, we calculate the critical value $a_{2,u,\text{upper}}$ (typically a positive number) so that the probability that a upper-envelope track value exceeds $a_{2,u,\text{upper}}$ is η. On the other hand, for the lower-envelope pmf $p_{2,u,\text{lower}}$, we calculate the critical value $b_{2,u,\text{lower}}$ so that the probability that a lower-envelope track value falls below $b_{2,u,\text{lower}}$ is η. Then, we define the raw-track anomalies as the set of scans whose upper-envelope track values exceeds $a_{2,u,\text{upper}}$ or those whose lower-envelope track values falls below $b_{2,u,\text{lower}}$. The anomalies based on the positive and negative-peak tracks are performed in the exact same manner.

During the separability phase of the project, we had performed an extensive study on the choice of the probability threshold, η, using as a guide the visual inspection of the traces, as provided by RADAN, as well as comparisons with cored areas. The results that are shown below use the threshold values that we deemed reasonable. A similar examination was applied to the scan-rejection parameter, r. It is interesting to note that the parameter r can be chosen to reject certain anomalous structures, such as bridges, are identified and consequently removed. In this work we removed a good portion of bridge scans because they would uselessly deplete the quota for anomalies, as dictated by the probability threshold η.

Figure A.9 shows an example on the detection process using the raw uniformity (correlation) metric. The blue and green dashed lines represent the level thresholds $a_{u,1}$ and $b_{u,1}$, respectively, computed according to a threshold probability of η=0.998. Any crossings by $T_{u,1}$ above the blue line or below the green line is flagged as an anomaly.

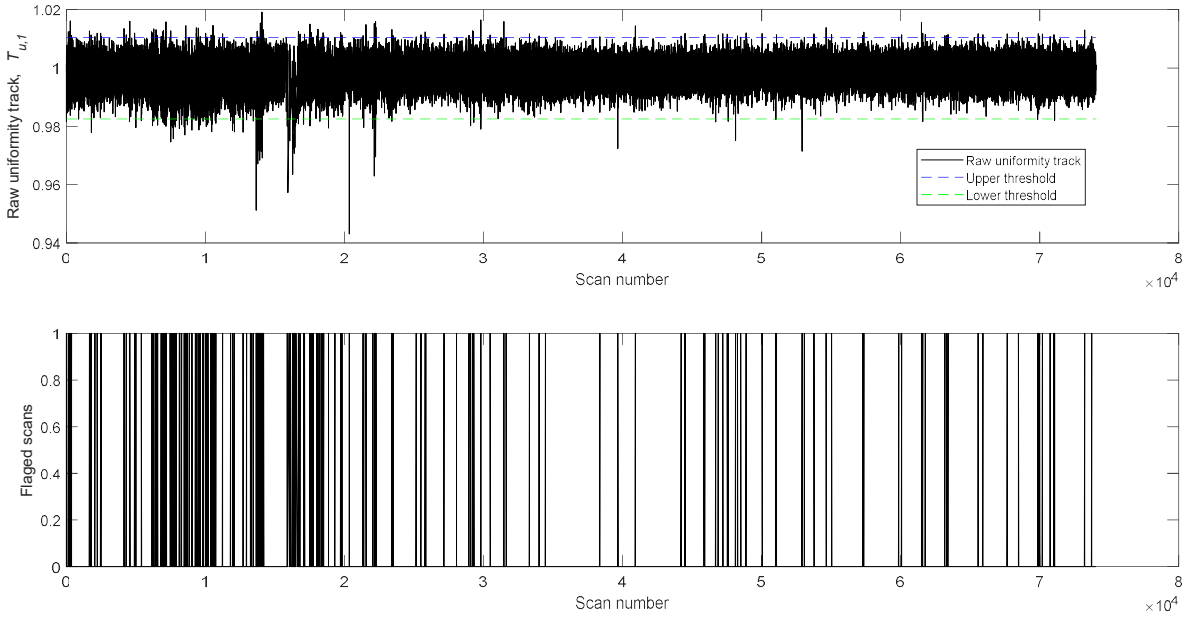

Fig. A.9. An example on the detection process using the raw correlation metric. The blue and green dashed lines represent the level thresholds $a_{u,1}$ and $b_{u,1}$, respectively, according to a threshold probability of η=0.998. Crossings above the blue line and below the green line are flagged.

