Development of a Real-Time Transverse Pavement Profile Measurement System
Research Report 0-1782-1

THE UNIVERSITY OF TEXAS AT ARLINGTON
TRANSPORTATION INSTRUMENTATION LABORATORY

RESEARCH REPORT 0-1782-1

Roger S. Walker, Ph.D., P.E
Stephen A. Underwood, Ph.D., P.E

Date: February 2003

Development of a Real-Time Transverse Pavement Profile Measurement System
Research Report 0-1782-1

THE UNIVERSITY OF TEXAS AT ARLINGTON
TRANSPORTATION INSTRUMENTATION LABORATORY

RESEARCH REPORT 0-1782-1

Roger S. Walker, Ph.D., P.E
Stephen A. Underwood, Ph.D., P.E

Date: February 2003

Notice **–** The United States Government and the State of Texas do not endorse products or manufacturers. Trade or manufacturers’ names appear solely because they are considered essential to the object of the report.

Technical Report Documentation Page

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
0-1782-1		

4. Title and Subtitle	5. Report Date
Development of a Real-Time Transverse Pavement Profile Measurement System	February 2003
	6. Performing Organization Code
7. Author(s) Roger S. Walker, PhD, and Stephen A. Underwood, PhD	8. Performing Organization Report No. Research Report 0-1782-1
9. Performing Organization Name and Address The University of Texas at Arlington 416 Yates Street Arlington, TX 76019	10. Work Unit No. (TRAIS) 11. Contract or Grant No. Project No. 0-1782
12. Sponsoring Agency Name and Address Texas Department of Transportation Research and Technology Transfer Office P.O. Box 5080 Austin, TX 78763-5080	13. Type of Report and Period Covered Final – 14. Sponsoring Agency Code

15. Supplementary Notes

Research performed in cooperation with the Texas Department of Transportation.

16. Abstract

The Texas Department of Transportation (TxDOT) has been using a five-sensor rut bar system, using ultrasonic sensor technology, to automatically collect estimates of pavement rutting for PMIS purposes. TxDOT also is presently using a laser-based rut measurement system on one of the TxDOT Profilers. Laser and acoustic base systems are spot-specific, with the sensors located at fixed distances across the eight foot rut bar. These techniques limit the rut measurements to fixed points along an imaginary stringline.

This project has investigated the use of scanning laser technology for measuring transverse profile and rut information. A functional system has been developed to scan the full width of the paving lane and to report and store the rut condition of each wheel path. An algorithm has also been developed to measure and report the deepest rut per scan. The system has been tested at several sites and is ready for limited implementation.

17. Key Word	18. Distribution Statement
Scanning Laser, Rut Measurements, TxDOT Profilers	No Restrictions

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	93	

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

DISCLAIMER(S)

The contents of this report reflect the views of the author(s), who is (are) responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official view or policies of the Texas Department of Transportation. This report does not constitute a standard, specification, or regulation.

There was no invention or discovery conceived or first actually reduced to practice in the course of or under this contract, including any art, method, process, machine, manufacture, design or composition of matter, or any new useful improvement thereof, or any variety of plant, which is or may be patentable under patent laws of the United States of America or any foreign country.

ACKNOWLEDGEMENTS

The authors would like first to acknowledge Carl Bertrand of the Texas Department of Transportation. Without his support, this research project would not have been possible. Acknowledgements are also due the students and staff personnel, in particular Eric Becker at the Transportation Instrumentation Laboratory facility at the University of Texas at Arlington. Finally, it should also be noted that Dr. Stephen A. Underwood, co-author of this report, passed away in October of 2003. Dr. Underwood was a major contributor to the success of the project.

Table of Contents

List of Figures

List of Tables

Chapter 1

1.1 Introduction

This report discusses a project between the Texas Department of Transportation and The University of Texas at Arlington, titled 'Development of a Real-Time Transverse Pavement Profile Measurement System', Research Project 1782. The Texas Department of Transportation (TxDOT) has been using a five-sensor rut bar system, implemented with ultrasonic sensor technology, to automatically collect estimates of pavement rutting for PMIS purposes. A number of problems have occurred while using the acoustic sensors for this purpose. The project was initiated to investigate the possibility of using scanning laser technology for measurement of rut that would alleviate some of the problems of the acoustic sensor systems.

Research project personnel, after reviewing various laser systems, found two scanning laser systems that might be suitable for measuring rut. The first laser system, sold by Phoenix Scientific, was much too costly and the availability questionable at the time of the project. The second laser system was a low cost scanning laser system manufactured by Acuity Inc. During the course of the project, two Acuity systems were acquired. The project has investigated the use of an Acuity system, and then developed interface hardware and software procedures so that TxDOT could begin implementing these systems for rut measurements.

During the project a functional system has been developed to scan the full width of the paving lane and to report and store the rut condition of each wheel path. In Figure 1.1 the comparisons between readings from the scanning laser system and the five rut measurement method is illustrated. Instead of five sensors, the scanning laser will provide a much larger set of measurements (typically greater than 200, depending on the speed of the vehicle and scanner rotation).

Development of the functional system has not been without problems. These unexpected problems resulted in a one-year extension of the original time that was thought sufficient for finding, obtaining and developing a procedure for using the scanning laser system. The main problem has been in noise, the errors in the signal that occurs as the laser beam is swept across the pavement surface. The laser sensor, in a stationary configuration, met the technical specifications as indicated by the manufacturer. However, when scanning the laser across a pavement or other non-smooth surface, noise spikes much greater than the profile signal would occur. Thus, much of the project research effort was focused on developing a means to distinguish the signal from the noise, and to address a means by which measurement methods could be done at highway speeds. Additionally, when ordering the scanning system, Acuity only provides the algorithms for the laser measurement calibration. Scanning software and coordinates adjustments had to be developed.

A working system for rut measurements has been developed and provided to TxDOT for implementation. Initially, it was also planned to investigate the use of the transverse profile data, in other applications, such as estimates of overlay quantities, estimates of pavement cross fall, etc. These additional investigations, however, were not possible because of the efforts required in developing a rut measurement system using the Acuity laser scanning equipment. The rut reporting uses the string-line procedure for measuring and reporting rut for both wheel paths. The method provides the average and deepest rut for user selectable intervals.

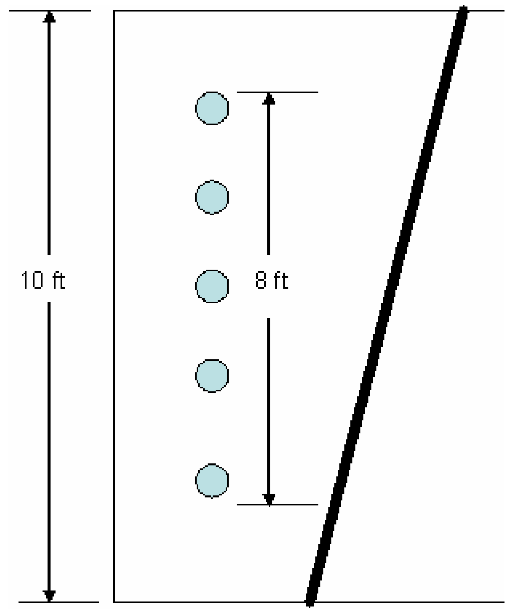


Figure 1.1 Sensor Footprints

1.2 Report Contents

The report contains six chapters including this chapter. The hardware characteristics of the Acuity Scanning Laser System along with the UTA developed interface circuitry are described in Chapter 2. Chapter 3 discusses the tests performed for checking or verifying the proper operation of the scanning laser system, and initial efforts in collecting and analyzing data. Chapter 4 and Chapter 5 then describe the programs developed for making rut measurements and the method of using these programs. Chapter 6 provides the project's summary and conclusions.

Chapter 2

Scanning Laser System Components

2.1 Introduction

This Chapter discusses the hardware characteristics of the Scanning Laser System. Several subsystems make up the scanning laser system. Three unique components make up the AccuRange subsystem. These components are the AccuRange AR4000-LIR Laser, AccuRange Line Scanner, and the AccuRange High Speed Interface. The next component of the scanning laser is the High Speed Interface Control. Finally, a host machine with a PC-104 or ISA bus, with a keyboard, and graphics display. Figure 2.1 illustrates these systems components.

Acuity Research Incorporated provided the AccuRange components. A detailed description is available in the manual, AccuRange 4000. The theory of operation of the AR4000 Laser is available at the Internet site http://www.acuityresearch.com. The high-speed interface control and host computer were assembled by The University of Texas at Arlington Transportation Instrumentation Laboratory.

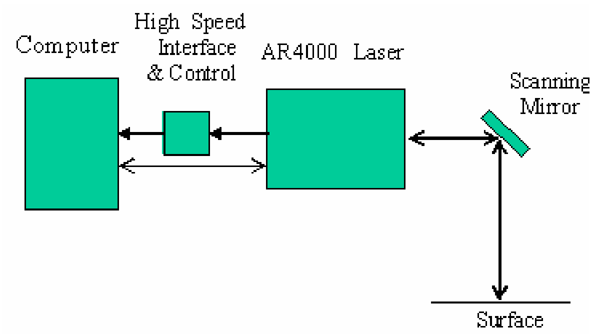


Figure 2.1 Scanning Laser System

2.2 Theory of Operation

The basic methods of measuring distances with a laser are:

- Triangulation Devices – provide the most accuracy but are limited in range and would be difficult to design and work with scanning mirrors.
- Time of Flight Distance Measurement – measures the time difference between transmitted and received laser pulse but is limited in accuracy by the accuracy of the measurement of very short time intervals and the rise time of laser pulses.
- Modulated Beam Systems –measures the phase shift (time delay) between a transmitted and received modulated laser beam but is limited by the accuracy of phase measurement and the distortion in the modulated signal by the reflecting surface.

The AccuRange AR4000-LIR Laser is a Modulated Beam type system, which uses a range-to frequency conversion method instead of phase measurement to determine distances. The system contains a 780 nm IR laser diode that is amplitude modulated by an oscillator. The reflected beam is shifted in phase by an amount that is proportional to the distance to the reflecting surface. This reflected signal is detected by a photodiode and returned to the modulating oscillator to form a resonant feedback circuit. Therefore, the frequency of oscillation of the laser beam is roughly inversely proportional to the distance to the reflecting surface (from 50 MHz at zero range to 4 MHz at 50 feet).

The AR4000-LIR Laser contains a microprocessor, which measures the frequency of the oscillation and controls the operation of the laser. The communication with the microprocessor is provided through an asynchronous serial port. The default parameters of the serial port are 9600-baud, no parity, 8 data bits, 1 stop bit. The protocol contains 22 different commands that can be sent to the AR4000-LIR, but the scanner program only uses four of these commands:

- Laser Power On: 'H'
- Laser Power Off: 'L'
- Set Sample Interval: 'S'<sample interval in μ seconds>
- Set Maximum Range: 'F'<maximum range in inches>

The AR4000-LIR transmits the calibrated range from the serial port if the sample rate is less than or equal to 770 samples per second. The unit also sends the uncalibrated range value as a pulse-width modulated signal and analog signals representing the received signal amplitude, sensor temperature, and ambient light in the AR4000-LIR laser via a cable with a DB-9M connector.

The measurement accuracy of the AR4000-LIR is affected by three internal factors and an unknown number of external factors:

- Detector Thermal Noise – increases with sample rate.
- Frequency Measurement Clock Error – increases with sample rate.
- Long Term Drift –varies with time.
- Frequency Distortion by the Target – this is the major source of error and is caused by the motion of the laser beam during the sampling interval.

Acuity Research documents the first three factors. The fourth factor is only present when the scanning mirror is in motion and increases with texture roughness of the reflecting target.

2.3 The Scanner Hardware

Two different AccuRange AR4000-LIR Lasers were used in this research effort. One unit used a standard 30 milli-watt laser diode and the other unit used a special-order 50 milli-watt laser diode. Both units used an optical filter with the photodiode detector to reduce the effect of solar reflection.

The AccuRange Line Scanner consists of a rotating scanning mirror, a DC motor, and an encoder with an index pulse. These components are mounted in a waterproof case along with the AR4000-LIR laser and a DC power supply for the AR4000-LIR laser. The motor operates at 2600 revolutions per minute with a 12VDC input. The encoder produces 2000 counts per revolution and an index pulse for each revolution. The relation of the index pulse to the mirror position is not mechanically calibrated but will be determined in the Scanner Program initialization procedure.

The AccuRange High Speed Interface (HSI) is used to digitize the raw uncalibrated values from the laser. The High Speed Interface digitizes four signals from the laser. The first signal is the raw range pulse width modulated signal. The pulse width represents the range of the reflected signal. The second signal is the analog intensity signal, which is the amplitude of the reflected signal. The next signal is the analog sensor temperature signal, which is the temperature of the laser photodiode detector. The last signal is the ambient light intensity. Software in the Scanner Program uses a calibration table that is provided with each individual AR4000-LIR/High Speed Interface combination by the vendor and a sensor temperature correction formula to calculate the calibrated range value. The individual AR4000-LIR and High Speed Interface components cannot be exchanged between other systems because of the calibration table values for a matched pair.

The High Speed Interface places the digitized values, along with other values, into an 8-bit FIFO (first-in-first-out buffer) that can be read with bus I/O read commands from the Base Address on an ISA or PC-104 bus. Each sample of data (one sample interval) generates the following data in the FIFO:

- Byte 0: Signal Amplitude (used in calibration table look-up)
- Byte 1: Ambient Light Value
- Byte 2: Sensor Temperature (used in range calibration)
- Byte 3: Bits 7-5: Bits 2-0 of raw range value
  - Bit 4 : Always 0
  - Bit 3 : Set to 1 if the FIFO overflows
  - Bit 2 : External Input 3
  - Bit 1 : External Input 2
  - Bit 0 : External Input 1
- Byte 4: Bits 10-3 of raw range value
- Byte 5: Bits 18-11 of raw range value
- Byte 6: Encoder 1 Value
- Byte 7: Encoder 2 Value (not used in the scanner system)

The High Speed Interface Control Board modifies the following in the FIFO data:

- External Input 1 for the scanning mirror index pulse from the encoder (default).
- External Input 3 for the “Start” signal supplied by TXDOT.
- Encoder 1 Value for either the scanning mirror encoder position (default) or the “Distance Pulse” simulated encoder value from the High Speed Interface Control Board that is produced from the “Distance” pulse supplied by TXDOT.

The status of the FIFO is accessed by reading from Base Address+1 for FIFO empty (Bit 0) and FIFO half full (Bit 1). Writing a 3 to Base Address+0 will reset the High Speed Interface processor and writing a 1 will reset the FIFO overflow flag (Bit 3 of Byte 3). If the variable speed option is selected on the High Speed Interface Control Board, the speed of the scanning mirror motor can be controlled by writing a value (0 to 63) to Base Address+2. Due to voltage losses in the circuit, however, full speed (2600 RPM @ 12 VDC) cannot be obtained using the variable speed option. The DC power for the scanning mirror motor is applied to the High Speed Interface board from the PC-104 bus or an external source.

The High Speed Interface comes as a standard 8-bit ISA (used with the 50 mw AR4000-LIR) or

PC-104 (used with the 50 mw AR4000-LIR) board and is mounted in the Scanner Program computer. The interface is jumpered for a Base Address of 300 hexadecimal and no interrupts. The AR4000 Laser is connected through a DB-9F connector, and the Line Scanner is connected through a DB-25F connector.

The Signal Interface Control provides options for the controlling of the scanning mirror motor and the data that is stored in the FIFO output of the High Speed Interface. It also contains circuits for the counting of the "Distance Pulses" supplied by TXDOT and the detection of reflective tape. The controller is the only hardware component, other than the Computer, which was not provided by the vendor (Acuity). The controller is mounted on a PC-104 board but only uses the +5VDC & +12VDC pins of the bus. The controller has been mounted on the PC-104 bus for convenience and could be mounted elsewhere. The DB25M cable from the AccuRange Line Scanner plugs into the High Speed Interface Control Board and a cable from the High Speed Interface Control Board connect to the DB25F connection on the AccuRange High Speed Interface Board.

The board has the following jumper options:

Jumper

- 1-2 Motor Drive Power is provided externally from a pigtail on the DB25M cable from the AccuRange Line Scanner
- 2-3 Motor Drive Power is provided from the Control Board
- 4-5&7-8 Motor Drive is direct from the Power Source
- 5-6&8-9 Motor Drive is from the High Speed Interface Board Variable Power Drive
- 10-11&13-14 Distance Pulse count is stored in the FIFO Encoder1 data
- 11-12& 4-15 Scanner Motor Encoder is stored in the FIFO Encoder1 data
- 16-17&19-20 +5VDC & +12VDC is provided from the PC-104 computer bus
- 17-18&20-21 +5VDC & +12VDC is provided externally

An external positive pulse on the “Distance Pulse” connector (via a NTE3093 Opt Isolator and 74LS240 inverter) will increment an encoder counter (7474 dual D-Flip-Flop).
The signal will then be fed to the Encoder 1 location of the High Speed Interface digital output (byte 6 from the FIFO) via 74LS240 inverters, Jumpers 10-11 & 13-14, and Pins 21 & 22 of the DB25M connector. This signal is used to provide a module-8 “Distance Traveled” count to the Scanner Program.

The Scanner Program will automatically start the storing of the data in the StoreScan() Option when a ground is applied to the “Start Detector” connector. This ground signal is then fed via a NTE3093 Opt Isolator, 74LS240 inverter, and Pin 25 of the DB25M connector to the External Input 3 bit (Bit 2 in Byte 3) in the High Speed Interface FIFO output. The ground will be provided by a “Reflective Tape Detector”, which is supplied by TXDOT.

If the External Power option is selected with the jumpers, +12VDC from the External Power connector and +5VDC from the 78T05CT regulator is used to power the High Speed Interface Control board and the board can be mounted externally from the PC-104 bus.

Chapter 3
Scanning Laser Tests

3.1 Introduction

Once the scanning laser system was acquired, efforts were focused on trying to develop a system that could be used for measurements. As noted in Chapter 1, the basic Acuity system provided software that could be used for slow speed operations. A data acquisition program for high-speed operations specifically for the project had to be developed. During and following this development period, a number of data collection tests were performed to determine the capabilities of the system. Analysis efforts were required to distinguish usable signal from operational noise. A considerable amount of effort was required as well as in conducting various laboratory and field tests so that the final project data processing, analysis, and procedures could be developed. This chapter will describe these efforts and tests. The final project software and operational procedures are described in the following two chapters.

3.2 Pharr Data Collection Activities:

A section of US 281, north of Pharr, Texas was being investigated in September of 2001 by TxDOT to determine the reason for severe rutting on one of the southbound lanes. The MLS scanner, a slow speed transverse scanning system developed by TxDOT, was being used to obtain transverse profiles before a section of the pavement was removed to further investigate the base material of the pavement. Although the scanner software and distance interfacing hardware was not yet completely developed, it seemed to be a good time to see if comparisons between the profile from the MLS and laser scanning system could be made.

