MAPPING AND DETERMINATION OF SPATIAL VARIABILITY OF WINTER WHEAT AND TURFGRASS USING HIGH-RESOLUTION SENSOR DATA

By

TRAVIS S. TSUNEMORI

Bachelor of Science

Oklahoma State University

Stillwater, Oklahoma

1997

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
December, 2000

MAPPING AND DETERMINATION OF SPATIAL

VARIABILITY OF WINTER WHEAT

AND TURFGRASS USING

HIGH-RESOLUTION

SENSOR DATA

Thesis Approved:

John B. Solie

Thesis Advisor

Gregory E. Bell

Alfred Carlozzi

Dean of the Graduate College

ACKNOWLEDGMENTS

Several members of the faculty and staff of Biosystems and Agricultural Engineering, Plant and Soil Science, and Horticulture and Landscape Architecture aided me in my research. In addition to my committee members, special thanks go to Wayne Kiner, the Biosystems and Agricultural Engineering Laboratory Manager, and the rest of the BAE Lab employees for their much-needed help in construction and repair of research equipment. Also, thanks to Keith Reed of the Oklahoma State Turf Research Center for providing a place to test research equipment. The Samuel Roberts Noble Foundation, John Deere and Co., Oklahoma State University Agricultural Experiment Station, and the Soil Fertility and Education Advisory Board were financially supportive of this project. My fellow graduate and undergraduate students aided in the collection of data, both with fabrication skills and efforts in the field. Their contributions were invaluable, and their involvement in the research process is greatly appreciated. Also thanks to the secretaries in Biosystems and Agricultural Engineering for help with paperwork and meeting deadlines, and for their tremendous support.

Table of Contents

Chapter I: Introduction 1

Chapter II: Field-of-View for Mapping Optically Sensed Data in Winter Wheat 2

- Abstract 2
- Introduction 2
- Objective 3
- Literature Review and Semivariance Analysis Background 3
- Discussion of Sensor Output 9
- Methods of Data Processing and Analysis 12
- Results and Analysis 18
- Summary 24

Chapter III: Use of High-Resolution Optical Sensor Data for Mapping of Golf Course Fairways 25

- Objective 25
- Methods of Data Processing and Analysis 25
- Results and Analysis 27
- Summary 35

Chapter IV: Overall Summary and Suggestions for Future Research 36

- Overall Summary 36
- Suggestions for Future Research 36

BIBLIOGRAPHY 39

Appendix A: Semivariograms 41

Appendix B: Surface Maps 53

Appendix C: Procedure for Taking and Processing Data 65

- Data Acquisition 65
- File Analysis 67

Appendix D: Visual Basic Programs for Post Processing of Data 72

- 4100 for grass 1.exe 72
- 4100 for grass 2.exe 78
- 4100 for grass 3.exe 88
- matrixconv.bas 90
- semivariance.bas 91

List of Tables

Table 1: Error Calculation for Positioning 15

Table 2: Relevant Semivariance Statistics for Winter Wheat Plots 19

Table 3: Relevant Semivariance Statistics for Golf Course Fairways Scanned 28

List of Figures

Figure 1: Example semivariogram with Sill, Range, and Integral Scale labeled and the indication of a possible gradient noted 6

Figure 2: Sensor and antenna location on vehicle 11

Figure 3: Dimensions of sensor spacing and field of view 11

Figure 4: Lake Carl Blackwell Plot, Stillwater, OK, February 10, 2000, North-South Semivariance 20

Figure 5: Map of Lake Carl Blackwell Plot, February 10, 2000 21

Figure 6: Lucien Field, Flat Plot, February 29, 2000, East-West Semivariance 22

Figure 7: Lake Carl Blackwell Plot, Stillwater, OK, February 10, 2000 East-West Semivariance 23

Figure 8: Map of Fairway 12 with semivariance sections indicated 26

Figure 9: Map of Fairway 13 with semivariance sections indicated 27

Figure 10: Fairway 13, Middle Section, Karsten Creek Golf Course, June 13, 2000, North-South Semivariance 29