Once the track for anomalies were generated, we applied a moving-average filter to each flag sequence from each track. The width of the filter was chosen as 10 scans. In effect, this produced a moving count of the flags in for each metric over a window of 10 scans. This process helps the visualization of the flags, and it helps put more emphasis on the occurrences of flags in a small vicinity.

Figure A.10 shows the results of the anomaly detection. This is an example of GPR data collected from the northbound I-40 freeway in Albuquerque, NM, in August 2017. The scan-rejection parameter was taken. The scan-rejection parameter is selected as r = 2.5 (described in Section 2.2.3), and the detection threshold probability is η=0.998. In total, 2,213 scans have been deleted (from a total of 74,032 scans), and a total of 754 scans are identified as anomalies. It is worth

mentioning that the sum of the detections by all tracks is 1,421; hence, many of the detections are in duplicate. In the plot designated as "total," the results of the moving average for the flags of

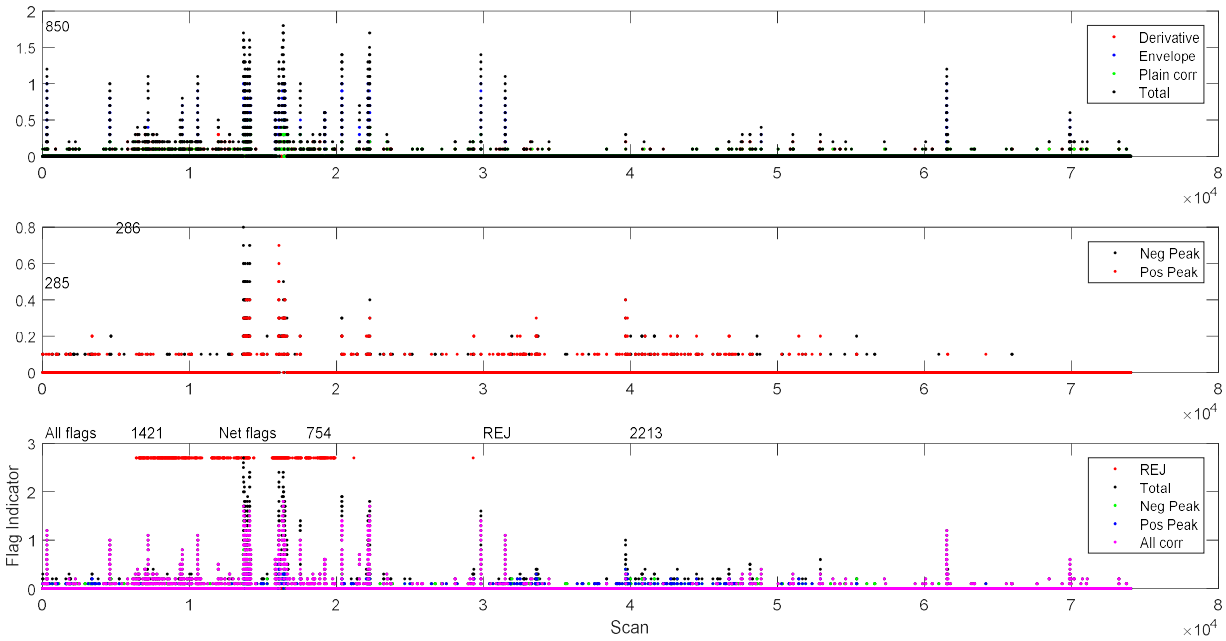

Fig. A.10. Anomaly detection based on the uniformity (correlation) and peak-detection tracks. The top figure shows the detections based on the three uniformity (correlation) metrics. The middle figure shows the detections based on the two peak-detection metrics. The bottom figure shows the consolidated detections. In these calculations, the scan-rejection parameter is selected as r = 2.5 (described in Section 5.2.2.3), and the detection threshold probability is η=0.998. In total 2,213 scans have been deleted (from a total of 74,032 scans), as shown by the red dots in the bottom figure, and a total of 754 scans are identified as anomalies.

each track have been added. For example, a value of 3 at a particular scan number indicates that three tracks produced flags in a window of width 10 preceding the scan number, and so on.

Figure A.11 shows the effect of choosing a tighter threshold probability, η=0.9999, instead of η=0.998. The number of total flags in this case is reduced from 754 to 392.