A mounting method on one of the TxDOT profiler vehicles for the scanning laser system was developed by TxDOT and project personnel. This system was then driven to the site where a number of scans were made on eight sections of this project. During this effort a number of operational and analysis problems with the scanning system were found. Although the result was that no useful data was obtained in the effort, it paved the way for the evolution of the analysis and hardware interfacing procedures developed later. For example, as the laser beam passes over an abrupt change in range reflection areas, the reflected intensity becomes quite low due to the defuse reflection angles that are encountered. This low intensity causes large spikes to occur in the calibrated range output. The best solution to this problem seemed to be to disregard the range output when spikes of low intensity occur and use pattern recognition techniques to fill in the missing data. A second problem noted was that the calibrated range data contained significant noise due to the rough texture that was being scanned. Because of these problems, a search for various noise filtering techniques began that could be used to reduce this noise. Figure 3.1 illustrates an example of one of the scans from one of the sections of the Pharr tests.

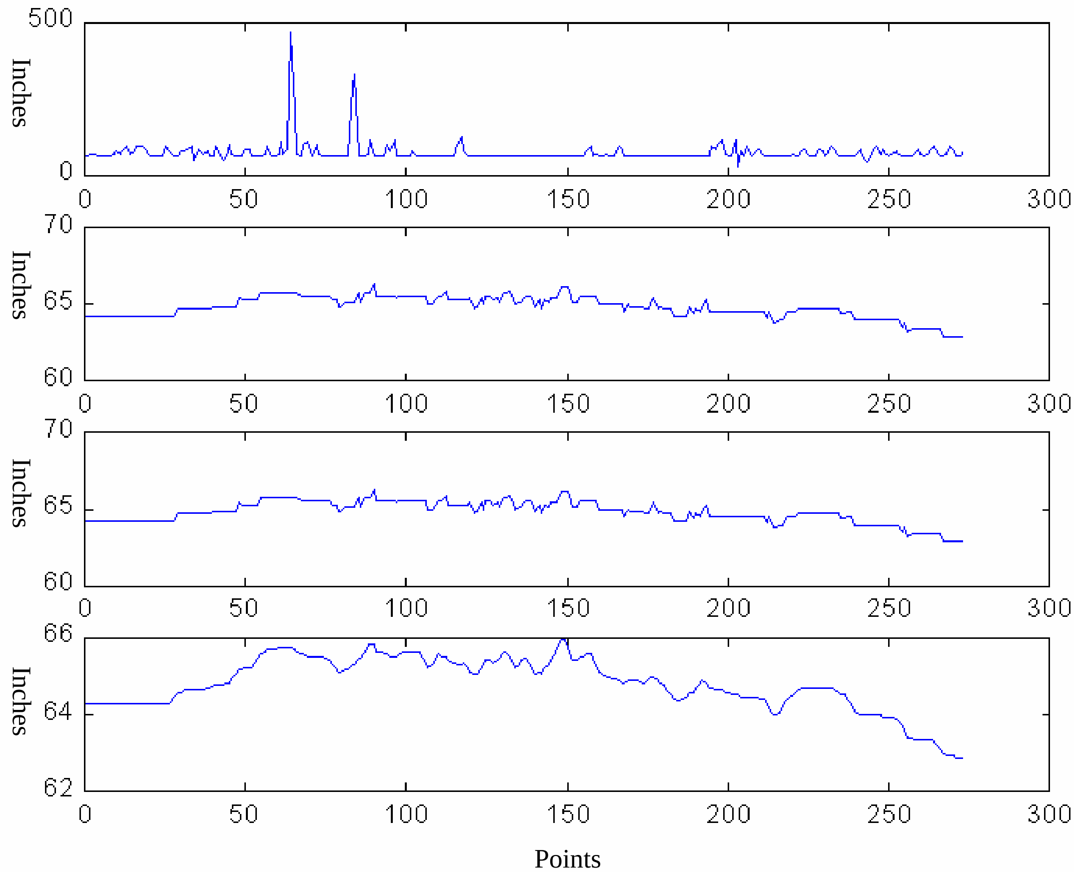


Figure 3.1 Example of Initial Noise Reduction Methods on Pharr Data Scan

Note: Y axis for all plots are in inches.

The figure illustrates four plots of a single scan from the scanning laser system to the target or a 12-foot cross section of the pavement. The first or upper plot is a plot of the raw laser scan. The vertical axis depicts the distance from the laser to the target in inches, and the horizontal axis represents the sample number as the scan transverses the pavement. As can be noticed, the profile is difficult to discern from this scan. Notice that noise spikes of 300 to 400 inches are indicated. The second through fourth subplots illustrated the effects of applying the initial noise reduction methods.

The scanner was also taken to the TTI/TxDOT Ride-Rut facility at the Texas A&M Riverside Annex for testing. Figure 3.2 illustrates the scanning system being driven over the simulated ruts at the facility.

Figure 3.2 Scanner on Vehicle at Test Track

Figure 3.3 illustrates in four subplots, the raw data and initial noise reduction methods applied to one of the scans taken over this facility at 40 miles per hour. The first or top subplot of Figure 3.3 illustrates the raw data from the scanning laser. However, in this case, the four-inch beams used to simulate the rut can be seen. The remaining subplots then indicate the results of applying the various initial processing steps. These analysis procedures included the polar to Cartesian coordinate correction procedures, windowing the data for noise spike removal, and converting the transverse distance points to equally spaced points, so that classical signal processing methods can then be applied for noise reduction or filtering. The third subplot of Figure 3.3 illustrates the use of the LMS (see Reference 3) for noise reduction. A running average filter was then applied to help further smooth the data and is illustrated in the fourth or bottom subplot in Figure 3.3. Matlab was used for much of this initial processing. The Matlab statements in Table 3.1 illustrate horizontal and vertical filtering methods. The coordinate correction methods are described along with the recommended processing in Chapter 4.

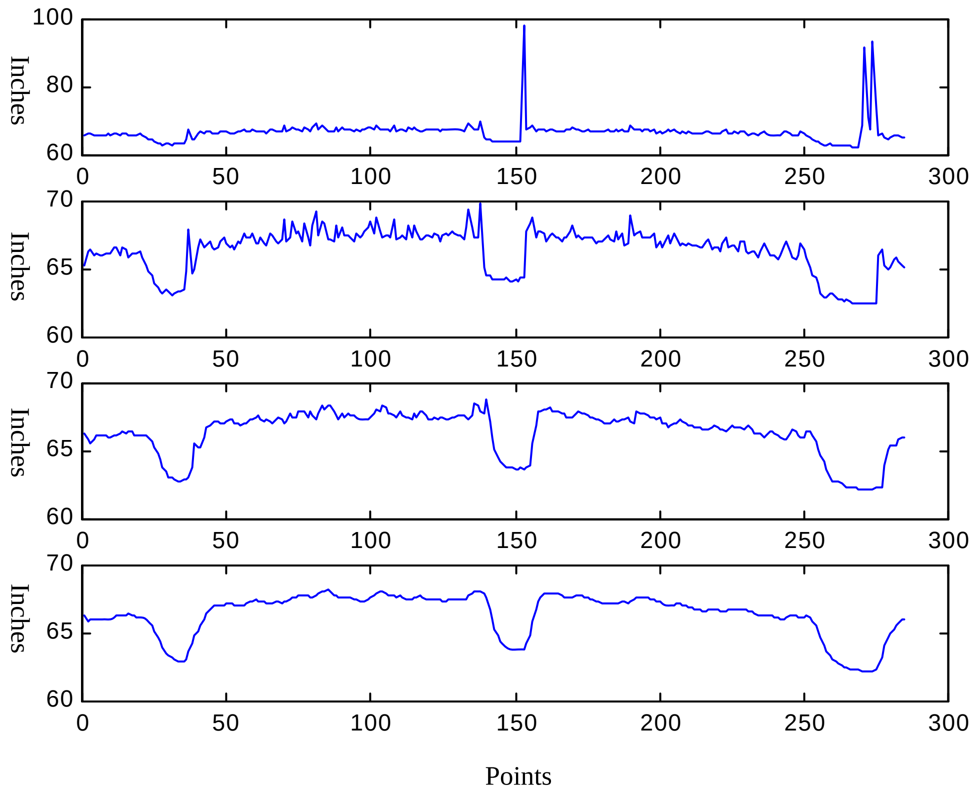


Figure 3.3 Initial Processing Procedures Applied to Scan from Ride/Rut Calibration Center. Run made at 40mph

As noted the LMS algorithm is used to filter the data. Before the LMS algorithm can be applied, the Matlab function *interp1* is used to adjust the point values to equally spaced distances. There are two methods of applying filtering and other processing algorithms for this type of application, horizontal filtering or processing and vertical processing. Horizontal filtering is used to filter the data for each scan. Vertical direction filtering is to filter the corresponding point in each scan. Horizontal and vertical smoothing or filtering is illustrated in Figure 3.4.

In the project example, each scan contains 250 reading. Table 3.1 illustrates a set of Matlab statements used for performing the LMS smoothing process for both methods. In this table the zzl array contains the data scans, where each row contains the points for a specific scan. Thus if there are 1000 scans and each scan contains 250 points, then the array size would be zzl(250,1000).

Horizontal Smoothing	**Vertical Smoothing**
for kk=1:ncols; *mu=0.1;sigma=1;alpha=0.1;L=20;N=B;* *bb=zeros(1,L+1);px=0;* *N=B+L+L+L* *% fold back data file for filter process* *nL=3*L* *for jj=1:nL;* *zzzz(jj)=zz1(nL-jj+1,kk);* *end;* *for jj=1:B;* *zzzz(jj+nL)=zz1(jj,kk);* *end;* *d(1)=zzzz(1);* *x(1)=0;* *for ll=2:N* *d(ll)=zzzz(ll);* *x(ll)=zzzz(ll-1);* *end;* *[y,bb,px]=spnlms(x,d,bb,mu,sigma,alpha,px)* *;* *for ll=1:B;* *zz1(ll,kk)=y(ll+nL-1);* *end;* *end;*	*for kk=1:B;* *mu=0.1;sigma=1;alpha=0.1;L=20;N=ncols;* *bb=zeros(1,L+1);px=0;* *N=ncols+L+L+L* *% fold back data file for filter process* *nL=3*L* *for jj=1:nL;* *zzzz(jj)=zz1(kk,nL-jj+1);* *end;* *for jj=1:ncols;* *zzzz(jj+nL)=zz1(kk,jj);* *end;* *d(1)=zzzz(1);* *x(1)=0;* *for ll=2:N* *d(ll)=zzzz(ll);* *x(ll)=zzzz(ll-1);* *end;* *[y,bb,px]=spnlms(x,d,bb,mu,sigma,alpha,px)* *;* *for ll=1:ncols;* *zz1(kk,ll)=y(ll+nL-1);* *end;* *end;*

Table 3.1 Matlab Statements Illustration Horizontal and Vertical Smoothing using the LMS Method

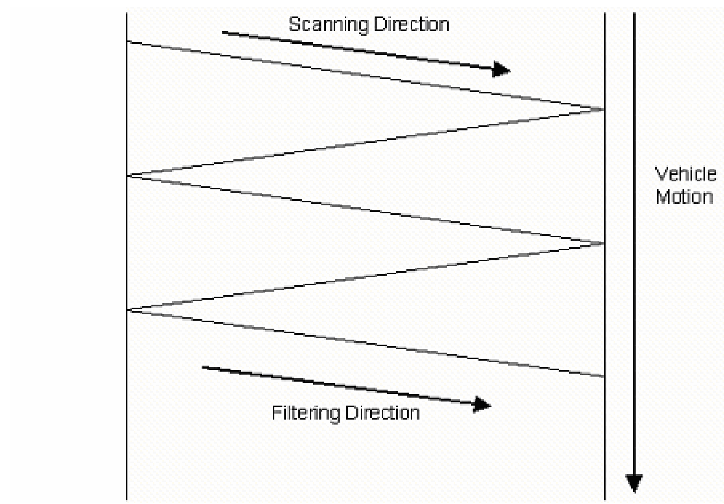
3.3 Static and Dynamic Laboratory Tests

A number of tests were conducted in the Transportation Instrumentation Lab (TIL) at The University of Texas at Arlington (UTA) on the scanning system to insure the laser and scanning system were functioning properly. The tests were not extensive and thus were only used to gain a better understanding of the problems noted in the field test and to aid in improving the noise reduction methods that were being developed and applied to obtain usable data.

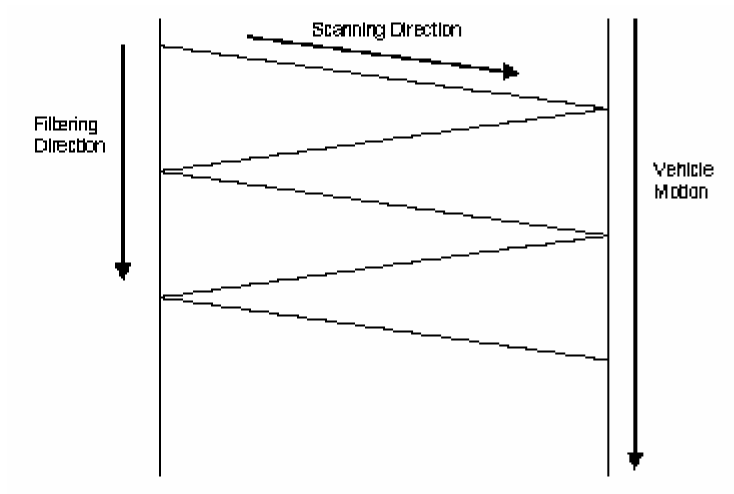
For the scanning system, the sample resolution and accuracy of the laser are a function of the sampling rate and range. The laser resolution for the different sample rates are given by the manufacturer is illustrated in Table 3.2.

Maximum Attainable Sample Rates (samples/second)			
Resolution in inches	**6 Feet**	**30 Feet**	**55 Feet**
0.0062	2304	677	390
0.0125	4609	1355	781
0.0250	9218	2711	1562
0.0500	18346	5422	3125
0.1000	36873	10845	6250
0.2000	50000	21691	12500
0.4000	50000	43382	25000
0.8000	50000	50000	50000

Table 3.2 Equipment Resolution



Horizontal
Smoothing



Vertical
Smoothing

Figure 3.4 Horizontal and Vertical Smoothing

In Chapter 2, it is noted that for the scanning system, rotating at 2600 RPM, 235 samples per scan are obtained. As discussed in Chapter 2, the laser is swept across the target by a rotating mirror. The laser must first be reflected by the rotating mirror, and then passed through a glass lens. The resolution provided in the Acuity specifications was considered the upper accuracy limit. The scanning laser system was tested with the rotating mirror assembly held stationary to determine the expected resolution after the beam has passed through the lens. The asphalt sample target was placed on a computer controlled translation table. The scanning laser system was then located above the translation table and the target moved vertically at various steps beneath the laser beam as illustrated in Figure 3.5

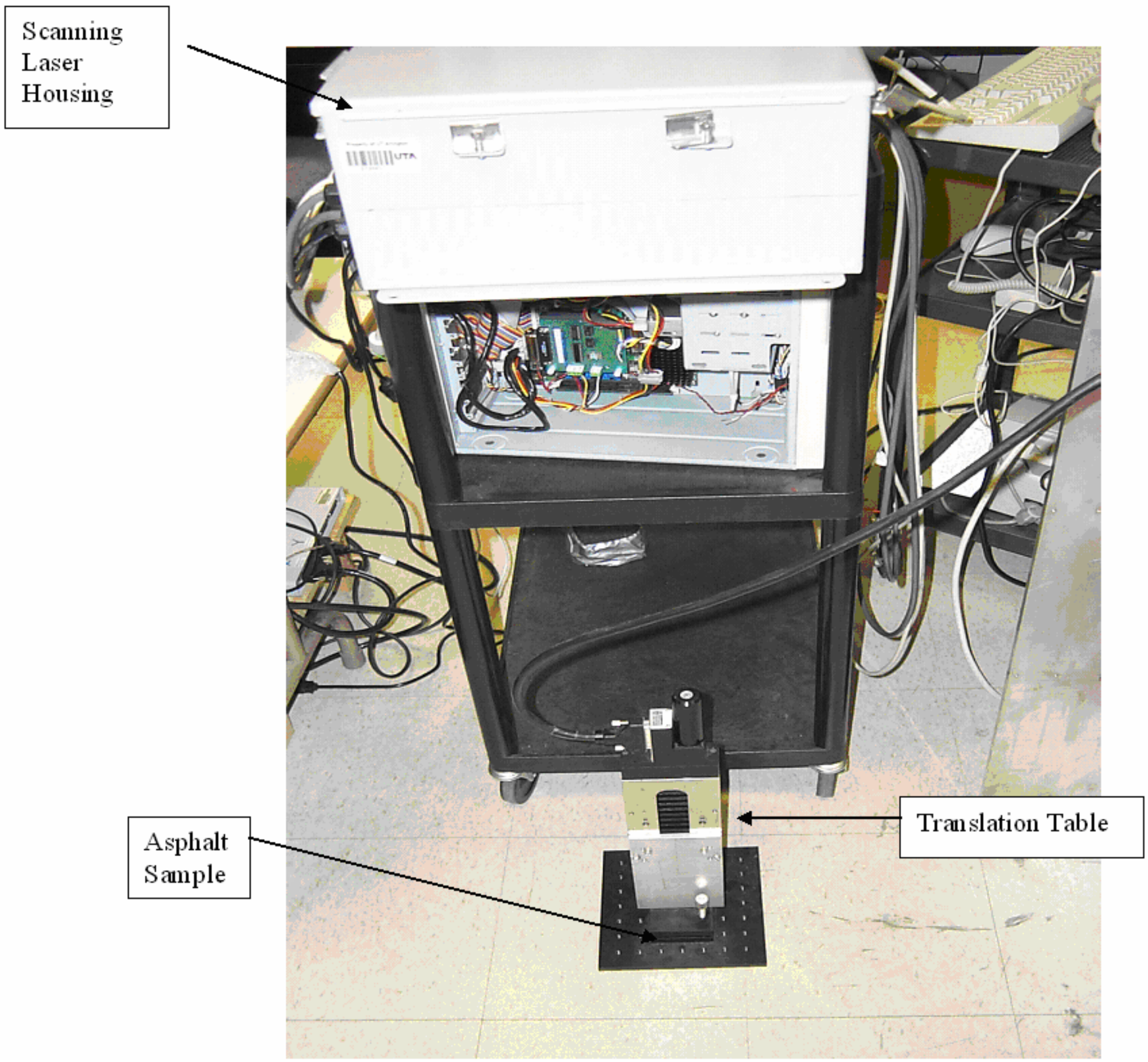


Figure 3.5 Static Displacement Tests

As noted in Table 3.2 the accuracy of the laser is a function of both the distance from the laser to the target and the sampling rate. The laser was approximately four feet from the target for these tests and the sample rate varied between 20 and 50 kHz. In Table 3.2 it is noted that for a six-foot range, providing a 12-foot scan, a sampling range between 36,873 and 50,000 samples per second would provide a 0.1-inch resolution. Note that better resolution is obtained by slowing down the sampling rate, but at the expense of the number of samples taken per scan.

The target was moved at various displacements beneath the laser and samples made and recorded. Different heights were investigated as well as different target samples. Static Readings using the translation table and scanning laser were made for seven sets of 0.05-inch (1270 micrometers) displacements at 20 thru 50 microseconds (50 – 30 KHz sampling rate). These tests were then repeated for 0.2-inch (5080 micrometers) displacements. Five hundred points were obtained for each position. The best sampling rate found for the static readings was a 33 KHz or a sampling interval of 30 microseconds. Table 3.3 provides the first 6 points for samples at each position for this best case. The averages in inches and standard error for each data set for each position are also provided.