Figure 11: Fairway 13, Southeast Section, Karsten Creek Golf Course, June 13, 2000, North-South Semivariance 30

Figure 12: Fairway 12, North Section, Karsten Creek Golf Course, June 20, 2000, North-South Semivariance 31

Figure 13: Fairway 12, North Section, Karsten Creek Golf Course, June 20, 2000, East-West Semivariance 31

Figure 14: Fairway 13, Karsten Creek Golf Course, Stillwater, OK, Scanned on June 13, 2000 33

Figure 15: Fairway 12, Karsten Creek Golf Course, Stillwater, OK, Scanned on June 20, 2000 34

Figure A1: Perkins, OK, January 21, 2000 East-West 41

Figure A2: Perkins, OK, January 21, 2000 North-South 41

Figure A3: Perkins, OK, February 8, 2000, East-West 42

Figure A4: Perkins, OK, February 8, 2000, North-South 42

Figure A5: Perkins, OK, February 29, 2000, East-West 43

Figure A6: Perkins, OK, February 29, 2000, North-South 43

Figure A7: Terrace Plot, Lucien, OK, February 29, 2000, East-West 44

Figure A8: Terrace Plot, Lucien, OK, February 29, 2000, North-South 44

Figure A9: Flat Plot, Lucien, OK, February 29, 2000, East-West 45

Figure A10: Flat Plot, Lucien, OK, February 29, 2000, North-South 45

Figure A11: Lake Carl Blackwell Plot, Stillwater, OK, February 10, 2000, East-West . 46

Figure A12: Lake Carl Blackwell Plot, Stillwater, OK, February 10, 2000, North-South 46

Figure A13: Lake Carl Blackwell Plot, Stillwater, OK, February 29, 2000, East-West 47

Figure A14: Lake Carl Blackwell Plot, Stillwater, OK, February 29, 2000, North-South 47

Figure A15: Karsten Creek Fairway 13, Northwest Section, Stillwater, OK, June 13, 2000, East-West 48

Figure A16: Karsten Creek Fairway 13, Northwest Section, Stillwater, OK, June 13, 2000, North-South 48

Figure A17: Karsten Creek Fairway 13, Middle Section, Stillwater, OK, June 13, 2000, East-West 49

Figure A18: Karsten Creek Fairway 13, Middle Section, Stillwater, OK, June 13, 2000, North-South 49

Figure A19: Karsten Creek Fairway 13, Southeast Section, Stillwater, OK, June 13, 2000, East-West 50

Figure A20: Karsten Creek Fairway 13, Southeast Section, Stillwater, OK, June 13, 2000, North-South 50

Figure A21: Karsten Creek Fairway 12, North Section, Stillwater, OK, June 20, 2000, East-West 51

Figure A22: Karsten Creek Fairway 12, North Section, Stillwater, OK, June 20, 2000, North-South 51

Figure A23: Karsten Creek Fairway 12, South Section, Stillwater, OK, June 20, 2000, East-West 52

Figure A24: Karsten Creek Fairway 12, South Section, Stillwater, OK, June 20, 2000, North-South 52

Figure B1: Perkins, OK, January 21, 2000 53

Figure B2: Perkins, OK, February 8, 2000 54

Figure B3: Perkins, OK, February 29, 2000 55

Figure B4: Terrace Plot, Lucien, OK, February 29, 2000 56

Figure B5: Flat Plot, Lucien, OK, February 29, 2000 57

Figure B6: Lake Carl Blackwell Plot, Stillwater, OK, February 10, 2000 58

Figure B7: Lake Carl Blackwell Plot, Stillwater, OK, February 29, 2000 59

Figure B8: Karsten Creek Fairway 13, Northwest Section, Stillwater, OK, June 13, 2000 60

Figure B9: Karsten Creek Fairway 13, Middle Section, Stillwater, OK, June 13, 2000 61

Figure B10: Karsten Creek Fairway 13, Southeast Section, Stillwater, OK, June 13, 2000 62