Finally, during the project we have also refined the code to display the flags as a function of the milepost instead of the scan number. This was done by a simple conversion, using the total length of the scanned pavement, the starting milepost, as well as the scan frequency. The milepost results are snot shown in this report for brevity.

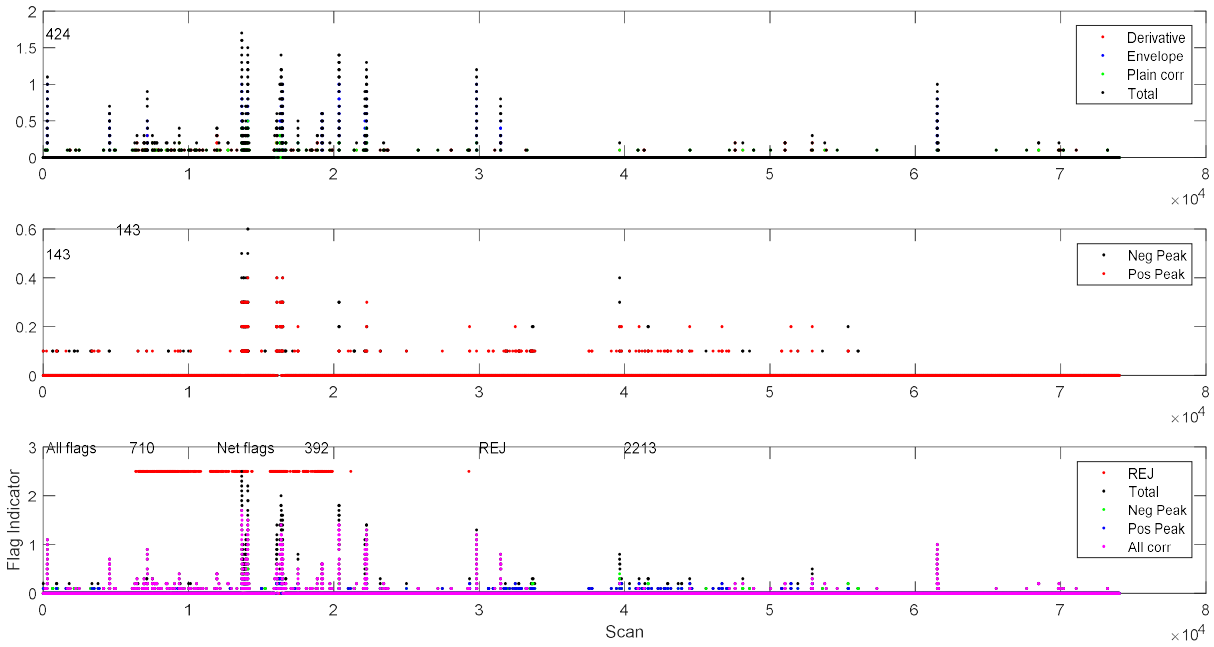

Fig. A.11. Same as the plots in Fig. A.10 except that the detection threshold probability is η=0.999. A reduced total of 392 scans are identified as anomalies.

5.3. Part 3 Testing/Verifying

The following pavements have been tested. The detailed reports are in the research project.

- Core locations vs. core condition
- NM 264 cores vs. algorithm
- I-40 District 6 cores vs algorithm
- I-40 D3 cores vs. algorithm
- US 491 cores vs. algorithm
- NM 14 cores vs. algorithm
- NM 6 metal plate experiment vs. algorithm

Bibliography

[1] Lahouar, S., "Development of Data Analysis Algorithms for Interpretation of Ground Penetrating Radar Data," Ph.D. Dissertation. Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 2003.

[2] Rmeili, E., and T. Scullion, "Detecting Stripping in Asphalt Concrete Layers using Ground Penetrating Radar", Transportation Research Record 1568, pp. 165-174, 1997.

[3] Saarenketo, T. and P. Roimela, "Ground Penetrating Radar Technique in Asphalt Pavement Density Quality Control", Proceedings of the Seventh International Conference on Ground Penetrating Radar, Vol. 2, pp. 461-466, Lawrence KS, May 27-30, 1998.

[4] Saarenketo, T., "Using Ground Penetrating Radar and Dielectric Probe Measurements in Pavement Density Quality Control", Transportation Research Record 1997, pp. 34-41, 1997.

New Mexico Department of Transportation
RESEARCH BUREAU
7500B Pan American Freeway NE
PO Box 94690
Albuquerque, NM 87199-4690