Raw Data For 30 Microsecond Data Sampling							
Displacements (inches)	Readings (inches)						
	0	1	2	3	4	5	6
0	36.81	36.87	36.82	36.75	36.87	36.82	36.68
0.2	36.69	36.63	36.43	36.43	36.62	36.43	36.55
0.4	36.3	36.43	36.48	36.43	36.41	36.43	36.43
0.6	36.31	36.37	36.18	36.31	36.44	36.24	36.24
0.8	36.05	35.92	36.11	36.11	35.99	36.06	35.99
1	35.82	35.77	35.82	35.84	35.69	35.82	35.69

Statistics		
mean	std error	range
36.78	0.03414	0.160
36.57	0.04605	0.224
36.40	0.03874	0.222
36.25	0.04295	0.234
35.97	0.03796	0.222
35.77	0.03621	0.182

Table 3.3 Displacement Readings and Statistics for 33 KHz sampling

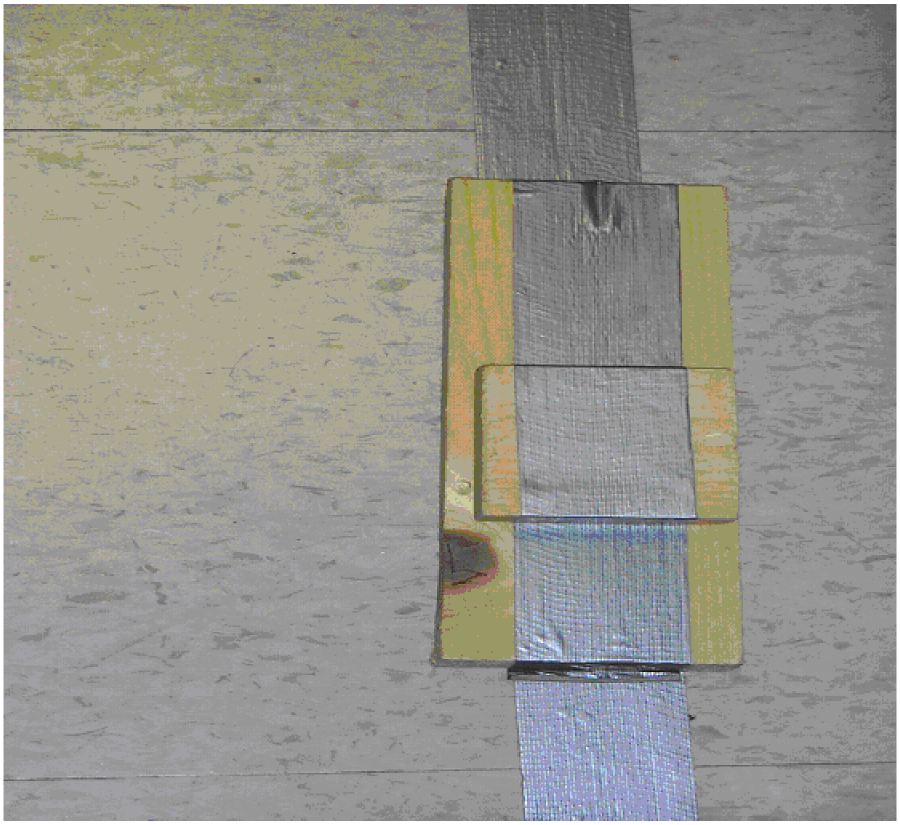


Figure 3.6 Target Used for the Dynamic Laser Tests

To simulate the effects of the laser while rotating, dynamic measurements were performed. The scanning laser system was placed over the laboratory floor and a target as illustrated in Figure 3.6. It was also tested at various positions along the scanning line. For the target shown, a ¾ inch block of wood was placed over a second block to simulate abrupt changes in the displacement measurement path of the scanning laser. The block was positioned for tests at both the ends and center of the measurement scan. No noticeable difference was found in the final result because of the target position. In Figure 3.7, the noise occurring due to the abrupt displacement changes is noted and the same noise reduction methods discussed earlier are applied. From the results of the static and dynamic tests the resolution does appears to approach the manufacturer specifications. The static tests did not reveal the noise problems noted in the dynamic tests and field data collections as the laser beam and target were stationary.

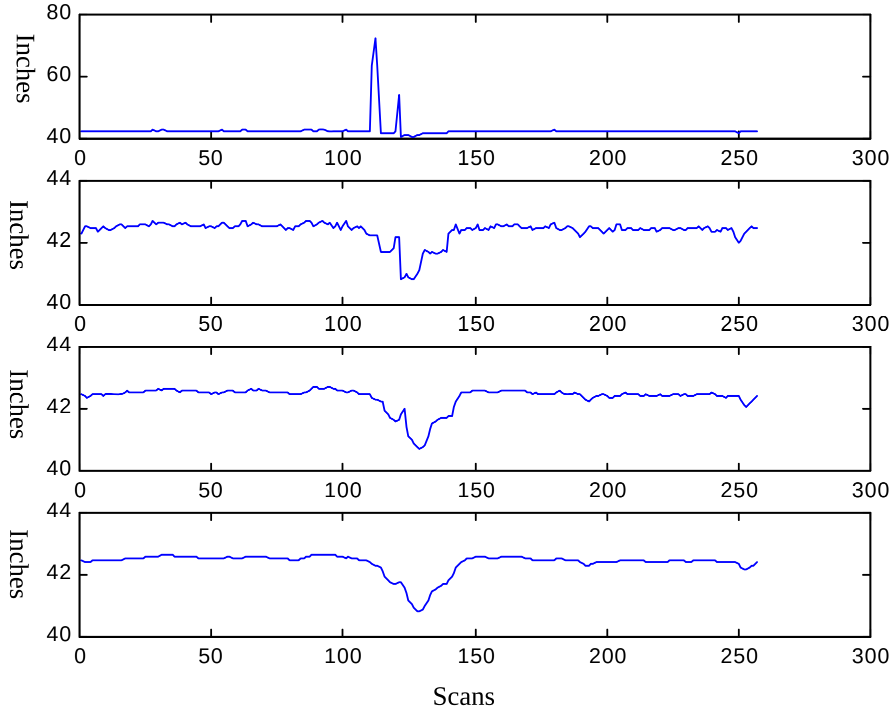
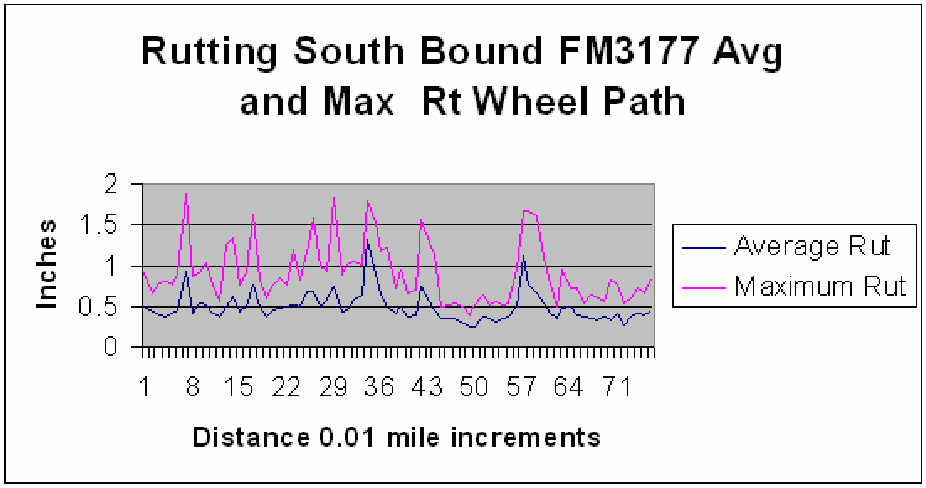


Figure 3.7 Raw and Processed Data from Dynamic Tests

3.4 Austin Field Tests

As a result of the laboratory tests and investigations of the data, the changes to the processing methods were made as will be discussed in the next chapter. Additionally, a procedure was added to the processing to provide an estimated rut reading using the string line rut measuring procedure. The scanning laser system was taken to a site where rutting was noted on FM 3177 outside of Austin. Figure 3.8 depicts the scanning laser mounted on the back of the TxDOT test vehicle. The scanning system was run over the southbound lane, just south of Decker Lane. The average maximum rut measurements using the scanning laser system and applying the string line method was computed. The scanning laser data was acquired for approximately 0.8 mile in the southbound lane. The results are illustrated for each wheel path in Figures 3.9. The figure provides the maximum and average rut for the 0.01 mile for each set of scans (every four feet) and the average of the maximum rut reading. The plots illustrate the results of processing all laser scans readings for each wheel path as the vehicle is driven over the 0.8-mile section.

Figure 3.8 Scanning Laser Mounted on TxDOT Test Vehicle



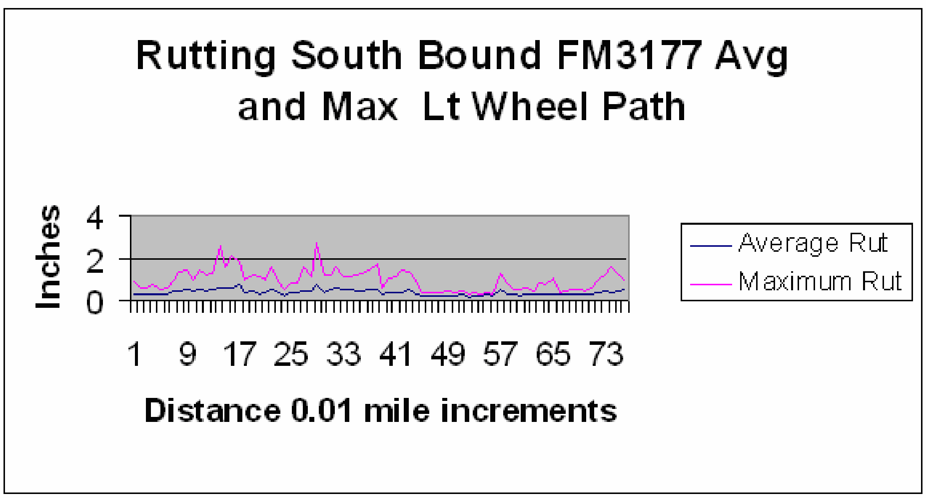


Figure 3.9 Continuous Rut Measurements Right Wheel Path

Measurements were next taken at a site just east of FM 3177. This site is located at the end of the FM 3177 section just before arriving at Decker Lane. The measurements began just south of the FM 3177 and Decker Lane intersection on Decker Lane. The test vehicle is shown at the beginning of this measurement site in Figure 3.10. Deep rutting had been observed earlier, for this site and continuing east for approximately 0.1 mile. The extent of the rutting at this site is illustrated in Figure 3.11. A run was made at 40 miles per hour. Figure 3.12 illustrates the resulting 3-D plot of the multiple scans made from the scanning laser over the position depicted in Figure 3.11. As can be observed, a close approximation to the actual pavement is illustrated.

Figure 3.10 Test Vehicle at Deep Rutting Site on Decker Lane

Figure 3.11 Deep Rutting at Decker Lane Location

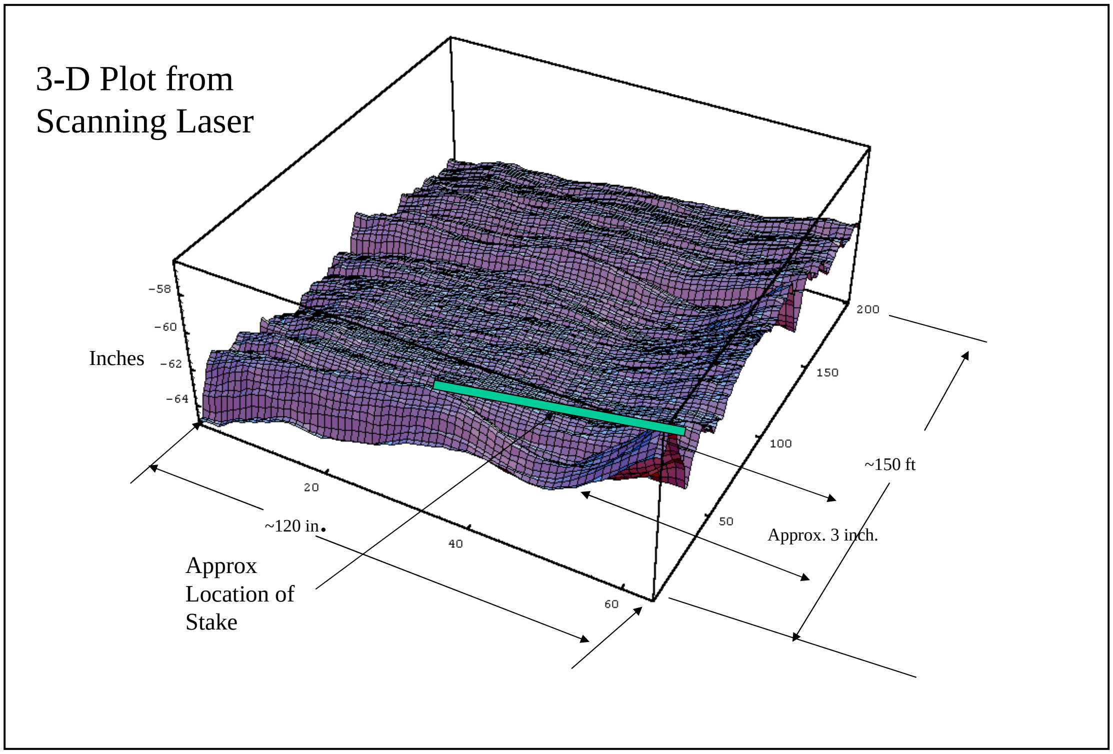


Figure 3.12 3-D plot from Scanning Laser of Decker Lane Site

Two scanning laser systems were purchased during the project, one for interfacing to the PC via the ISA bus, and the second via the PC104 bus. The second was purchased for easier interfacing with the smaller embedded PC's. Only the ISA bus version was available at the time the first unit was purchased. The data collected on the Pharr project was with the ISA unit. Although both systems experienced extensive noise problems during field and laboratory test, it was noted that the first system had more noise than the second did. It was noted that much of the noise would occur when the amplitude of the return signal was below the normal operating range. In discussions with the manufacturer, it was decided that a better return signal might be possible by increasing the laser power. Since Acuity had indicated that the TxDOT application of the scanning laser was the only one that was being used for that purpose, no experience was available to determine if this power increase would help. However, at the time it seemed plausible. Thus, the first system was sent back to the manufacturer and the laser was upgraded to a 50 milli-watt unit. During the laboratory tests at UTA, however, no discernable difference between the more powerful laser and the 35 milli-watt laser was noted, except when the target was directly perpendicular to the pavement surface. When the target was perpendicular to the floor, the 50 milli-watt laser actually gave worst results. A displacement was indicated when there was not a displacement. Later, it was conjectured this indication was due to the displacement lookup table used by the laser system to determine the reading sent to the PC. Both versions were used for the tests conducted on FM 971 near Granger, Texas. These tests are discussed next.

3.5 Granger Rutting Tests

Following the Pharr tests in the fall of 2001 researchers began to focus on methods to separate the profile signal from the large noise spikes. During the early part of 2002, it became less and less likely that classical filtering methods alone would be able to satisfactorily smooth the signal. The Bezier Curve was investigated as a means of fitting a smooth surface to the profile. The curve was fit to the profile data after the spike removal and filtering procedures were applied. This procedure resulted in a significant improvement to the signature of the profile scans. The method was used in processing the plots shown in Figures 3.9 and 3.12. The plots shown in Figures 3.9 were the result of further applying the string-line rut method to the scans after they had been processed.

Because of the success in the April 2002 tests in Austin, it was decided that the system and analysis methods were ready for a more complete evaluation to determine how well the rut measurements compared to existing rut measuring methods. In May of 2002 stationary (static) and moving (dynamic) measurements were conducted at a site on FM 971 near Granger, Texas for performing these comparisons. In addition to the scanning laser, the dipstick, straight edge, MLS profiler, laser and acoustic rut systems were all used. For the static tests, readings from stationary scans of the scanning laser, the dipstick, the MLS, and the straight edge were used to find the maximum and average maximum rut in both the right and left wheel paths. For the dynamic tests, rut measurements were made with the scanning laser and the rut readings from the TxDOT laser and acoustic rut systems. The tests were conducted on the eastbound lane of FM 971 approximately 2 miles west of Granger (see Figure 3-13).

Figure 3.13 Site of Scanning Laser Test on FM 971

The section selected had rutting at all levels from no rutting at all to severe rutting in both wheel paths, and all of these cases occur within a few hundred feet of each other. The Section layout is illustrated in Figure 3.14. For the experiment, four sections were selected, where section one had no or little rutting, and sections two to four had various levels from light to severe.

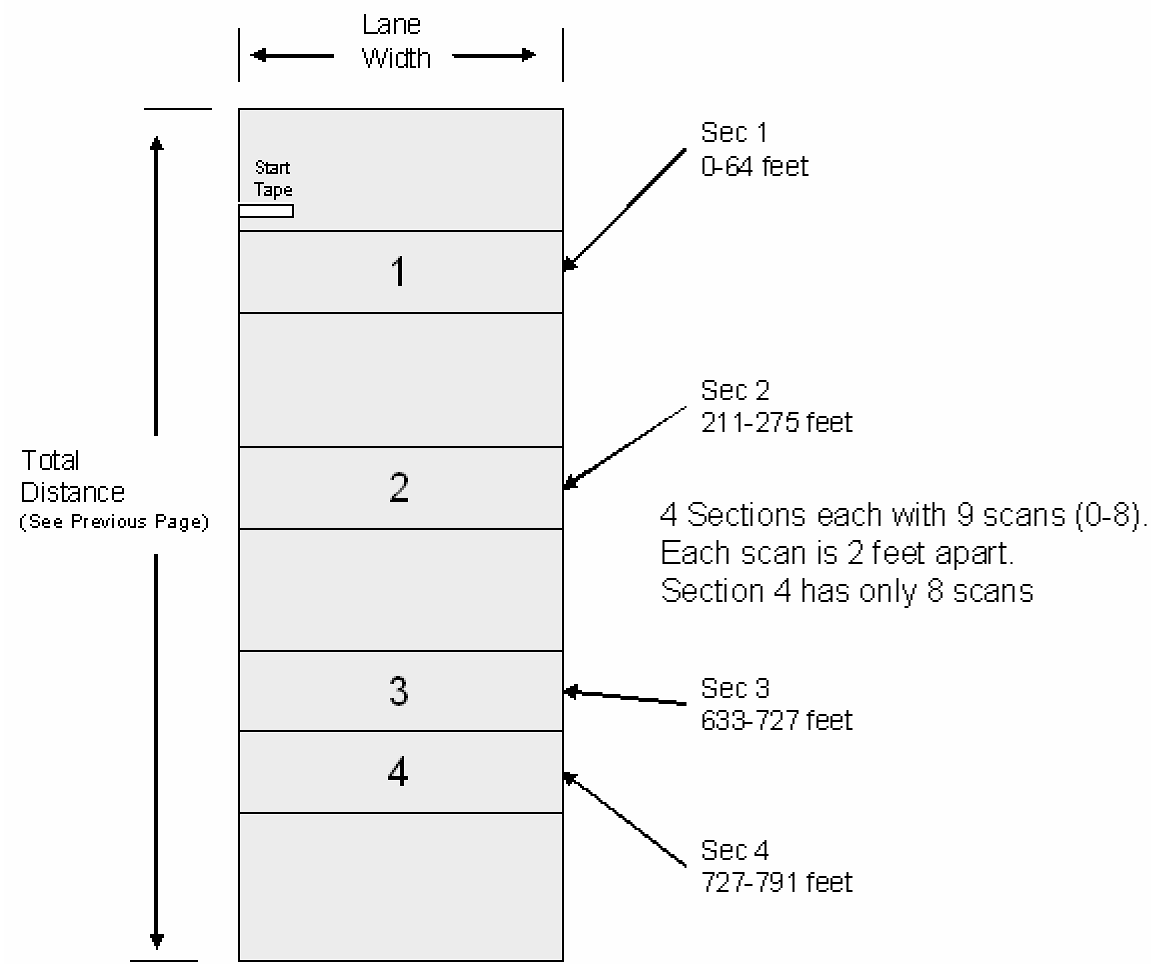


Figure 3.14 Layout of Sections for Tests on FM 971

For the static tests, nine sets of measurements were made by all six devices on sections one to three. Each set of measurements was eight feet apart. Only eight sets of measurements were taken on section four as the last set on section three is shared with the first set on section four. Each set of points or measurements are referred to as a scan. The scanning laser was placed over each scan and multiple scans made at each of the positions. Similar measurements were made by the other devices, the MLS, dip stick, straight edge, and the acoustic and laser rut vehicles. Three additional measurements were taken with the straight edge between each of the eight feet scans. The MLS, dip stick and scanning laser systems provided a set of transverse profile points which were processed using the string line algorithm in order to get the respective rut reading. The TxDOT acoustic and laser rut measurement vehicles used the standard PMIS rut algorithm (three point string-line method). The straight edge was used for obtaining the maximum rut measurements from the bar to the road. Figure 3-15 illustrates the straight edge placed over an area with severe rutting on section four. Figure 3-16 illustrates operations using the dip stick (foreground) and MLS (background) while making measurements on section one.