Figure B11: Karsten Creek Fairway 12, North Section, Stillwater, OK, June 13, 2000 63

Figure B12: Karsten Creek Fairway 12, South Section, Stillwater, OK, June 13, 2000 64

Chapter I

Introduction

This thesis contains one paper for presentation to the American Society of Agricultural Engineers (ASAE), and one paper for publication in a turfgrass journal. The first paper, Chapter II, "Optimum Field-of-View for Mapping Optically Sensed Data in Winter Wheat," was presented at the ASAE International Conference in Milwaukee, Wisconsin, in July 2000. The second paper, Chapter III, "Use of High Resolution Optical Sensor Data for Mapping of Golf Course Fairways," will be submitted to one or more turfgrass publications. The first paper is complete as written. The second will require some integration of the introduction of the first paper for completion. Both papers are original research done by the author under advisement by Dr. M. L. Stone, Dr. J. B. Solie, Dr. W. R. Raun and Dr. G. E. Bell.

Chapter II

Field-of-View for Mapping Optically Sensed Data in Winter Wheat

Abstract

Research was conducted to determine a field-of-view for optical sensors used to indirectly measure biomass and total N uptake in winter wheat. Measurements were made with Patchen PhD600 optical sensors with integral lighting. These sensors measured red and near-infrared reflectance and were calibrated to produce normalized difference vegetative index (NDVI). The sensors were mounted across the front of a tractor on a toolbar at 0.30 m spacing at a height of 1.22 m. Field variability was analyzed by performing geostatistical analysis of the sensor data. Multiple semivariance ranges and integral scales were discovered, with highest NDVI relatedness of less than 1.5 m. A recommended field element size was determined to be 1.0 m to 1.5 m.

Introduction

Considerable research has been conducted to improve technology for precision agriculture, a method for applying only the appropriate amounts of chemicals (pesticides, fertilizers, etc.) to specific areas of agronomic crops. The goal of this site-specific application is to assure the grower that each small area of a field receives the precise amount of product required. In order to apply the correct amount of chemicals or nutrients to an individual area, the requirements of that area must be determined. A proposed method for determination of the needs of that area has been to use an optical sensor to measure the amount of electromagnetic radiation reflected by the plant and surrounding soil (Stone et al., 1996; Sawyer, 1994). The detected radiation includes, but is not limited to, the visible spectrum. To effectively treat a specific area, a resolution for crop sensing must be established. This research will aid in selecting an optimum size area that can be sensed and treated for maximum product effectiveness and efficiency.

Objective

The objective of this study was to determine a "field element size" or an area of winter wheat that is, statistically speaking, significantly different from its surrounding areas when indirectly measured by an optical sensor.

Literature Review and Semivariance Analysis Background

Normalized Difference Vegetative Index (NDVI) and other spectral indices have been shown to be highly correlated to plant tissue biomass and chlorophyll content (Stone et al., 1996; Wanjura and Hatfield, 1987; Kleman and Fagerlund, 1987; Dusek et al., 1985). NDVI is the ratio of the difference of red and near-infrared irradiance to the sum of the red and near-infrared irradiance reflected from the surface of interest.

Spatial variability has been studied for several nutrients and soil properties, as well as for optical sensor data. Solie et al. (1996) used an optical sensor to calculate a spectral index (PNSI) to determine a field element size for nitrogen application for winter wheat, using single pass transects in one direction. They found that sub-meter variability existed for all plots in that study.

Analysis of total N in a pasture by West et al. (1989) and of nitrate-N in cotton by Tabor et al. (1985) showed that a drift, or trend (also called a gradient), might occur across the pasture or field.

Guertal (1993) used semivariance to examine spatial variability of nitrate-N and total-N in a field, the 5-m interval between samples was too coarse to determine small-scale spatial variations. This work demonstrated evidence of pseudocycling, the effect of some apparent periodic repeating of a measured variable, as might be induced by tillage, or an applicator that applies a greater amount of chemical to one side than the other.

Cahn et al. (1994) presented a strong argument for high-resolution sampling of soil nutrients, especially nitrate-N. In addition to nitrate-N, they measured organic carbon, soil water content, phosphate-P, and potassium. They discovered that nitrate-N had a spatial correlation of less than 5 m. Using a sampling resolution of 1 m, rather than the original 50-m resolution, the measured variance of nitrate-N between two sample points was decreased by a significant amount. This indicated a higher degree of relatedness between points that were separated by a shorter distance.

Spatial variability within a field can be seen on a surface map of NDVI. This variability can be quantified by determining the distance between unrelated NDVI points. As distance between two sample points increases, there should be some distance at which they are not statistically related. The character of the relationship with respect to separation distance can be determined using semivariance. Semivariance of all samples in a data set is the statistical moment of inertia, $\gamma(h)$.

$$\gamma(h) = \frac{1}{2n} \sum_{i=1}^{i=n} (z(i) - z(i+h))^2 \qquad \text{(Eq. 1)}$$

$$h = \Delta/\delta \qquad \text{(Eq. 2)}$$

n = number of pairs of points at a separation distance Δ

z(i) = value at the i^{th} point

Δ = separation distance between two points of interest (an even multiple of δ)

δ = incremental distance between adjacent sample points

Alternatively stated, it is one-half the average squared difference between the values of all points separated by a given distance (Isaaks and Srivastava, 1989).

Semivariance for a set of data can be computed for separation distances from $\delta \leq h \leq L$, where L is the maximum distance where greater than 100 sample pairs exist. At least 100 sample pairs are needed for an accurate estimation of the true semivariance (Russo and Jury, 1987).

A plot of semivariance versus separation distance is called a semivariogram. The distinguishing characteristics of a semivariogram that are used to establish a maximum resolution for determining crop variability are the nugget, sill, range, and integral scale (Figure 1).