The program, FINDRUT, described in the next chapter, was applied on the raw scanning laser data to compute the rut for both the left and right wheel paths. In order to compare these readings with the other devices, the string line algorithm was applied to the set of profile points from both the MLS and dip stick. The method used for the straight edge provides a similar maximum rut for each position measured.

The rut readings measured between the different processes averaged over the four 64 foot sections were similar to each other, although the acoustic unit provided consistently larger readings on the left side (see Figure 3.17). The measurements within each 8-foot scan had more variation. This can be noted in Table 3.4 and Table 3.5, and from the plots in Figure 3.18 to Figure 3.21.

The individual eight-foot section scans are given in Table 3.5. The table shows the differences between readings of each of the methods. The comparisons between each of the measurement methods with the scanning laser measurements determined from these tables are illustrated in Figures 3.18 to 3.21. These plots illustrate the differences between the scanning laser, and the variations between each of the different devices.

The variations shown in Figures 3.18 to 3.21 can be expected because of the different lengths of the transverse or lane width measurements covered by the MLS beam, the scanning laser, and the dip stick. The scan across the lane width by the scanning laser, for instance covered a 12-foot length, and was greater than those of the other instruments. The path covered by the dip stick was greater than the MLS and the paths covered by both the dip stick and MLS were greater width than the five acoustic and laser sensors. The length of the maximum value found using the string line method is affected by the length and number of points used. Figure 3.22 can be used to illustrate why the max rut readings can be different depending on the transverse path lane for the string line method. For example, in this figure the width, W1, is greater than the width, W2. The maximum perpendicular distance from the road surface to the string for W1 is the distance D1. This distance is greater than D2, which is the maximum perpendicular distance for W2.

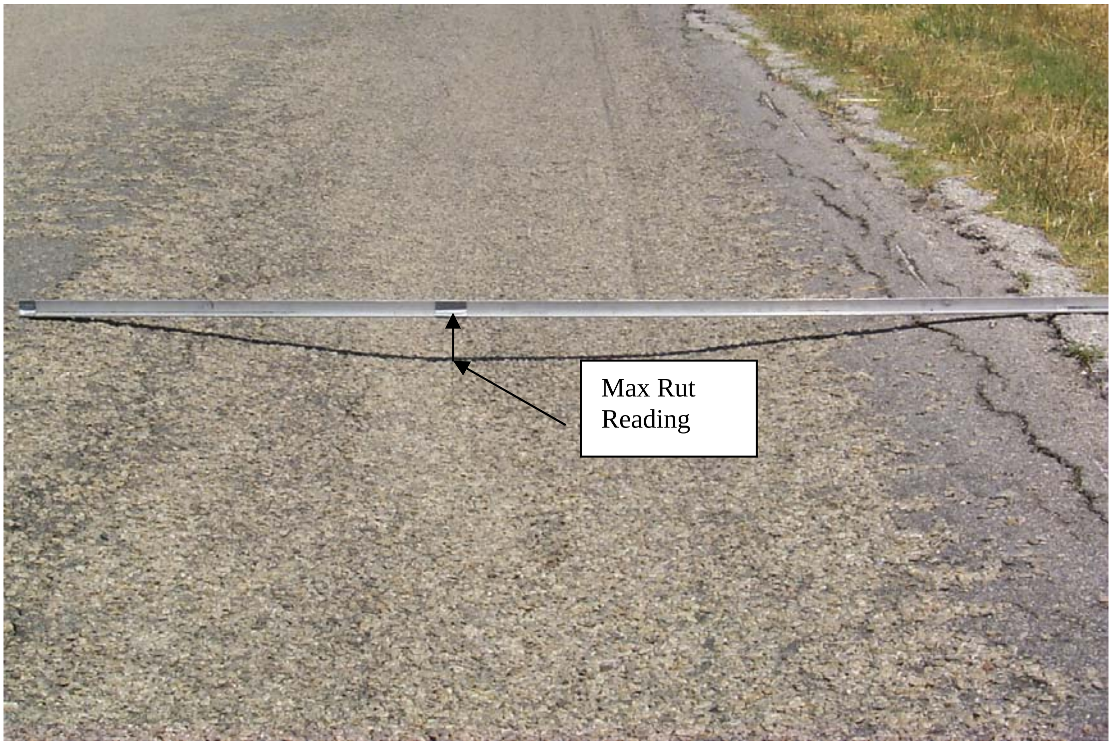
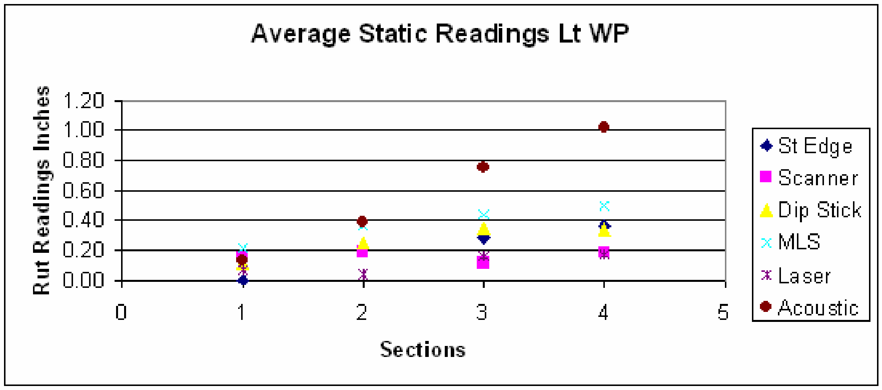


Figure 3.15 Straight Edge at Section 4 (Severe Rutting) Site

Figure 3.16 Dip Stick and MLS In Use At Section 1

Sections 1 Through 4 Average Static Rut Readings In Inches												
Section	Straight Edge		Scanner		Dip Stick		MLS		Laser Rut		Acoustic Rut	
	Left	**Right**	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**
1	0.00	0.00	0.15	0.29	0.11	0.08	0.22	0.20	0.07	0.10	0.13	0.00
2	0.21	0.92	0.19	0.77	0.25	0.73	0.36	0.98	0.04	0.53	0.39	0.28
3	0.28	1.17	0.12	0.92	0.34	1.16	0.44	0.90	0.16	0.56	0.75	0.42
4	0.36	2.39	0.18	1.96	0.34	2.19	0.50	1.63	0.17	1.08	1.02	1.21

Table 3.4 Average Static Rut Readings between Each Device For Each Section



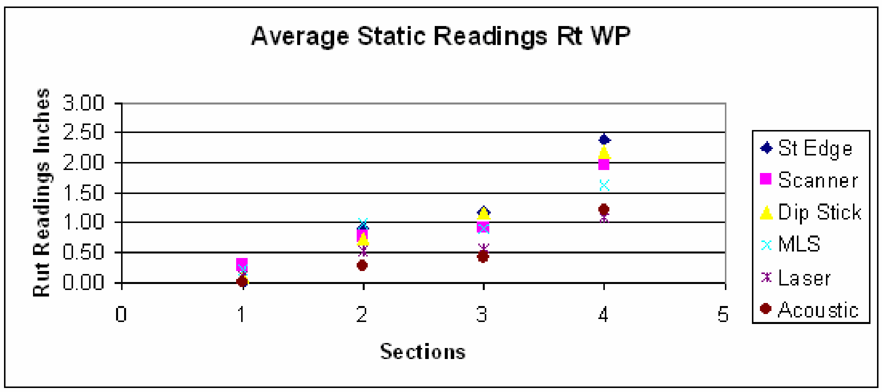


Figure 3.17 Average Static Comparisons All Sections

Section 1 Static Device Readings In Inches for Rut Test May 15, 2002 on FM 971												
Scan	**Straight Edge**		**Scanner**		**Dip Stick**		**MLS**		**Laser Rut**		**Acoustic Rut**	
#	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**
0	0.00	0.00	0.08	0.35	0.08	0.15	0.22	0.27	0.00	0.11	0.44	0.00
1	0.00	0.00	0.22	0.32	0.06	0.11			0.00	0.22	0.03	0.00
2	0.00	0.00	0.26	0.35	0.10	0.07	0.19	0.14	0.00	0.01	0.04	0.00
3	0.00	0.00	0.04	0.30	0.13	0.00	0.24	0.21	0.05	0.07	0.04	0.00
4	0.00	0.00	0.15	0.14	0.19	0.07	0.26	0.18	0.00	0.07	0.03	0.00
5	0.00	0.00	0.18	0.27	0.14	0.09	0.21	0.22	0.00	0.02	0.20	0.00
6	0.00	0.00	0.20	0.30	0.15	0.07	0.22	0.22	0.12	0.07	0.17	0.00
7	0.00	0.00	0.11	0.34	0.06	0.08	0.20	0.19	0.03	0.18	0.04	0.00
8	0.00	0.00	0.15	0.27	0.12	0.11	0.19	0.20	0.00	0.19	0.17	0.00
AVG	0.00	0.00	0.15	0.29	0.11	0.08	0.22	0.20	0.02	0.10	0.13	0.00

Table 3.5 Static Comparisons between Scanning Systems and Other Devices

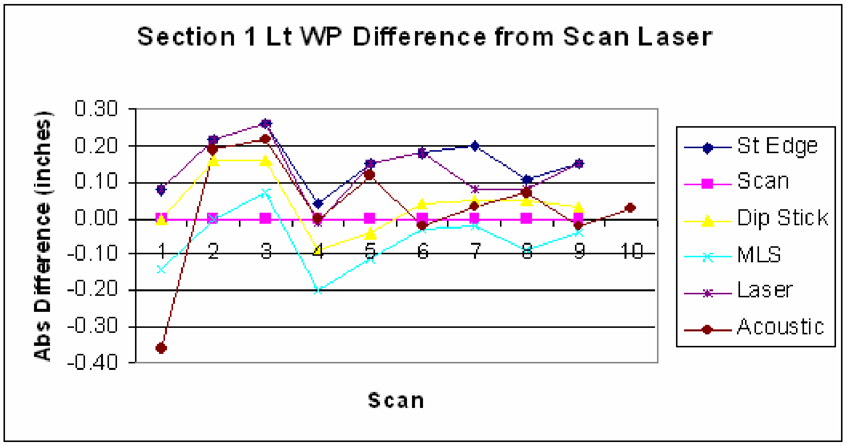
Section 2 Static Device Readings In Inches for Rut Test May 15, 2002 on FM 971												
Scan	**Straight Edge**		**Scanner**		**Dip Stick**		**MLS**		**Laser Rut**		**Acoustic Rut**	
#	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**
0	0.63	0.88	0.22	0.62	0.46	0.56	0.63	0.87	0.05	0.38	0.24	0.32
1	0.38	0.88	0.32	0.61	0.30	0.61	0.44	0.91	0.00	0.42	0.33	0.27
2	0.38	0.50	0.23	0.45	0.21	0.45	0.37	0.65	0.11	0.51	0.26	0.11
3	0.25	1.00	0.17	0.73	0.20	0.68	0.32	0.91	0.04	0.45	0.31	0.25
4	0.25	1.38	0.16	1.09	0.13	1.01	0.29	1.47	0.00	0.69	0.37	0.58
5	0.00	1.25	0.07	1.10	0.24	0.92	0.27	1.28	0.00	0.70	0.58	0.46
6	0.00	0.88	0.17	0.80	0.22	0.85	0.32	0.93	0.07	0.60	0.47	0.22
7	0.00	0.75	0.12	0.79	0.22	0.73	0.32	0.88	0.00	0.49	0.48	0.15
8	0.00	0.75	0.24	0.75	0.23	0.72	0.32	0.94	0.06	0.51	0.44	0.20
AVG	0.21	0.92	0.19	0.77	0.25	0.73	0.36	0.98	0.04	0.53	0.39	0.28

Section 3 Static Device Readings In Inches for Rut Test May 15, 2002 on FM 971												
Scan	**Straight Edge**		**Scanner**		**Dip Stick**		**MLS**		**Laser Rut**		**Acoustic Rut**	
#	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**
0	0.00	0.88	0.13	0.88	0.24	0.94	0.33	0.79	0.06	0.51	0.57	0.15
1	0.25	1.13	0.13	0.90	0.33	0.99	0.41	0.67	0.17	0.47	0.80	0.19
2	0.25	1.38	0.12	1.00	0.27	1.26	0.41	1.14	0.13	0.41	0.74	0.35
3	0.38	1.00	0.08	0.89	0.33	1.28	0.43	1.05	0.11	0.62	0.77	0.49
4	0.25	1.13	0.14	0.86	0.35	1.16	0.42	0.87	0.24	0.65	0.60	0.28
5	0.25	1.13	0.10	0.79	0.27	1.02	0.41	0.74	0.13	0.44	0.80	0.37
6	0.38	1.25	0.05	0.97	0.42	1.21	0.52	0.95	0.21	0.45	0.71	0.41
7	0.50	1.50	0.17	1.06	0.53	1.42	0.59	0.98	0.20	0.66	0.82	0.70
8*	0.63	2.38	0.22	1.95	0.51	2.00	0.59	1.47	0.24	0.87	0.90	0.81
AVG	0.32	1.31	0.13	1.03	0.36	1.25	0.46	0.96	0.16	0.56	0.75	0.42

Table 3.5 (Cont) Static Comparisons between Scanning Systems and Other Devices

Section 4 Static Device Readings In Inches for Rut Test May 15, 2002 on FM 971												
Scan	**Straight Edge**		**Scanner**		**Dip Stick**		**MLS**		**Laser Rut**		**Acoustic Rut**	
#	**Left**	**Rt. W. P.**	**Left**	**Rt. W. P.**	**Left**	**Rt. W. P.**	**Left**	**Right**	**Left**	**Right**	**Left**	**Right**
0*	0.63	2.38	0.22	1.95	0.51	2.00	0.59	1.47	0.24	0.87	0.90	0.81
1	0.38	2.50	0.28	2.40	0.37	2.35	0.52	2.05	0.07	1.10	1.08	1.21
2	0.38	2.88	0.22	2.61	0.33	2.66	0.57	2.18	0.12	1.20	1.30	1.62
3	0.50	2.63	0.21	2.46	0.40	2.71	0.67	1.62	0.18	1.21	1.03	1.58
4	0.50	2.63	0.14	2.41	0.41	2.03	0.54	1.44	0.17	1.08	0.91	1.42
5	0.25	2.13	0.17	1.49	0.35	2.00	0.53	1.52	0.74	1.13	0.91	1.08
6	0.38	2.50	0.13	1.76	0.26	2.28	0.39	1.64	0.00	1.19	1.36	1.30
7	0.25	2.00	0.16	1.38	0.21	2.04	0.39	1.58	0.00	1.24	0.92	1.20
8	0.00	1.88	0.10	1.22	0.21	1.63	0.29	1.15	0.04	0.66	0.73	0.70
AVG	0.36	2.39	0.18	1.96	0.34	2.19	0.50	1.63	0.17	1.08	1.02	1.21

***** Note: Scan 8 of Section 3 is the same as Scan 0 of section 4 because of the contiguous sections.



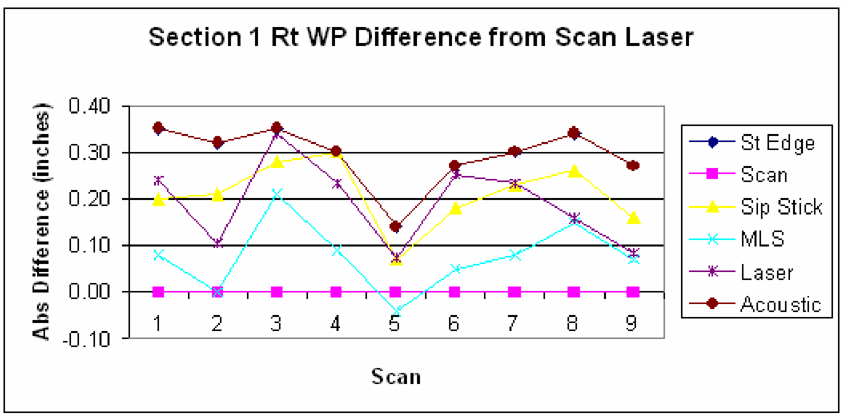
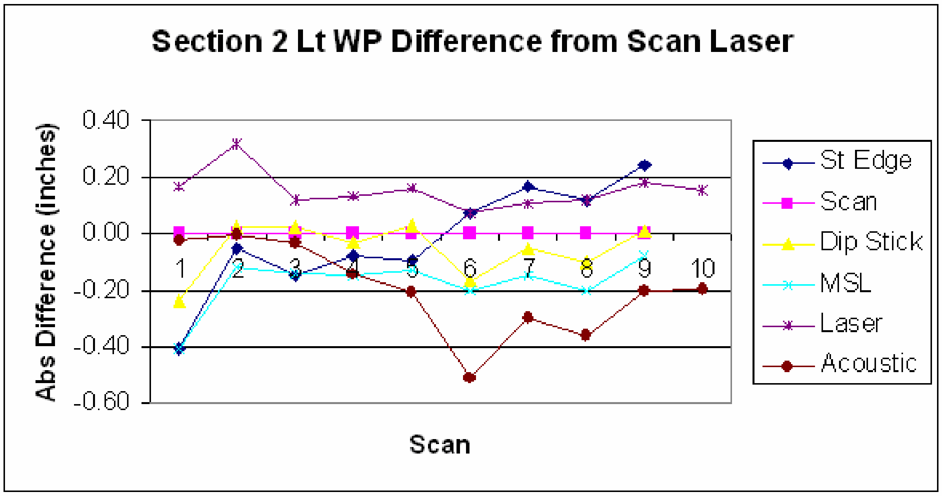