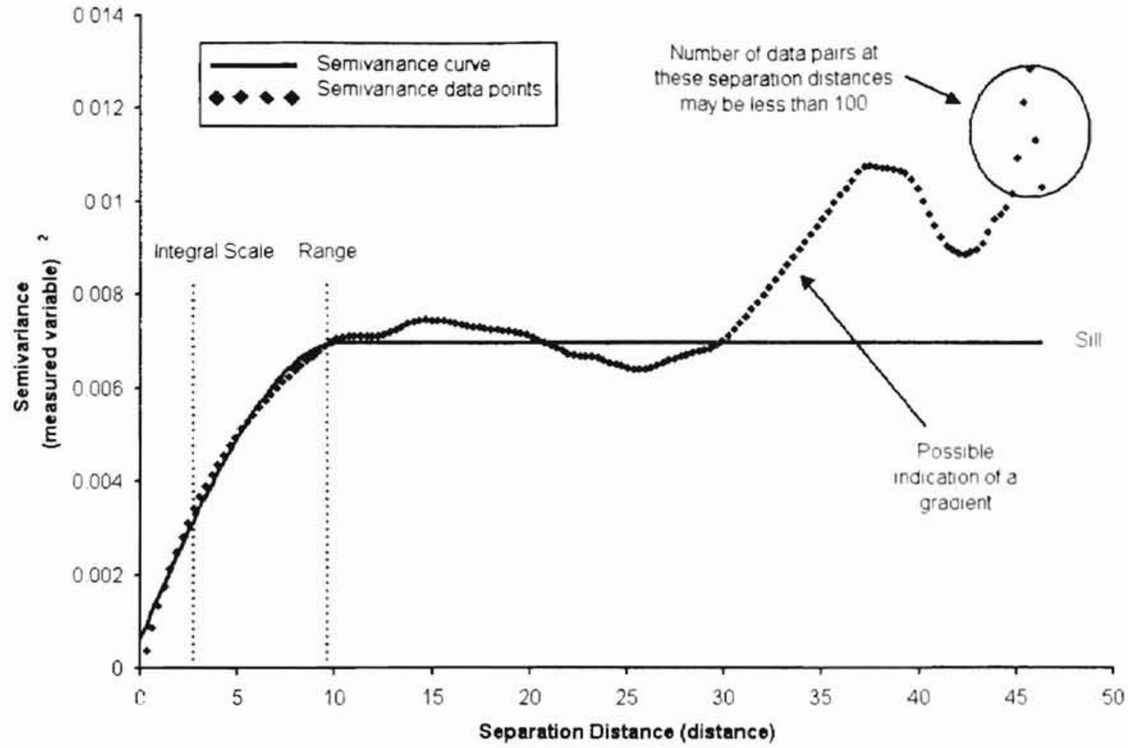

Figure 1: Example semivariogram with Sill, Range, and Integral Scale labeled and the indication of a possible gradient noted

The nugget is the point where the semivariogram curve intercepts the ordinate (semivariance)-axis. It is indicative of the variability within the sample. The amount of plant coverage, affected by plant density and size, soil color, and noise generated within the sensor are causes of most of the variability within a sample. A sill is a maximum, or plateau, of the semivariance curve, and its value is an estimation of the semivariance of the entire population. The range is the separation distance at which two sample points become unrelated to each other (Isaaks and Srivastava, 1989). A change in slope of the semivariance data beyond the range may indicate an overall gradient, or drift in the field (Royle et al., 1980).

The semivariance curve can be fitted to the portion of the semivariance data from the nugget to the beginning of the sill. That curve can take on several forms: Spherical, Linear, Exponential, and Gaussian forms are the most common (Knighton et al., 1987; Isaaks and Srivastava, 1989). The equations for these forms are as follows:

Spherical: $\gamma(h) = C[(3h/2a) - (h^3/2a^3)] + C_o \quad h<a$ (Eq. 3)
$\gamma(h) = C \quad h>a$

Linear: $\gamma(h) = Ah + B$ (Eq. 4)

Exponential: $\gamma(h) = C_o + C[1-\exp(-h/a)]$ (Eq. 5)

Gaussian: $\gamma(h) = C_o + C[1-\exp(-h^2/a^2)]$ (Eq. 6)

h = separation distance between any two sample points

Han et al. (1994) used the mean correlation distance (MCD), or the integral scale, of the semivariogram curve as the maximum field element size used in order to effectively treat spatial variability in the crop. Integral scale is calculated in the following manner:

$$\text{Integral Scale} = \int_0^{h_{max}} \rho(h)dh \qquad \text{(Eq. 7)}$$

h_{max} is the maximum separation distance between sample points

$$\rho(h) = \frac{(c_0 + c) - \gamma(h)}{c_0 + c} \qquad \text{(Eq. 8)}$$

c_0 is the nugget

$c_0 + c$ is the sill

$\gamma(h)$ is the experimental model of semivariance as defined above.

An estimate of the integral scale can be performed by finding the h-coordinate of the centroid of the area beneath the line defined by the sill ($\gamma = c_0 + c$) and the curve $\gamma(h)$.

The field element size proposed by Han et al. was 20 m. The results from their research showed variation at much smaller separation distances, but they had also proposed a minimum field element size to compensate for applicator response times.

yield monitor width, and GPS positioning accuracy for equipment available at that time. The precision of GPS systems has increased since 1994, with resolution improving from several meters to less than one meter. In addition, significant work has been done since then on applicator response time, allowing more rapid changes in application rates. Although the minimum field element size, or area to be treated as one unit, proposed by Han et al. was relatively large, the minimum field element size may be as small as a single plant's rooting diameter. Most wheat plants have a roots that spread to a radius of about 0.3 m (Lersten, 1987) from the stalk of the plant; thus a minimum field element size of 0.6 by 0.6 m is plausible.

Variability of many soil properties including, but not limited to, soil pH, total N, and organic carbon, as well as plant properties were documented at intervals as small as 1.5 m by Taylor (1996). Evidence to support the occurrence of variability at smaller resolutions was also demonstrated. Raun et al., 1998, harvested bermudagrass forage and soil sampled at 0.30 by 0.30 m resolution and found significant variability in samples at less than 1 m resolutions.

An agronomically appropriate field element size would be the minimum field element size that should be treated differently than its neighboring elements. The size would be found without regard to application equipment, GPS resolution, or other constraints. The statistical variability may or may not be of the same order of magnitude as the agronomic variability. A need exists to analyze the statistical spatial variability at very high resolution to determine the crop variability. From this statistical variability, a basis for an appropriate field element size can be determined. One method for doing this would be to use high-resolution optical sensor data to determine plant variability. Sensors to detect plant irradiance can be used to determine NDVI, and from this high-resolution data the spatial variability of the crop can be determined, based on semivariance. From the semivariogram, a proposed field element size can be established.

Discussion of Sensor Output

Spectral indices based on reflected electromagnetic radiation from a plant have been shown to correspond to plant N-uptake and biomass. NDVI, a commonly used spectral index for characterization of plant properties, is the ratio of the difference of near-infrared and red reflectance to the sum of their reflectance (Stone et al., 1996).

$$\text{NDVI} = \frac{\%\text{Red} - \%\text{NIR}}{\%\text{Red} + \%\text{NIR}} \qquad \text{(Eq. 9)}$$

$$\%\text{Red} = \frac{\text{Red Reflected}}{\text{Red Incident}} \qquad \text{(Eq. 10)}$$

$$\%\text{NIR} = \frac{\text{NIR Reflected}}{\text{NIR Incident}} \qquad \text{(Eq. 11)}$$

To determine NDVI, Patchen PhD600 (Patchen, Inc., Ukiah, CA) optical sensors were used to detect energy in the Red (approximately 670 nm) and Near-Infrared (NIR) (approximately 780 nm) bands. Each sensor emitted modulated energy in each of those bands from integrated light emitting diodes (LEDs) at 80 kHz with the phase of 90^{0} between each band. The sensor used a high pass filter to filter out background energy, summed the Red and NIR energy reflected from the target, and determined the phase shift of the sum of reflected energy. NDVI calculated using reflectance was found to be linearly proportional to voltage output from the sensor. The calibration was performed using another optical sensor that had been previously calibrated to a barium sulfate coated white plate. Barium sulfate has reflective qualities that reflect nearly one hundred percent of energy from 200 nm to 2500 nm.

Preliminary research indicated that sensor output was affected by temperature. Sensors were not electronically uniform, and required individual calibration. The following calibration equation was used to calculate NDVI as a function of sensor voltage and ambient air temperature:

$$\text{NDVI} = a\frac{V}{bT^3 + cT^2 + dT + e} + f \qquad \text{(Eq. 12)}$$

a, b, c, d, e. and f are calibrated constants that vary from sensor to sensor

V = Voltage from 0 to +5 Volts

T = Ambient air temperature in degrees Fahrenheit

Refer to Appendix C for values of 'a, b, c, d, e. and f.'

Four sensors were positioned 0.30 m apart across the front of a John Deere 4100 (Deere and Co., Moline, IL) utility tractor (Figure 2). Each sensor had a field of view of approximately 0.30 m by 0.0095 m (Figure 3). The tractor was equipped with a Trimble Ag 132 (Trimble Navigation, Sunnyvale, CA) Global Positioning System (GPS) with differential correction by Omnistar (OmniSTAR, Inc., Houston, TX). The antenna was positioned 2.6 m behind the row of sensors (Figure 2).