Figure 3.18 Static Comparisons with Scanning Laser Sec 1



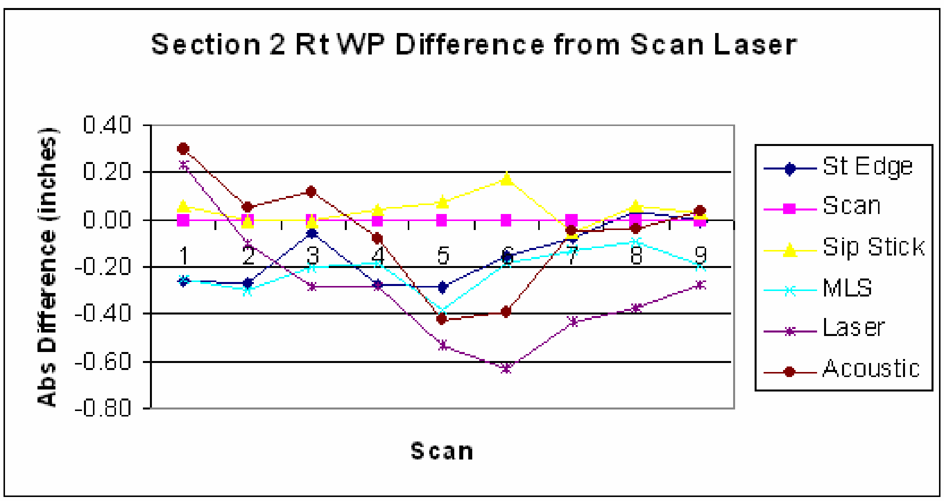
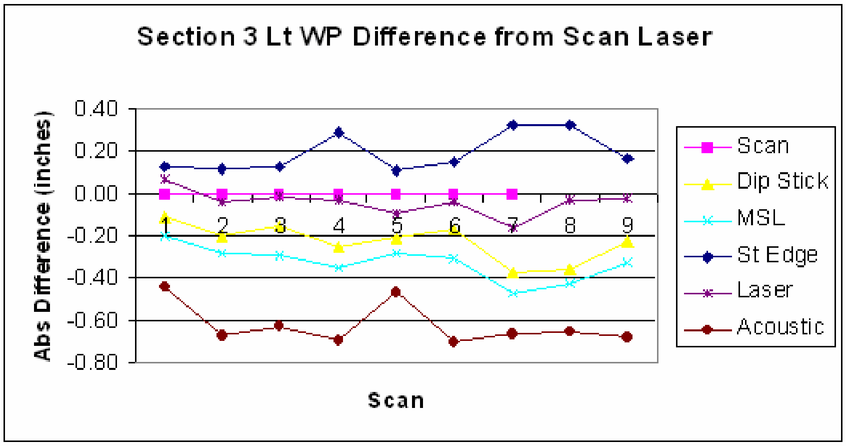


Figure 3.19 Static Comparisons with Scanning Laser Sec 2



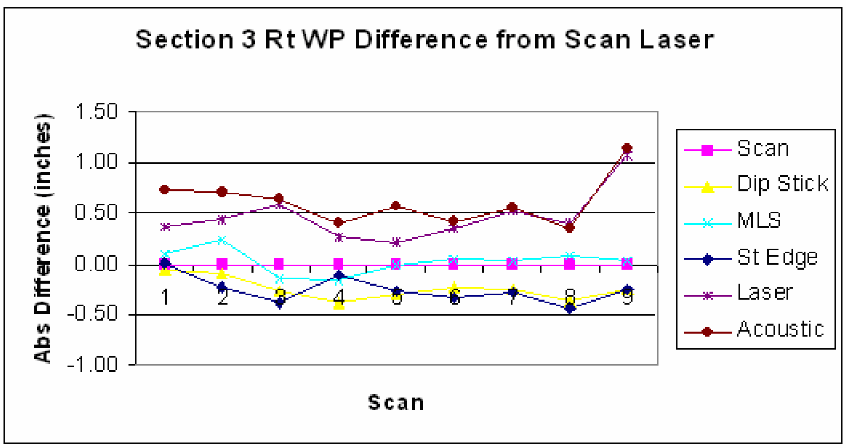
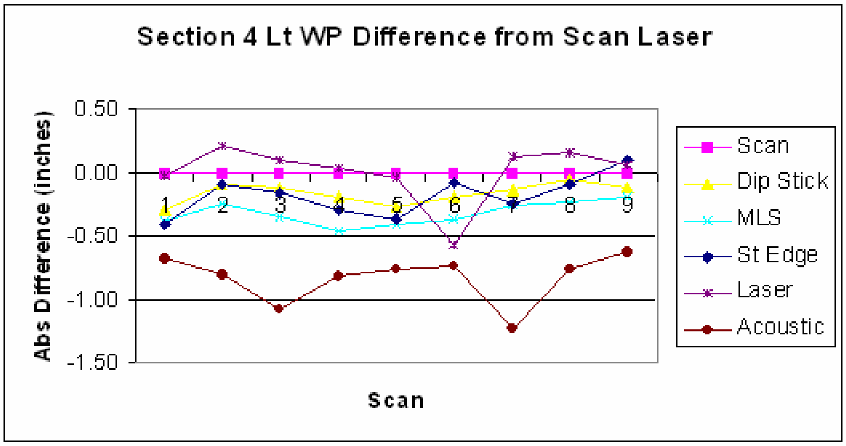


Figure 3.20 Static Comparisons with Scanning Laser Sec 3



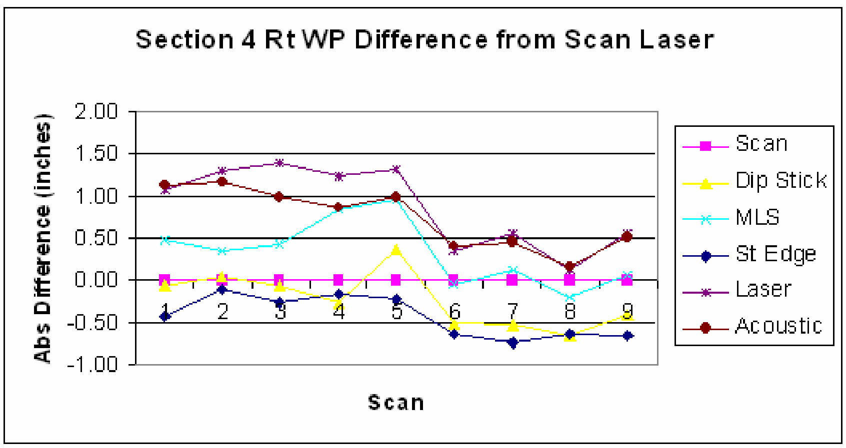


Figure 3.21 Static Comparisons with Scanning Laser Sec 4

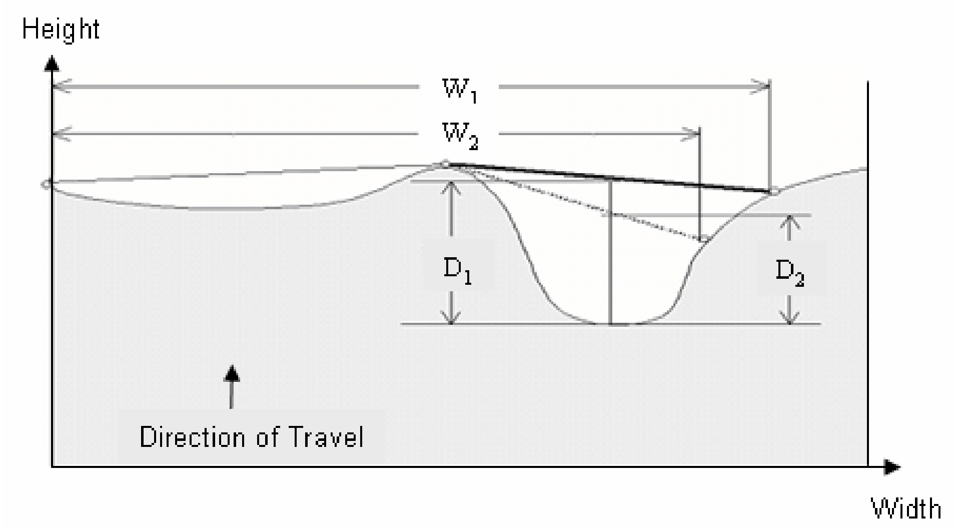
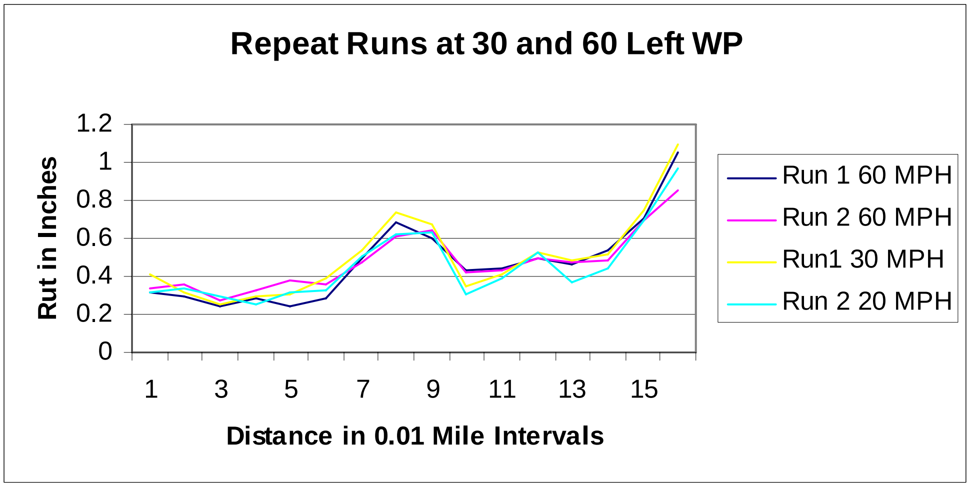


Figure 3.22 Measurements from String Line Method is Dependent on String Length

The Scanning laser system and the TxDOT acoustic and laser rut vehicles were then driven over the four sections. The runs were made for the complete test area that included not only the 64-foot sections but also the areas between each section. All runs began at the first section (section one in the test layout shown in Figure 3.14). To begin each run at the same location, silver tape was placed at the start of section one and an infrared red start sensor used to detect the tape. This works well for the TxDOT acoustic and laser rut systems; however, the actual position where measurements begin for the scanning laser will depend on the position of the rotating mirror. Thus, the starting distance for the scanner can vary. At the measurement speeds used, this distance will only be about one to three feet. Thus for the comparison, section locations for each device are selected based on their closest point to the actual measured locations.

The scanning laser was run at the four speeds: 30, 40, 50, and 60 MPH. Two runs at each of these speeds were made. The repeatable of the 30 and 60 MPH runs for the average readings for approximate one hundred foot sections for each wheel path are illustrated in Figure 3.23. The other runs indicated similar results.

The 30 and 60 MPH rut readings for each of the individual sections are compared with the static readings in Figure 3.24**.** As can be noted in this figure, the values measured by the scanning laser follow the static readings at each of the sections. The acoustic and scanning measurements were similar for the right wheel path, but a negative offset is noticed on the left wheel path. This can be caused by an incorrect offset value of the left acoustic during the calibration setup procedure and the values were replaced with zeros. A contour of the section area, sections one to four, is illustrated in Figure 3.26.



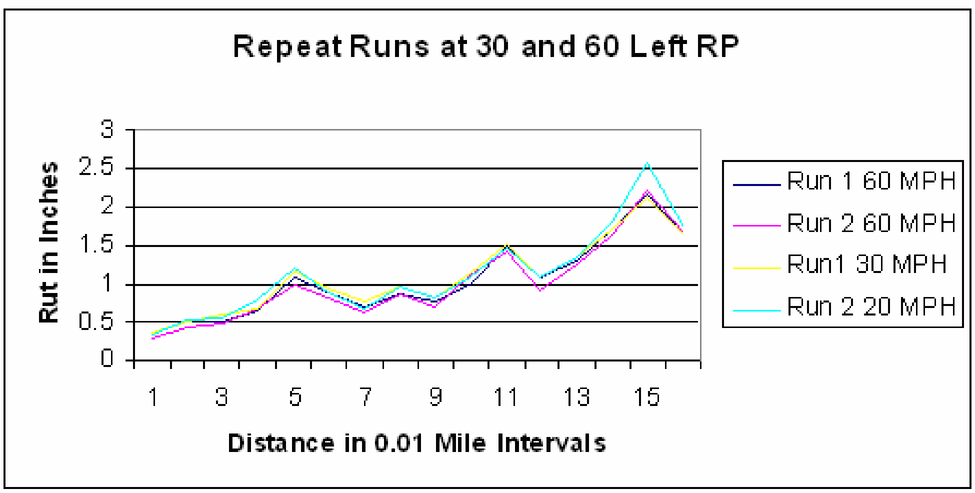
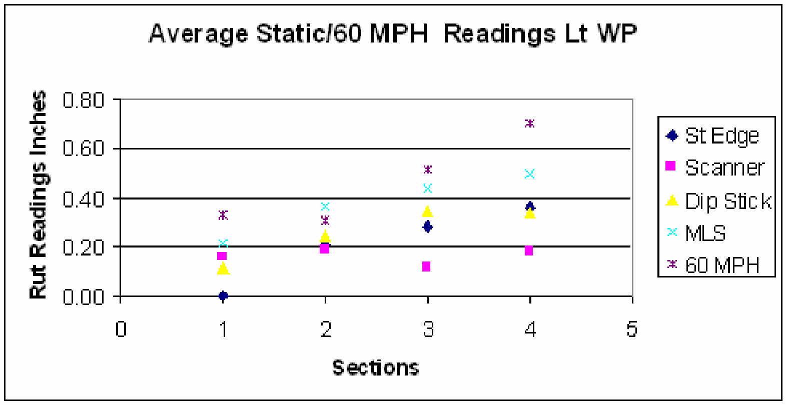


Figure 3.23 Repeat Runs Over Test Area Includes at 30 and 60 MPH



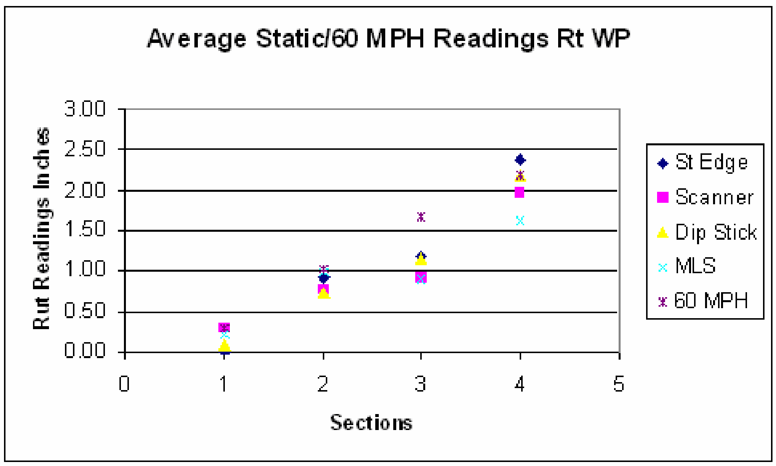
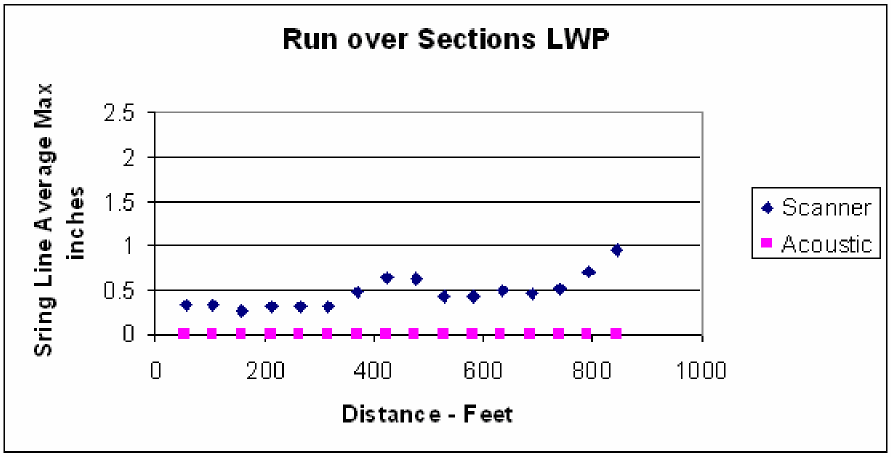


Figure 3.24 Comparisons of 60 MPH Scanning Laser Runs with Static Reading

60 MPH Run Over Sections Using Scanning Laser and Acoustic Rut System						
Section	Distance	Scanner Lt	Scanner Rt	Distance	Acoustic Lt*	Acoustic Rt
1	55.1	0.32	0.28	52.8	0	0.77
	105.8	0.3	0.43	105.6	0	0.6
	158.5	0.24	0.46	158.4	0	0.74
	211.5	0.28	0.68	211.2	0	0.22
2	264.2	0.24	1	264	0	0.93
	317.3	0.28	0.81	316.8	0	0.68
	370.4	0.49	0.63	369.6	0	1.23
	423.1	0.68	0.86	422.4	0	0.95
	476.2	0.6	0.7	475.2	0	0.72
	529.3	0.43	1.09	528	0	0.99
	582.4	0.44	1.4	580.8	0	1.44
	635.4	0.49	0.92	633.6	0	0.25
	688.1	0.46	1.26	686.4	0	1.56
3	739.2	0.54	1.65	739.2	0	1.65
4	792.3	0.71	2.21	792	0	1.93
	845	1.05	1.68	844.8	0	1.52

* There could have been a problem with the mounting corrections for the left measurements.