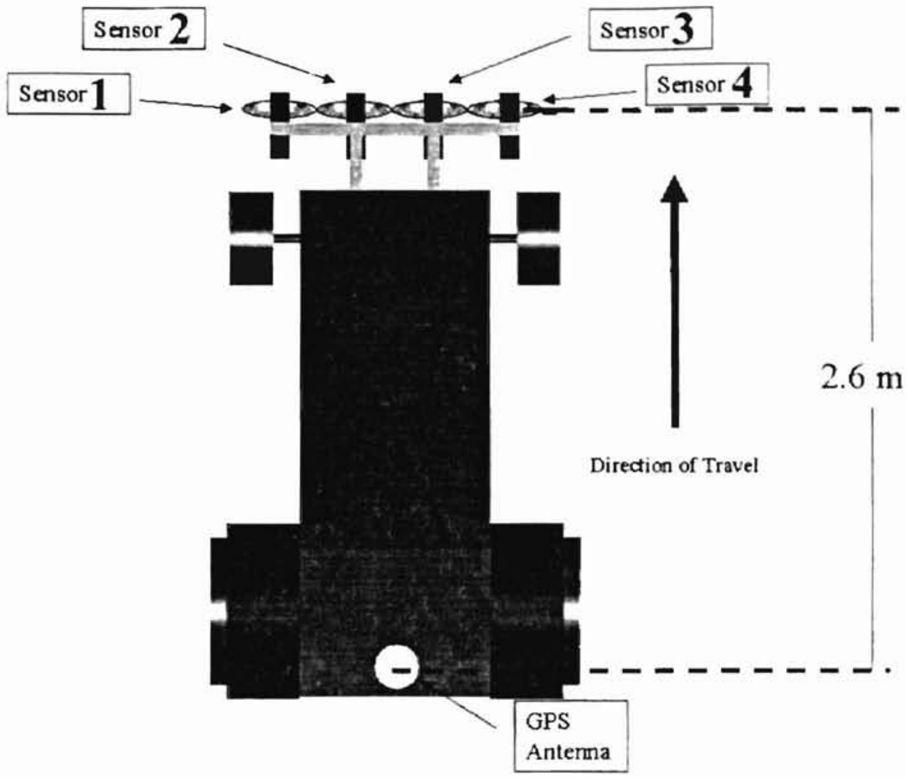

Figure 2: Sensor and antenna location on vehicle

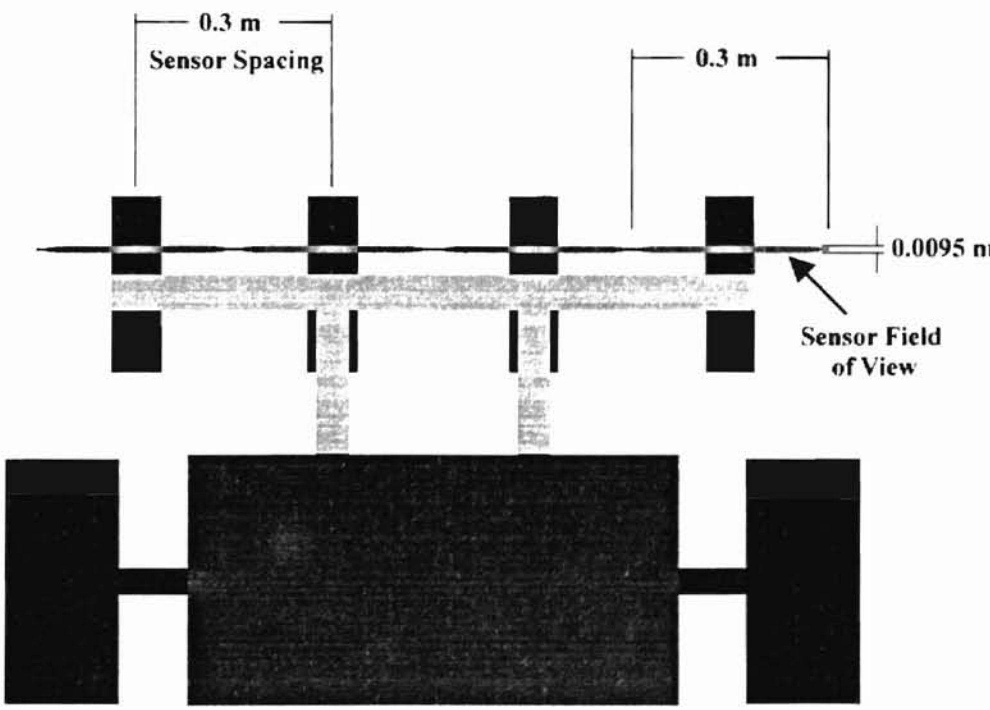

Figure 3: Dimensions of sensor spacing and field of view

A Dynapar (Danaher Controls, Elizabethtown, NC) optical shaft encoder attached to a front tire of the tractor transmitted a TTL signal at 200 pulses per revolution of the encoder. The encoder was belt driven by a front wheel and produced 640 pulses per meter of vehicle travel. The frequency was reduced through a digital divider that transmitted 1 outgoing pulse per 32 incoming pulses. Based on the wheel size, shaft encoder output, and reductions, this transmitted a pulse to an IOtech DaqBook/100 (IOtech, Inc., Cleveland, OH) data acquisition device for every 0.05 m of travel. As the DaqBook read each pulse, the digital value of each sensor voltage was converted through an IOtech DBK11 12-bit A/D board and transmitted through a parallel port to a laptop computer. A custom software program read the voltage each time an encoder pulse was received. The program simultaneously read geographic position messages from a serial port as they were updated. These voltage values and GPS data were saved and post-processed using the procedure described below.

Methods of Data Processing and Analysis

The GPS recalculated position every 0.2 seconds (5 Hz). However, because the program recorded a sensor reading every 0.05 m, the number of readings per second typically exceeded 5 per second, depending on vehicle speed. This caused every set of data points recorded between GPS updates to be assigned exactly the same latitude and longitude, even though they were actually recorded at different locations. Because of this, interpolation was necessary to assign latitude and longitude for each set of sensor readings. Linear, cubic spline, and LaGrange polynomial methods for interpolation were investigated for this purpose.

Linear interpolation fits a straight line between two known points (Hornbeck, 1975). In cases where large distances do not separate points, and changes in direction of travel between points are small, a linear interpolation method is accurate.

The cubic spline method fits a third order polynomial between each pair of known points (Hornbeck, 1975). It is useful for interpolation between unevenly spaced points. The number of data points corresponding to a single position is dependent on velocity. The velocity may change frequently during the course of taking data, resulting in widely varying numbers of data points between GPS updates.