Table 3.6 60 MPH Run Over Sections Using Scanning Laser and Acoustic Rut System



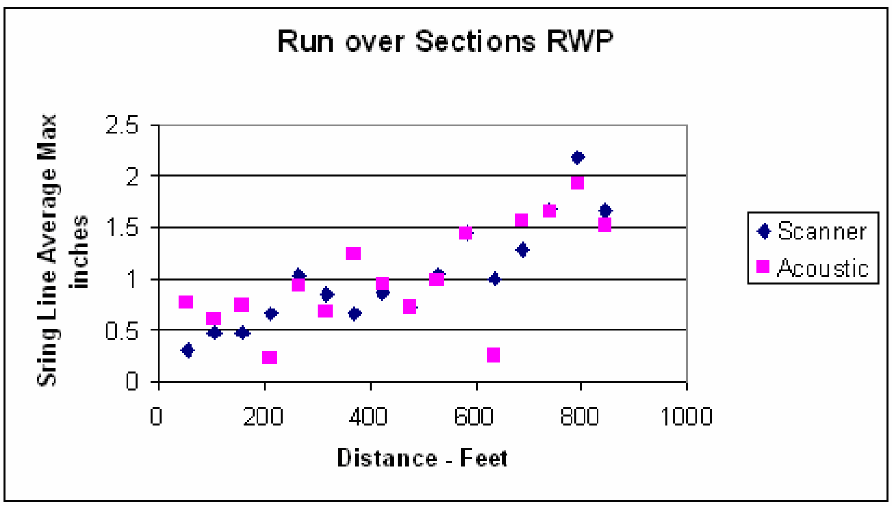


Figure 3.25 Run Over Sections

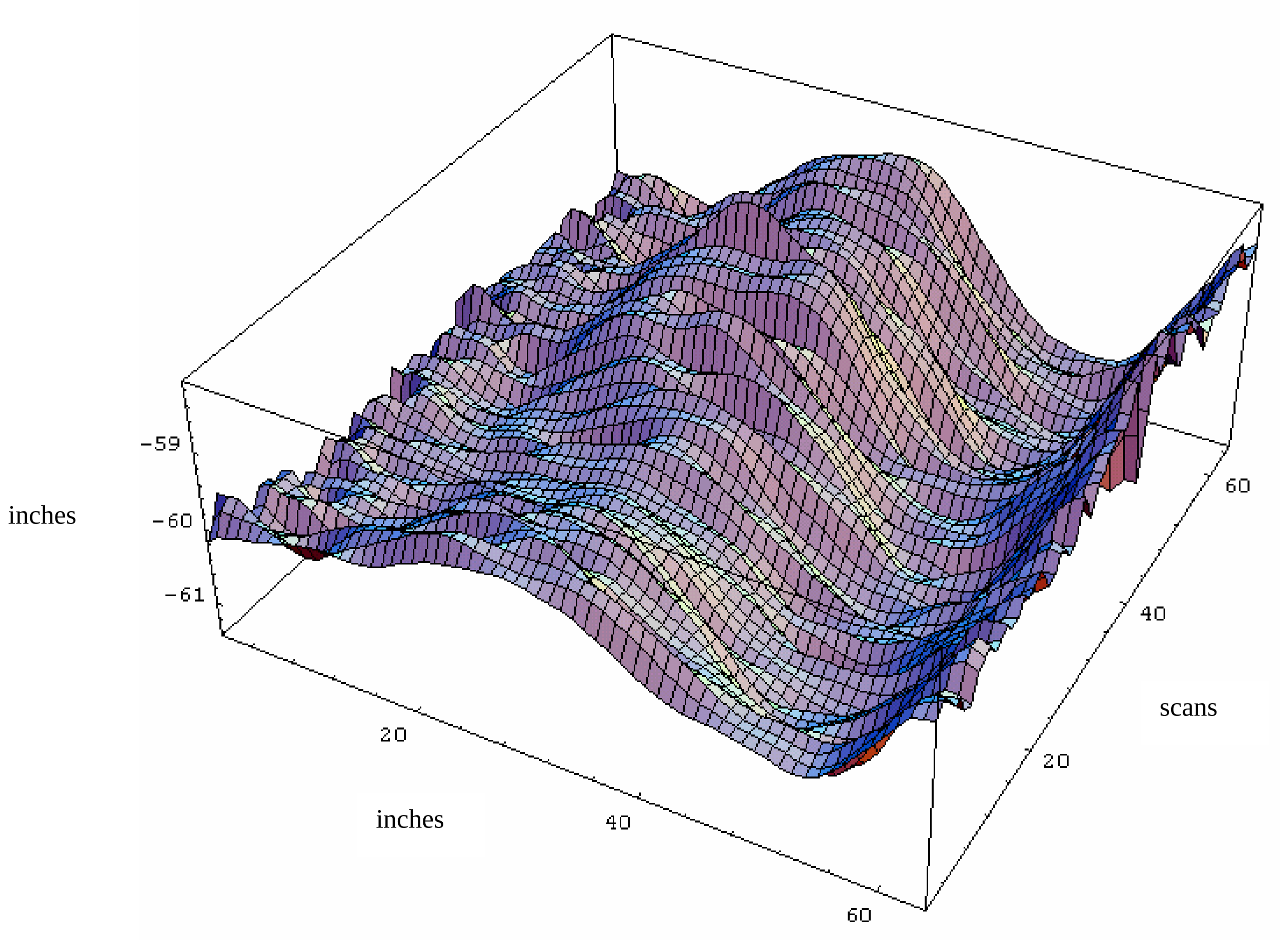


Figure 3.26 Contour Area of Sections 3 and 4 Taken With Scanning Laser at 60 MPH

As noted during the final year of the project, the processing method was changed to include both noise reduction fitting and the Bezier Curve to the resulting profile. The next chapter explains the recommended processing procedures.

Chapter 4 System Software Interface and Operations

4.1 Using The Scanner Program

The Scanner Program, named **RUTSCAN,** is written in the C Language using the Microsoft version 6.0 compiler and runs under the DOS operating system. Each copy of the Scanner Program is for a specific scanner system (i.e. RUT1, RUT2, …). This uniqueness is necessary because the program first reads an INI file which contains the file name of the look-up table for the Laser/HSI calibration. Next, the program reads in the location of the scanning mirror encoder index pulse in relation to the scanner mirror downward position (WindowCenter value). Finally, these programs subtract from the calibrated range values to compensate for the mirror-to-laser distance (MirrorOffset value).

Note: The Scanner-Laser HSL combination should not be interchanged, because these three elements are calibrated as a unit by the manufacturer.

The program contains certain default values that can be changed by parameters in the DOS level calling of the program (i.e. RUT1 <parm1> <parm2> ….). The parameters are:

#nn	Change the scanner Serial No. to nn (defaults to #1 for RUT1.EXE)
%Ddd	Change the DisplayWindow size to dd degrees (defaults to 90^o)
%SSS	CHANGE THE STOREWINDOW SIZE TO SS DEGREES (DEFAULTS TO 110^O)
Www	Change the SampleWidth to ww μ seconds (defaults to 30 μsec)
?	Display the parameter options

4.2 The Main Program

The main() program first sends initialization parameters to the AR4000-LIR laser and turns on the laser beam via the serial port COM1. The laser beam is invisible to the eye (780 nm) and can only be viewed with an Infrared Viewer.

Note: The laser beam is a Class IIIb laser and can cause eye damage with direct viewing.

The main program will display the following message:

```
Press  '(' to Change the Motor Speed
       'D' to Display the Range Scan
       'S' to Save the Range Scan to Disk
       ESC Twice to Exit
```

The program will continuously read the High Speed Interface FIFO until a key is pressed and update the display of the following information every second:

- The number of samples read per rotation of the scanning mirror
- The number of samples that will be displayed in the Display Window
- The High Speed Interface Variable Speed Drive value (63 maximum)
- The revolution speed of the scanning mirror in rpm

An error message will halt this display and prompt for a key press. The error is generally caused by the scanning mirror not rotating (check the +12VDC power connection to the motor).

While in this display mode, the following key presses produce the following actions:

- Pressing the '(' key will increment (63 max.) the Variable Speed Drive value and the ')' key will decrement the value.
- Pressing the 'D' key will call the DisplayScan() routine, which will display the laser calibrated range values on the computer display in real time. This routine will be described later.
- Pressing the 'S' key will call the StoreScan() routine in which the calibrated values are stored on a disk file. This routine will be described later.
- Pressing the ESC key twice will exit the program. When the program exits, the present values of WindowCenter and of MirrorOffset are stored in the INI file and the laser beam is turned off.

4.3 The DisplayScan() Routine

The DisplayScan() routine is used to observe the laser range values, to adjust the WindowCenter value so that the center of the laser scan will point in the right direction (i.e. perpendicular to the road surface), and to adjust the MirrorOffset value. The following information is displayed:

- A plot of the calibrated range output in a DisplayWindow size scan window. The X-axis represents the physical X coordinate
- One inch vertical gradients along the left edge of the screen to show the resolution of the plotted range signal
- The average range times 100 (R)
- The average amplitude of the reflected signal (A).
- The Temperature reading of the laser photodiode (T)
- The number of samples per scan displayed (L), as determined by the DisplayWindow value.
- A "SS" will be displayed if the plotted values have been processed by a "Spike Removal" algorithm
- The location of the center of the scan window in relation to the encoder index pulse expressed as a fraction (C)
- The MirrorOffset value in inches (M)
- The Distance Pulse count at the start of the scan (E)
- The Slope of the range display if the Automatic Window Centering mode is in operation (W)

Pressing the following keys will produce the actions:

- Pressing the 'A' key will switch to the plotting of the amplitude output of the laser. Switching on the amplitude plot will turn off the range plot. The peak amplitude value is displayed (PA). Press 'A' to switch back to the range plot.
- Pressing the 'X' key will switch the range output plot between the scanner output mode (polar mode) and the Cartesian mode (range multiplied by the Cosine of the mirror angle).

- Pressing the ‘S’ key will switch the range or signal amplitude plots to being or not being processed by a “Spike Removal” algorithm.
- Pressing the ‘P’ key will cause the plot to be paused (frozen). Press ‘P’ again to resume.
- Pressing the ‘W’ key will enable/disable the Automatic Window Centering mode. The slope of a Least Squares fit to the range data will be displayed as W=sss when the mode is enabled. The Automatic Window Centering mode will automatically adjust the WindowCenter value according to the value of the slope of the Least Squares fit (see the description of the ‘(‘ and ‘)‘ keys below).
- Pressing the ‘↑’ key will increase the range plotting vertical gain and the ‘↓’ key will decrease the gain.
- Pressing the ‘→’ key will increase the WindowCenter value and the ‘←’ key will decrease the value. Changing this value will cause the range display to rotate when plotting in the Cartesian mode. The scanner should be placed over a level surface and the WindowCenter value adjusted so that the range distance values of the scan are horizontal (equal values). The WindowCenter value will be stored in the INI file when the program exits in the outer loop. Pressing any of these two keys will disable the Automatic Window Centering mode.
- Pressing the ‘+’ key will increase the MirrorOffset value and the ‘–‘ key will decrease the value. The scanner should be placed over a level surface and the MirrorOffset value adjusted so that the average range value when plotting in the Cartesian mode is equal to the scanner box bottom height plus 3.0 inches above the surface. The MirrorOffset value will be stored in the INI file when the program exits in the outer loop.
- Press Esc to exit the ‘D’ Option.
- Press ‘?’ to display the key options – Press ‘P’ to resume.

4.4 The StoreScan() Routine

The StoreScan() routine is used to store the scanning laser output on a disk file. The data is stored in binary format, each rotation of the scanning mirror stored as follows:

- Each scan starts with four 2-byte integer values containing:

[0] The number of samples sampled per rotation of the mirror.

[1] The number of samples stored per rotation of the mirror (L), as determined by the StoreWindow value.

[2] The Distance Pulse count at the start of the scan.

[3] The SampleWidth of the laser in microseconds.

- The number of samples stored per rotation (L) 2-byte unsigned integer values that are the calibrated range values times 100.
- The number of samples stored per rotation (L) 1-byte unsigned integer values which are the amplitude values.

NOTE: IF THE CONFIGURATION HAS A MIRROR SPEED OF 2600 RPM, WITH A SAMPLE WIDTH OF 30 μ SECONDS, AND WITH A STORE WINDOW SIZE OF 110 ^{O}, THEN 769 SAMPLES ARE SAMPLED PER REVOLUTION. FOR THIS CASE, THE UNIT STORES 235 SAMPLES FOR EACH REVOLUTION(713 BYTES/REVOLUTION). IN THIS EXAMPLE, THE STORAGE MEDIA MUST OPERATE AT 30,897 BYTES/SECOND.

The program will first ask for the name of the disk file to store the scans. Any name starting with the character '#' will not be stored, as this is a test mode. Pressing the ESC key will exit the StoreScan() routine.

Next, the program will display the message:

```
Press Any Key to Start the Start Sequence
```

Pressing the ESC key will exit the StoreScan() routine. Otherwise, the Distance Count is zeroed and the following message is displayed:

Waiting For the Start Signal – Press 'S' to Start Override

Pressing the ESC key will exit the StoreScan() routine. Otherwise, grounding the "Start Detector" connector on the High Speed Interface Control, or pressing the 'S' key will start the storing of the scan information. The storing will continue until the ESC key is pressed.

After the storing of the data is terminated, the operator can display the data that is stored by pressing ‘Y’ to the question:

```
Do You Wish to Display the Stored Data?
```

The routine DisplayScanDisk() will display the range and/or amplitude data that is stored on the disk. Press ‘?’ to see the key options available.

4.5 How The Scanner Program Works

The routine **ReadToIndex()** is the basic routine that reads an array of sample data from the High Speed Interface FIFO until a mirror index pulse (Bit 0 of Byte 3) is read and returns the array, the number of samples in the array, and the number of samples read before the Sonic Range Enable signal goes from a logic 1 to a logic 0. The routine first waits for the FIFO to be half-full so that it does not have to worry about the FIFO becoming empty during a reading of one mirror rotation. All bytes are placed into the array except the Ambient Light and Encoder 2 values (6 bytes per sample are stored). This FIFO read operation is the limiting factor in the speed of operation of the program (limited by the 8 MHz speed of the ISA bus). The FIFO read uses approximately 40% of the total computer time.

The routine **GetLineScan()** uses the array from ReadToIndex() to form an array of samples that represents the portion (sector) of the mirror rotation that is directly below the scanner mirror. The WindowCenter value represents the center location of the sector in the array from ReadToIndex(). If the present array from ReadToIndex() does not contain enough samples for the sector, due to the location of the WindowCenter value, the next array from ReadToIndex() is used to complete the sector’s array.

The sector’s array is sent to the calibration routine **CalibrateRange()** where the 8-bit Signal Amplitude and 20-bit Raw Range values are used to index the calibration look-up tables. The Sensor Temperature and values determined by the SampleWidth and Maximum Range values are used to correct the look-up values.

The resulting calibrated range values are then displayed in CGA screen resolution in the routine **DisplayScan()** or stored on the specified disk file in the routine **StoreScan()**.

Other routines are:

- **SetupLookupTables()**, which reads in the calibration tables for the specific AR4000-LIR.
- **GetResolution(),** which calculates calibration parameters for CalibrateRange().
- **ReadHSI(),** which is the same as ReadToIndex() except the sample values are not stored.
- **CreateScanBuffer(),** which obtains dynamic storage for a Scan Buffer.
- **DeleteScanBuffer(),** which returns a Scan Buffer to dynamic storage.
- **ClipSpikes(),** which reduces the spikes in the range data that is caused by motion. It is a simplified version of the ClipSpikes() routine that is used in the Analysis Routine.
- **Reverse16(),** which reverses the order in an array.
- **LeastSquaresSlope()** returns the slope of the range data.
- **DisplayScanDisk(),** which displays a scan that is stored on a disk file.
- **ExitScanner(),** which turns off the scanner system.

4.6 Using ECP and DMA For Data Transmission

The program **DMASCAN** transmits the scanner data in the StoreScan routine to an external computer using the parallel port (LPT1:) operating in the Extended Capabilities Port (ECP) mode and using the Direct Memory Access (DMA) circuits. The interconnecting cable between the two computers is shown in Figure 4.1. The scanner ECP circuit places 8-bits of data on the D7-D0 data lines and sets the HostClk line low until the PeriphAck line from the receiver is set to high.

The program transfers the scan data to a buffer and initiates the DMA Demand Mode with Address Increment and Memory to I/O on DMA channel # 3. If the external computer has not completely read a scan when the next scan is ready for transmission (approximately 23 milli-seconds), the next scan is discarded and not transmitted.

	Pin #		Pin #	
Scanner (HostClk)	1	to	10	Receiver (PeriphClk)
Scanner (Data D0)	2	to	2	Receiver (Data D0)
Scanner (Data D1)	3	to	3	Receiver (Data D1)
Scanner (Data D2)	4	to	4	Receiver (Data D2)
Scanner (Data D3)	5	to	5	Receiver (Data D3)
Scanner (Data D4)	6	to	6	Receiver (Data D4)
Scanner (Data D5)	7	to	7	Receiver (Data D5)
Scanner (Data D6)	8	to	8	Receiver (Data D6)
Scanner (Data D7)	9	to	9	Receiver (Data D7)
Scanner (PeriphAck)	11	to	14	Receiver (HostAck)
Scanner (Gnd)	18-25	to	18-25	Receiver (Gnd)
Scanner (nAckReverse)	12		12	Receiver (nAckReverse)
	to		to	
Scanner (nReverseReq)	16		16	Receiver (nReverseReq)

Figure 4.1 ECP Interconnect Cable

4.7 How To Use The Scanner Data

The first thing to remember when using the scanner data is that it is in polar coordinate form (distance from the mirror origin and mirror angle). The distance output is stored as inches times 100 (0.00 to 655.35 inches). The angle increment is calculated from the number of samples per rotation of the mirror (first integer in a scan record) and it is assumed that the midpoint of the scan is directly below the scanner mirror. The angle of a sample is the sample displacement from the midpoint of the scan times the angle increment.

The first step towards analysis of the scanner data is to convert the polar coordinate data to

Cartesian coordinates. The sample's Y value can be expressed as the polar distance times the Cosine of the sample angle. The sample's X value can be expressed as the Y value times the Tangent of the sample angle.

If standard digital filtering techniques are to be used, the x-coordinate spacing between the samples must be equal. Therefore, the data must be decimated before the filtering is done.

Chapter 4 will describe the analysis method that was used in this project. Heuristic methods were used for the elimination of noise in the range signal and the detection of ruts in the data.

Chapter 5 Data Analysis Software

5.1 The Analysis Program

The Analysis Program (**FindRut()**) takes the raw scan data that was stored by the Save command in the Scanner Program and produces transverse profile data for each rotation of the scanner mirror and the "String Line" depth and transverse location for the maximum rut to the left and right of the scanner center position. The program goes through a multiple step process to produce these results that consists of the following steps for each scan. Each step will be further explained later in this Chapter.

Step 1. Coordinate Transformation - The raw range data is converted from the scanner polar coordinate form to Cartesian coordinate form. An "analysis window" subset of the "scanner window" data that was stored by the Scanner Program is used. This analysis window can be shifted left or right to compensate for any roll in the scanner data (data that was taken when the scanner was not aligned perpendicularly to the road surface). The default analysis window size is 90^{o} and the default scanner window size was 110^{o}.

Step 2. Histogram Clipping - A "Histogram Algorithm" is used to limit the spike values that are present in the raw range data. The spikes are caused by the reflection scattering of the laser beam during laser spot motion.

Step 3. Spike Suppression – A multi-step algorithm is used to suppress the remaining spikes in the raw data and produce a signal that follows the contours of the road surface.

Step 4. Scan Comparison - A "Best of NumScans" comparison is used to smooth the transition of succeeding scans as the scanner travels along the road and to throw out any individual bad samples of data from the scan.

Step 5. Curve Fitting - A "Bezier Curve" description of the resulting curve is formed to produce a mathematical description of the road surface contour that is continuous.

Step 6. Rut Detection - The maximum "String Line" rut depth and transverse location to the left and right of the scan center is computed.

The program has several run-time variables that have default values that can be changed by using parameters at the DOS level call of the program (i.e. FINDRUT <parm1> <parm2> ….). The variables are:

- The **ScanSize** value is the number of degrees of the scanner data that is to be analyzed. The default value is 90°. This value is overridden by using the parameter %xx at the DOS level call of the analysis program (xx is the number of degrees to use).
- The **NumScans** value is the number of scans that are used by the "Best of NumScans" routine to smooth the scan transition. The default value is 3. This value is overridden by using the parameter Sxx at the DOS level call of the analysis program (xx is the number of scans to use). This value is used extensively in the program.
- The **SpikeMax** value is the initial maximum spike value that is to be allowed in the spike suppression routine. The default value is 2.0. This value is overridden by using the parameter Mxxx at the DOS level call of the analysis program (xx is the initial maximum value to use).
- The **NumBezier** value is the number of Bezier points to produce in the Bezier Curve description. The default value is 64. This is overridden by using the parameter Bxx at the DOS level call of the analysis program (xx is the number of Bezier points to use).
- The **OffSet** value is a number that is subtracted from the range data from the scanner to compensate for the physical laser to mirror distance in the scanner. The default value is zero. This compensation is usually done in the scanner program. This value is overridden by using the parameter Oxx at the DOS level call of the analysis program (xx is the mirror OffSet value to use).
- The **FtPerPulse** value is the distance between "Distance Pulses" that TXDOT supplies to the scanner. The default value is 5280/13328. This value is overridden by using the parameter Dxxxx at the DOS level call of the analysis program (xxxx is the number of pulses per mile).
- The **ReportInterval** is the distance in feet between the recording of the average and maximum rut depths by the program. The default value is 52.8 feet. This value is overridden by using the parameter Rxx at the DOS level call of the analysis program (xx is the distance interval to use for reports).

- The **SingleStep** mode is entered by using the parameter S at the DOS level call of the analysis program. The default mode is continuous analysis with no graphics display and automatic window shifting. In single step mode, the graphics is displayed. Use the key press options that are described later in this chapter to switch display types, switch out of the single step mode, and turn the graphical display on or off.
- The "**?**" parameter (i.e. FINDRUT?) will display the available parameters that are available.

The data results from the various steps of data processing sequence are stored in a data structure called **scans.XY[ ]**, which contains (NumScans+4) address pointers to one-dimensional arrays. These arrays are dynamically allocated by the program. Each element of an array contains the (x,y) coordinate of a point on a curve. The data structure arrays are:

- **scans.XY[0]** is the String Line curve derived from scans.XY[1].
- **scans.XY[1]** is the Bezier Curve derived from scans.XY[2].
- **scans.XY[2]** is the Best of NumScans curve for the *"scan that is being analyzed*" that was derived from scans.XY[3] through scans.XY[NumScans+2].
- **scans.XY[3]** through **scans.XY[NumScans+2]** are the last NumScans scans that have been read and processed.
- **scans.XY[NumScans+3]** is the Backward Clip for the last scan read.

When the programs first starts, (NumScans-1) scans are read and processed through the Coordinate Transformation, Histogram Clip, and Spike Suppression algorithms only. Every scan is read into the location scans.XY[NumScans+2]. After a scan is processed, the stack consisting of scans.XY[3] through scans.XY[NumScans+2] is pushed so that it contains the last NumScans scans of data (if NumScans>1 then each scan is moved to the next lower index number position in the data structure and the scan in scans.XY[3] is lost). After NumScans scans have been read, the other steps are processed and the final results are obtained.

For example, if NumScans equals 3 and scan number 10 is the *"scan that is being analyzed*":

scans.XY[0] contains the String Line curve for scan number 10

scans.XY[1] contains the Bezier approximation for scan number 10

scans.XY[2] contains the "Best" of scan numbers 9, 10, and 11

scans.XY[3] contains scan number 9 after spike suppression processing

scans.XY[4] contains scan number 10 after spike suppression processing

scans.XY[5] contains scan number 11 after spike suppression processing

scans.XY[6] contains the backward clip of scan number 11

The pushdown stack consists of scans.XY[5] (top) through scans.XY[3]

All scans are read into scans.XY[5] (the top of the stack)

5.2 Coordinate Transformation

Each scan that was recorded by the Scanner Program contains a header which contains the number of samples in the 360^o rotation of the mirror from which the scan was derived and the number of samples that are stored for the scan. This header was described in Chapter 4 (The StoreScan() Routine).

The program first reads all of the scans in the file to determine the average angle increment (**dAngle**) of each sample (360^o divided by the average number of samples in a rotation of the mirror). From this angle increment, a table of trigonometric Cosine and Tangent values is formed for each sample in the analysis window. The range values of each sample in the analysis window are multiplied by the corresponding angle's Cosine value to convert the range value from polar to its Cartesian Y coordinate. The X coordinate is represented by the angle's Tangent value at the present time. The results are placed into the location scans.XY[NumScans+2] (See Figures 5.2 & 5.3).

If the analysis window size is smaller than the window size of the recorded data (i.e. 90^o vs. 110^o), the analysis window center can be shifted left or right by the operator, providing that sufficient sample data exists to the left or right of the analysis window. This shifting of the analysis window is done automatically if the system is in the W mode, which is described later. The algorithms that are used for developing the final results are not dependent upon the scan being perpendicular to the road plane but the scan should be approximately horizontal for the maximum coverage of the road lane. [The rolling of the scanner data can be caused by operator misalignment of the scanner program, motion of the scanner mount on the scanner vehicle, and the natural rolling of the vehicle (see Figure 5.1 for an example of vehicle roll).] Readings can also be adversely affected by vertical pitch if some type of body correction method is not employed.

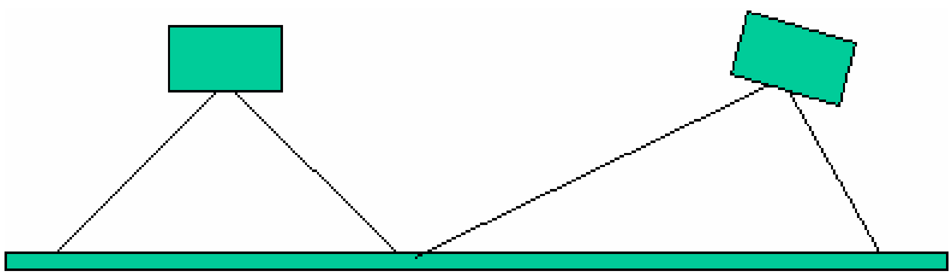


Figure 5.1 Effect of Vehicle Roll on the Area Covered by the Scanner

Figure 5.2 shows a typical polar coordinate scan of a road surface (the X coordinate represents the tangent of the angle). This view is scan #335 from a test run driven at 60 mph with the scanner mounted 5 feet above the road surface. Figure 5.3 shows the Cartesian coordinate equivalent. The large spikes in the data are caused by reflection scattering of the laser beam at those locations of the road surface. All range values are measured in inches from the scanner mirror and are negative in value.

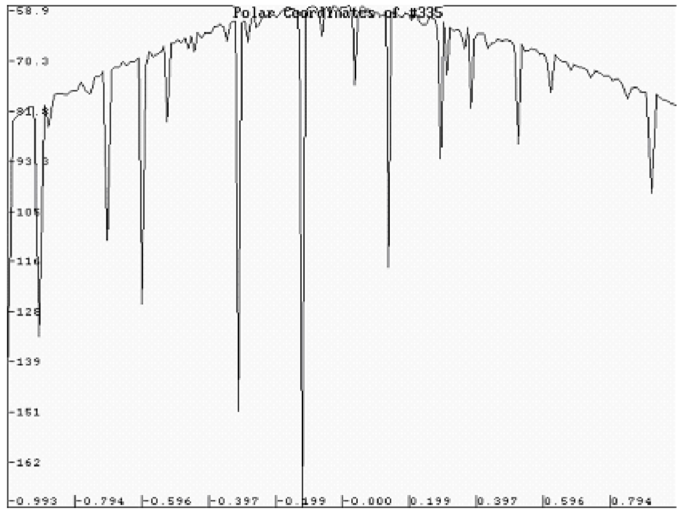


Figure 5.2 Polar Coordinates

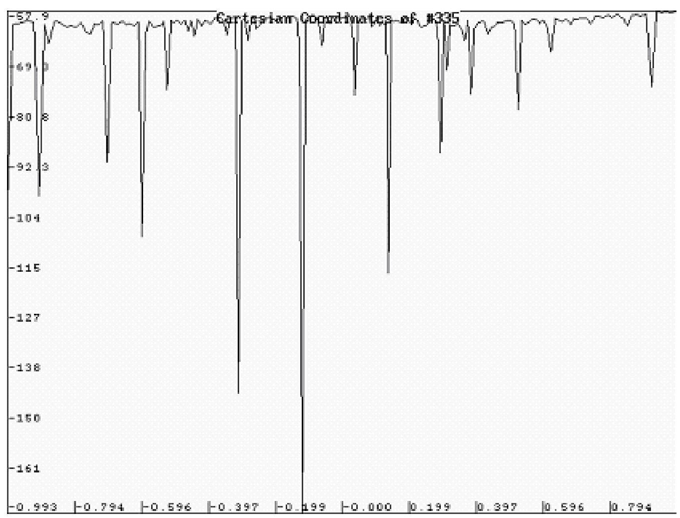


Figure 5.3 Cartesian Coordinates

5.3 Histogram Clipping

Histogram clipping (**HistoClip()**) is used to eliminate any large spikes in the data in the location scans.XY[NumScans+2]. As mentioned before, scattering of the laser beam reflection causes large variations in the range values. These large variations, called “spikes”, are evident to the human eye as variations to the relatively smooth surface of the target road surface.

A “Least Squares” linear line fit is made to the data and the histogram represents the deviation of the range data from the linear line (Y = slope * X + intercept). The first histogram span (range of histogram values calculated) is determined by the minimum and maximum deviation of the range values from the linear line fit.

The histogram for the data deviation is calculated (in terms of ten range bands) and the maximum value is found in the histogram. Any range value which produces a deviation that exceeds the boundaries of the histogram is clipped so that the deviation is within the histogram boundaries.

A new linear line fit of the clipped data is computed, the histogram span is set to the previous span divided by two and centered about the maximum location in the present histogram, and the procedure is repeated until the histogram span is less than 8.0 inches (an arbitrary value). The resulting range data contains spikes that are limited to ±4.0 inches

Figure 5.4 shows the result of the histogram clipping of Figure 5.3. Note that the spikes to –172 inches in Figure 5.3 have been reduced to –68 inches. It has been found that the results of this routine are accomplished by the next level of processing (Spike Suppression), so therefore this routine could be eliminated in the future.

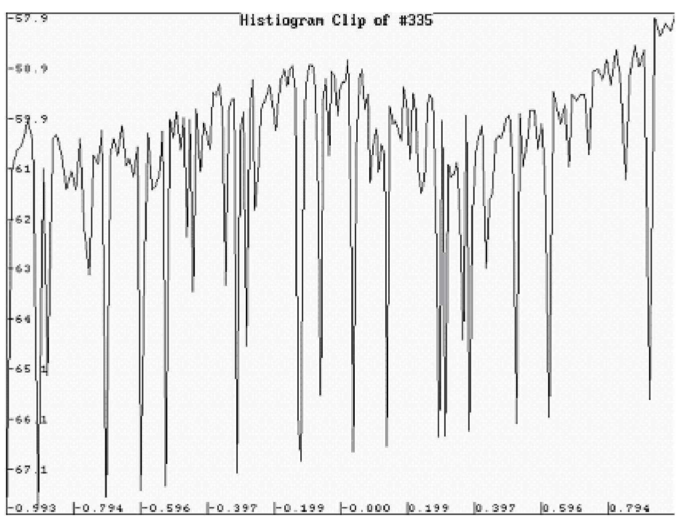


Figure 5.4 Histogram Clipping

5.4 Spike Suppression

Spike suppression is accomplished by executing the following routines four times, each time the maximum spike value that is allowed is reduced by a factor of two. The initial value of the maximum spike value allowed is SpikeMax.

- The routine **ClipSpikesForBack()** takes the data in the location scans.XY[NumScans+2] and makes a copy of it with the Y values only reversed in order into the location scans.XY[NumScans+3]. Since the X values are symmetrical, this reverses the scan from left to right to right to left with the X values unchanged except for their sign.

  Both forward and backward scans are processed through the Spike Suppression routine (**ClipSpikes()**). This routine will be fully described later. The backward result in the location scans.XY[NumScans+3] is then reversed. Now the Y values in both locations have the same order again.

  The reason for the forward and backward process is that the starting left-hand values in the scan have an effect on the final results. If large values of "noise" are present at the start of the scan, then the comparison with the backward scan analysis will detect an error in the results.

- Now the forward and backward results of the spike suppression are compared with the previously processed scan. If no previous scan exits, then the values in the location scans.XY[NumScans+3] are averaged with the values in the location scans.XY[NumScans+2] and the results placed into the location scans.XY[NumScans+2]. Otherwise the forward and backward scans are processed by the routine **CompareForBack()** to determine if an error exists in the process.

  In the **CompareForBack()** routine, for each corresponding sample (x_i,y_i) in the previous scan (scans.XY[NumScans+1]), forward scan (scans.XY[NumScans+2]), and backward scan (scans.XY[NumScans+3]), the absolute difference between the three scans is computed. The average of the two samples with the smallest difference between them is placed into the location scans.XY[NumScans+2]. This action will tend to throw out any bad samples that occur in the processed scans. The calculation will also produce a recursive feedback if the forward and backward scans do not agree.

The **ClipSpikes()** routine is a sequential routine that scans from left to right using a small clip-window of previously analyzed data to analyze the next sample in the scan. The clip-window distance is the result of the function of the number of points in the Bezier Curve and the number of points in the analysis window ((last x-value minus first x-value) divided by (NumBezier minus one)). This window represents a distance along the x-axis of the scan. The routine also uses a maximum spike value to allow (a function of SpikeMax).

Define the following values:

x_{cw} is the clip-window distance

y_{ms} is the maximum spike value

For the sample (x_i,y_i):

- Set y_{av} to the average of the y_k $(k < i)$ values over a x_{cw} distance of the samples previous to x_i:

  $$y_{av} = \text{Average } y_k \text{ for } (k < i) \text{ and } ((x_i - x_k) \leq x_{cw})$$

  If the value of x_i is not large enough to form a clip-window to the left, then set y_{av} to the value of y at x_i using a Least Squares linear line fit of the first one-third of the data.

- If the absolute deviation of y_i from y_{av} exceeds the maximum spike value $(|y_i - y_{av}| > y_{ms})$, then the program looks to the right of x_i to see if the value of y deviation from y_{av} comes to within one-half of the maximum spike value within two times the x_{cw} distance to the right of x_i:

  if $(y_i > (y_{av} + y_{ms}))$ then

  if $(y_k \leq (y_{av} + y_{ms}/2))$ for $(k > i)$ and $((x_k - x_i) \leq (2 \times x_{cw}))$

  if $(y_i < (y_{av} - y_{ms}))$ then

  if $(y_k \geq (y_{av} - y_{ms}/2))$ for $(k > i)$ and $((x_k - x_i) \leq (2 \times x_{cw}))$

  If so, or if the scan ends within two times the clip-window distance to the right of x_i, then y_i is set to the value of y_{av}. Otherwise, the value at y_i is limited to the one-half of the spike value deviation from y_{av}.

The scan in the location scans.XY[NumScans+2] is processed through the **ClipSpikesForBack() - ClipSpikes() - CompareForBack()** sequence four times, each time the value of the maximum spike is reduced by one-half. The initial value of the maximum spike is the value of SpikeMax. Therefore, the spikes are limited to ±SpikeMax/16 inches from the road contour after the processing.

Figure 5.5 shows the results of four passes through the spike suppression algorithm on the results of Figure 5.4 using a SpikeMax value of 2. Note that the contour of the road surface is now obvious with noise values limited to ±0.125 inches.

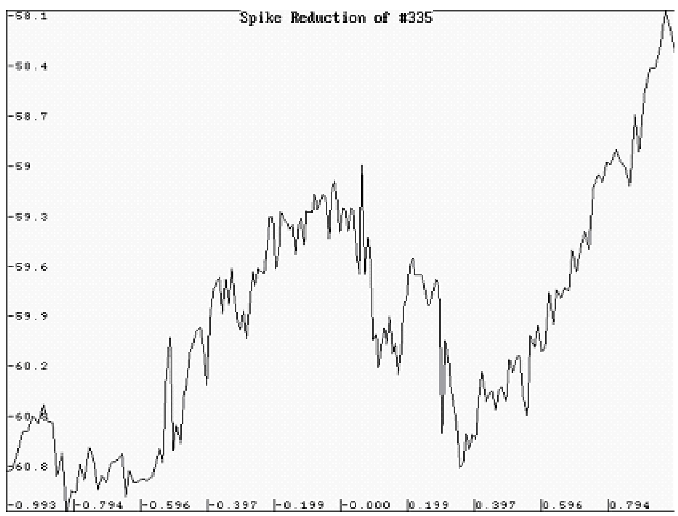


Figure 5.5 Spike Suppression

5.5 Scan Comparison

If NumScans equals one, then the values in the location scans.XY[3] are copied to the location scans.XY[2] and the "*scan that is being analyzed*" is the last scan that was processed. Otherwise the adjacent scans to the "*scan that is being analyzed*" (a scan that is located somewhere between scans.XY[3] and scans.XY[NumScans+2]) are used to smooth the transition between the scans and to throw out any bad (x_i,y_i) points that occur after the previous processing. This produces an averaging low pass filter in the direction of travel of the scanner.

If NumScans is an odd number greater than one, then each (x_i,y_i) point in the "*scan that is being analyzed*" is compared to the corresponding points of floor(NumScans/2) scans preceding and following the scan. For example, if NumScans is an even number, then each (x_i,y_i) point in the "*scan that is being analyzed*" is compared to the corresponding points of (NumScans/2) scans preceding the current scan and the (NumScans/2-1) scans following the current scan.

The **Best()** routine makes a list of the corresponding (x_i,y_i) value from each of the NumScans scans and finds the sample that produces the largest absolute difference between it and the other samples. This sample is removed from the list of samples and the process continues until only two samples remain in the list. The (x_i,y_i) sample in the location scans.XY[2] is set to the average of these two remaining samples.

5.6 Curve Fitting

A Bezier Curve is an approximating curve that is defined by a set of control points. The samples of the curve that is located in scans.XY[2] are the control points. The curve attaches to the two control end points with a slope that coincides with a straight line from the end point to its adjacent point. Like the General B-spline curve, the location of a point on the curve is represented as a linear combination of polynomial functions called Bernstein polynomials. However, unlike the B-spline, which is used to generate points between two control points, the Bezier Curve uses n+1 control points to generate points between the two end points of the curve. For a given point on the Bezier curve, the control points that are closest to the point will have the strongest influence on the curve, but all points between the two end points will have some influence.

The parametric variable u ($0 \leq u \leq 1$) is the independent variable for the curve. The end points are represented by u=0 and u=1. For n+1 control points, the i^{th} Bernstein polynomial for the point at u is:

$$b_i(u) = \frac{n}{i!*(n-i)!} * u^i * (1-u)^{n-i} \quad \text{for } i = 0, 1, \ldots, n$$

With n+1 control points $\{(x_0,y_0),(x_1,y_1),\ldots,(x_n,y_n)\}$, the (x,y) location at point u is:

$$x(u) = \sum_{i=0}^{n}(x_i * b_i(u)),\ (0 \le u \le 1) \qquad y(u) = \sum_{i=0}^{n}(y_i * b_i(u)),\ (0 \le u \le 1)$$

The values in the location scans.XY[2] are used as control points to form a Bezier Curve with NumBezier. NumBezier points into the location scans.XY[1] using the routine **BezierCurve()**, which uses the routine **BezierPoint()**. The result is a smooth continuous curve, which can be used to produce a 3-D plot of the road surface. In the interest of computational speed, a table of logarithm of combination values, ln(n!/(i!×(n-i)!)), is precomputed by the routine **MakeCombinations()** (n is the number of control points). The Bernstein polynomials ($b_i(u)$) are computed by using logrithmetic arithmetic.

Figure 5.6 shows the results of spike reduction for scans number 333, 334, & 335. Figure 5.7 shows the "Best of NumScans" result and the Bezier approximation for scan #334. Note that the x-coordinate still represents the angle tangent values. Note also that the value of scan #334 in Figure 5.6 deviates from the other 2 scans in the left half of the plot. The "Best of 3 Scans" shown in Figure 5.7 uses the average of scans #333 and #335 in these locations.

The range values in the Bezier Curve are now multiplied to the normalized x-values (tangent values) to obtain the true transverse x measurement of the road surface. [Note that this transverse measurement is in reality along a diagonal whose length is determined by the vehicle speed (at 2600 rpm mirror speed and 60 mph, the scans are 2 feet apart and each scan travels ½ foot along the road).] A "Least Squares" linear line fit is made to the Bezier Curve to obtain the slope and intercept, which represents the "tilt" of the analysis window and a mean value of the road distance from the scanner. The Bezier Curve is rotated by this slope value about the intercept point so that it now has zero slope.

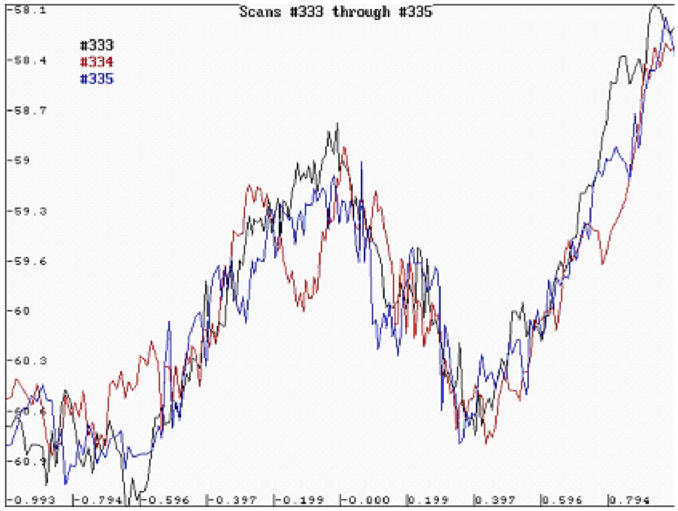


Figure 5.6 Results from NumScans=3 scans

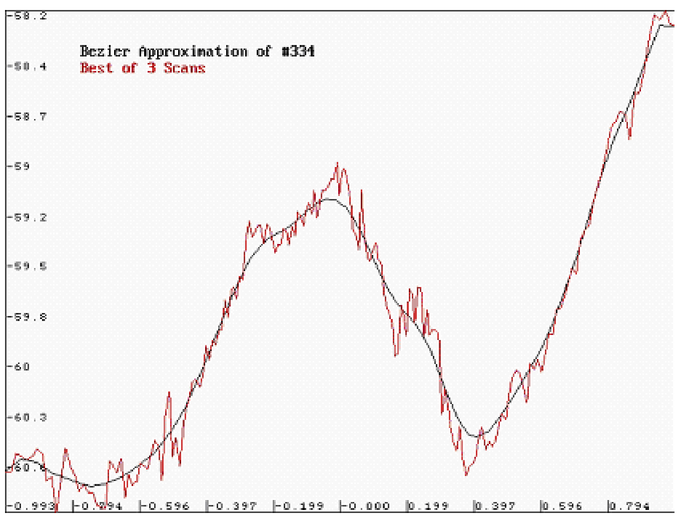


Figure 5.7 "Best of NumScans" and Bezier Approximation

If $|\text{atan}^{-1}(\text{slope})|$ of the linear line fit exceeds twice of the angle increment (dAngle), then the analysis window center is shifted left or right one dAngle value if the automatic window shifting mode is in effect. This shifting provides some compensation for vehicle and scanner mount roll.

5.7 Rut Detection

The "String Line" method of describing the amount of rutting in a road surface uses a straight edge between the high points in the road surface and measures the maximum depth between the straight edge and the bottom of the rut. The straight edge forms a "Convex Hull" which spans the high points along a curve.

Given a curve defined by n ordered points $\{(x_1,y_1),\ldots,(x_n,y_n)\}$, with the x's being in an increasing order, the "Graham-Scan Algorithm" produces an ordered set of points $\{(x',y')\}$ which define the Convex Hull of the curve.

The algorithm starts with an empty stack with (top) denoting the coordinate at the top location of the stack and (top-1) denoting the coordinate at the next to top location of the stack.

```
; Push the first 2 elements of {(x,y)} into the stack
        push((x1,y1)) , push((x2,y2)) ;
; Check the next element of {(x,y)}
        for i=2 to n do
        ;While the angle from (top-1) to (xi,yi) is greater then
        ;  the angle from (top) to (xi,yi)
                while angle((top-1) to (xi,yi)) ≥ angle((top) to (xi,yi)) do
                        pop     ; Pop the stack
                end while ;
                push((xi,yi))   ; Push (xi,yi) into the stack
        end for ;
```

The stack contains the Convex Hull coordinates at the end of the algorithm. The routine **ConvexHull()** takes the set of points that describe the Bezier Curve in the location scans.XY[1] and produces the Convex Hull coordinates in the location scans.XY[0].

The routine **StringLine()** first locates the highest point in the center of the road lane using the Bezier Curve data in scans.XY[1] and the routine **FindCenter()**. The **ConvexHull()** routine then forms the Convex Hull description of the left and right wheel paths in scans.XY[0].

For each wheel path of the String Line data in the location scans.XY[0], each segment of the String Line is used to form the line equation: $Ax + By + C = 0$

Then for every point (x_i, y_i) in the Bezier Curve in the location scans.XY[1] that is between the segment endpoints, the distance from the String Line segment to the Bezier Curve point is:

$$(A\, x_i + B\, y_i + C) / \mathrm{sqrt}(A^2 + B^2)$$

The maximum distance (rut depth) for each wheel path is found using the routine **MaxDistance()**.

Figure 5.8 shows the String Line of the road surface. Note that the x-coordinate now represents the true transverse road measurement (tangent of the angle times the distance value). The location that the routine FindCenter() found is marked with the vertical tick mark in the String Line curve.

The scans.XY[ ] data structure now contains:

scans.XY[0] contains the String Line curve for scan #334

scans.XY[1] contains the Bezier approximation for scan #334

scans.XY[2] contains the best of 3 scans for scans #333, #334, and #335

scans.XY[3] contains the processed scan #333

scans.XY[4] contains the processed scan #334

scans.XY[5] contains the processed scan #335

scans.XY[6] contains the backward clipped scan #335

The stack is now "pushed" (scans.XY[4] → scans.XY[3] and scans.XY[5] → scans.XY[4]) and the next scan (#336) is processed in scans.XY[5].

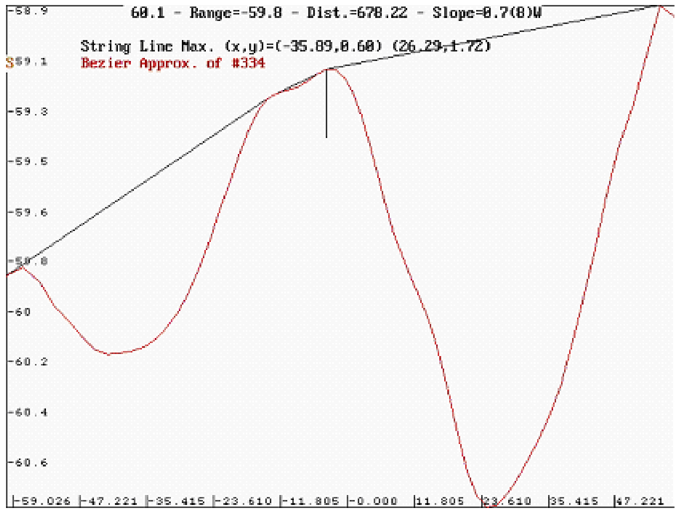


Figure 5.8 String Line

Other routines that are used in the FindRut( ) program are:

- **CreateAnalysisArrays()** which gets the dynamic storage for the scan data structure
- **DeleteAnalysisArrays()** which returns the dynamic storage
- **ReverseListY()** which reverses the order of the Y values in a scan
- **Rotate()** which does a 2-D rotation of a scan
- **LeastSquare()** which computes the slope and intercept of a scan
- **PlotXY()** which plots the scans on the display

5.8 Data Report Output

The output of the Findrut() program is two files:

- An ASCII file containing the (X,Y) coordinates of the contour of the road profile. The first record contains the number of samples per scan (NumBezier). This first record is followed by NumBezier records for each scan that is recorded. Each of these records contains the X and Y coordinates of a sample in the scan (ordered left to right).
- An ASCII file containing the following information per record for each report interval that is specified by the value of ReportInterval:
    a. The distance value (Dist) calculated from the Distance Pulse in feet.
    b. The average X location (LavX) and rut depth (LavY) of the left wheel path in the report interval in inches.

c. The X location (LmxX) of the maximum left wheel path rut depth (LmxY) in the report interval in inches.

d. The average X location (RavX) and rut depth (RavY) of the right wheel path in the report interval in inches.

e. The X location (RmxX) of the maximum right wheel path rut depth (RmxY) in the report interval in inches.

Figure 5.9 shows the output records of the maximum rut depth information from the program for the 60 mph section shown in the figures..

Dist	LavX	Lav Y	LmxX	LmxY	Rav X	Rav Y	RmxX	RmxY
55.1	-30.63	0.32	-21.18	0.59	33.48	0.32	19.54	0.74
105.8	-36.59	0.30	-33.87	0.51	25.58	0.52	24.79	0.93
158.5	-42.42	0.24	-50.13	0.59	13.30	0.49	21.29	0.94
211.5	-41.79	0.28	-49.67	0.63	17.92	0.65	14.72	1.13
264.2	-42.18	0.24	-45.11	0.37	17.11	1.07	16.37	1.48
317.3	-46.24	0.28	-42.00	0.58	17.35	0.89	19.81	1.34
370.4	-39.26	0.49	-40.03	0.89	21.27	0.70	17.98	1.18
423.1	-36.66	0.68	-36.00	0.98	21.70	0.86	28.33	1.22
476.2	-37.34	0.60	-39.79	0.90	20.22	0.75	8.25	1.23
529.3	-39.99	0.43	-35.61	0.60	26.09	0.99	18.07	1.22
582.4	-37.93	0.44	-35.67	0.74	26.25	1.49	21.39	2.11
635.4	-35.47	0.49	-34.11	0.94	28.00	1.07	30.04	1.55
688.1	-35.31	0.46	-36.08	0.71	27.33	1.31	28.39	1.92
739.2	-33.98	0.54	-30.12	1.14	27.31	1.70	28.11	2.20
792.3	-33.10	0.71	-34.08	1.42	28.76	2.16	28.33	2.46
845.0	-33.77	1.05	-38.54	1.73	31.09	1.65	28.25	2.30
898.1	-33.41	0.79	-32.50	1.19	27.44	1.52	26.53	1.84
950.8	-30.00	0.65	-20.92	1.01	26.75	0.87	22.82	1.81

Figure 5.9 Rut depth output of the FindRut( ) program

5.9 Using The Analysis Program

The program **FindRut()** is written in the C Language using the Microsoft version. 6.0 compiler. It runs under the DOS operating system or can be used with the DOS prompt of Windows. The program first asks for the directory in which the output of the scanner program is stored. A continuous loop is then entered from which the ESC key is used to exit the program.

A list of the files in the specified directory is displayed. The name of the scanner output file is requested. The name of the output Rut Depth and Road Profile files are requested. Pressing the ENTER key only will signify that no output file is to be produced.

The program will scan the scanner output file and report:

- The number of scans in the file
- The average number of samples per mirror rotation
- The speed of the mirror rotation in milli-seconds/Rotation
- The number of samples stored per rotation in the data
- The number of samples per rotation that will be used in the analysis

The program requests the first and last scan number that is to be analyzed. These values may be modified, as determined by the NumScans value and the number of scans that are available. The program now proceeds to display the analysis in the SingleStep mode if the S parameter was used in the DOS routine call to the program. Otherwise, the program will proceed in the non-graphics display mode until all scans have been processed. Press the 'S' key to get in or out of the SingleStep mode. The program will continue the analysis until the last specified scan number is reached or the operator presses the ESC key.

The program has two different display modes. Pressing the 'N' key will switch between each display mode:

- The processed scans that are stored in the locations scans.XY[3] through scans.XY[NumScans+2] and the "Best of NumScans" scan stored in the location scans.XY[2] are displayed.
- The Bezier Curve and String Line are displayed along with the location (X) and depth (Y) of the maximum rut depth for the left and right wheel path. Figure 5.8 is an example of this display mode.

Both modes display the file name, average range value of the scan, distance value that is calculated from the "Distance Pulse", and slope of the scan.

Pressing the following keys will produce the actions:

- Pressing the 'S' key will switch on/off the "Single Step" mode of analysis. Press any other key to continue is the "Single Step" mode.
- Pressing the 'N' key will switch between the two different plot modes.

- Pressing the ‘P’ key will switch the display pallet colors.
- Pressing the ‘D’ key will switch on/off the graphical display. When the display is turned off, only the scan number and distance value are displayed.
- Pressing the ← or → key will shift the “analysis window” in the “scanner window” left or right one sample. The number of samples shifted is displayed in the ( ) following the Slope information in the display.
- Pressing the ‘W’ key will switch on/off the automatic window shift mode. The program will automatically shift the “analysis window” left or right to provide a level scan when this mode is on. A ‘W’ symbol will be displayed when this mode is on.
- Press the + or - key to increase or decrease the mirror OffSet value.
- Press the ESC key to terminate the analysis.
- Press '?' to display the key options – Press any key to continue.

Chapter 6 Summary and Conclusions

6.1 Summary

This report discusses a project between the Texas Department of Transportation and The University of Texas at Arlington to investigate the uses of a scanning laser system for measurement of rut. The scanning laser, if operating properly and placed on a vehicle as it travels along the road provides longitudinal and transverse profile measurements of the road. The transverse profile would provide a means of measuring rut. Rut is currently measured using acoustic sensors mounted on the front of the vehicle in a specially designed TxDOT rut bar. Although this method of rut measurements has been used in Texas for a number of years, several problems affect the accuracy and amount of rutting that is measured. The use of the scanning laser for this application could potentially alleviate many of the problems.

During the project, a functional system has been developed based around the AccuRange 4000 scanning laser, which is manufactured by Acuity Research Incorporated. The system is used to scan the full width of the travel lane so that processing programs can use the data from the scan to report and store the rut condition of the pavement for each wheel path. A number of problems occurred in developing the system and are described in this report. The laser sensor, in a stationary configuration, met the technical specifications as indicated by the manufacturer. However, when scanning the laser across a pavement, noise spikes much greater than the transverse profile signal occur. Thus, much of the project research effort was focused on developing a means to distinguish the signal from the noise, and to address a means by which measurements could be done at highway speeds.

A method was developed to measure and report the deepest rut per scan. The project included three programs for both collecting and processing the transverse profile readings for computing rut. The source and execute modules of these programs were provided to TxDOT for implementation.

6.2 Recommendations and Conclusions

The project has provided a prototype scanning laser system for rut measurements and a project report to TxDOT for implementation. A string line algorithm applied to the transverse profile measurements provides a means to quantify rut. The accuracy of the system for measuring rut is not only a function of the laser, but is also directly related to the length of the scan, the rotational speed of the scanning mirror, and the smoothing of the Scanning laser data. Figure 6.1 illustrates the result of applying a Bezier Curve on the accuracy of the system and the differences between the different scan lengths. The figure provides a comparison between the MLS, Dipstick and Scanning Laser after manual alignment for Section 4 of the second scan. The standard error of difference between the Dip Stick and Scanning Laser (0.225 inch) could be in a large part due to the effect of the Bezier Curve on the shape of the scanning laser measurement.

.

Other items that directly affect measurement accuracy include the errors in distance measurements as caused by the different signal return intensity. Filtering methods are required in both the transverse and longitudinal direction. If too many points are included in the longitudinal filtering process, then the vehicle's suspension system characteristics may affect the results. The measurement process developed includes first using the scanning laser to acquire the transverse data scans, and then to store these values for post processing. As such, the uses of the system are probably only suited for project level applications. Real-Time network level measurements should be possible by minor modifications to the program and the use of a high-speed processor.

The system is ready for implementation at the project level. Additional studies are needed to investigate the use of the system for other applications, such as estimating cut and fill design requirements for new pavements or improving maintenance on existing pavements as discussed above.

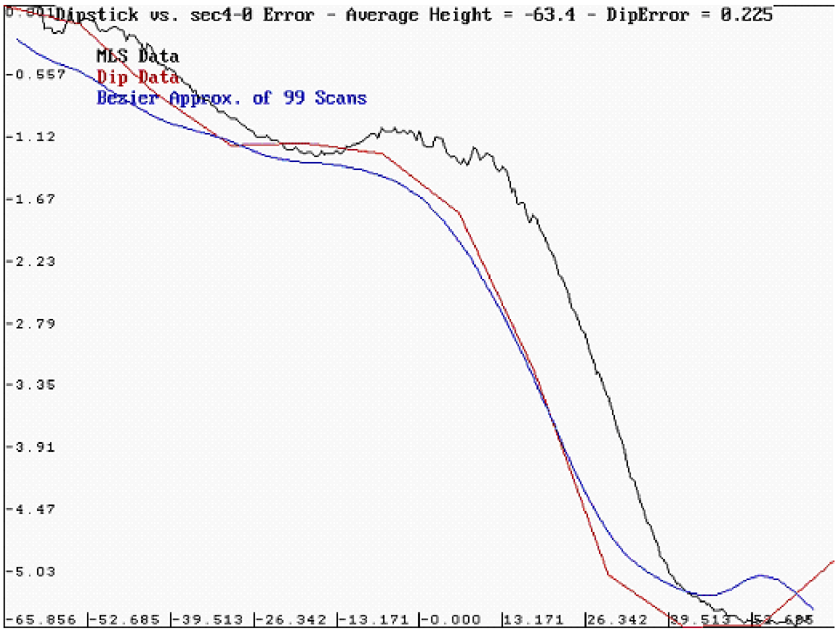


Figure 6.1 Comparison of MLS, Dipstick and Scanning Laser After Manual Alignment for Section 3, Second Scan

REFERENCES

1. Computer Graphics: The Principle Behind the Art and Science, Pokorny and Gerald, 2. Franklin, Beedle and Associates 1989, Section 8.3 Bezier Curves pp. 265-271

2.AccuRange 4000 Technical Reference, Acuity Research, 3475 Edison Way, Menlo Park Ca.,accuityresearch.com

3. Signal Processing Algorithms in Matlab, Samuel D. Stearns and Ruth A. David, Prentice Hall, 1996