The LaGrange polynomial method fits an *n*th order polynomial to n-1 unequally spaced points (Hornbeck, 1975). This works well for a small number of points; however, the data for a large field may contain several thousand or more points. For such data, it would be necessary to use a smaller set of points and shift the set for every new known point.

The three methods were evaluated in several fields, and results plotted in SSToolbox (Site Specific Technologies, Stillwater, OK), a GIS program. The deviation between methods was checked at a number of locations. The maximum distance deviation between any of the three methods averaged less than three inches. Because the differences in accuracy between the three methods were small, for computational simplicity and speed, linear interpolation was used.

To obtain high-resolution sensor data, a position coordinate must be assigned to each sensor reading. Because there was only one GPS antenna on the machine with multiple sensors, position coordinates had to be calculated for each sensor with respect to the GPS antenna. For this study, a post-processing program was written to assign a geographic position to each sensor reading.

To calculate a coordinate for each sensor reading, vehicle direction at the time of each measurement had to be established. Direction of travel was established by calculating a vector from previously measured position coordinates. The location of each sensor at each reading was then calculated based on machine geometry.

A test was performed to determine the positioning error from the program. Locations on a uniform grass surface were marked with flags, and the distance between each flag was measured. A path was driven over the grass and at every flagged location, a mirror was placed on the ground to create a "spike," or a sudden change in NDVI from the grass surface. The data was processed through the procedure described above and a visual map was created in SSToolbox. The distance between spikes was measured using the measure function of SSToolbox. The physical distance measured on the ground was compared to the distance measured from the map generated in SSToolbox. The error was computed in percent difference. On average, the percent difference was 5.9 percent, at an average physical separation distance of 12.0 m. Error for the Trimble Ag132 with differential correction has been reported to be ± 1 m with a 95 % confidence interval (personal contact, Dr. Paul Ayers. Colorado State University, Fort Collins, CO).

	Measured Distance (m)	Distance from GIS (m)	Difference (m)	Absolute Value of Difference (m)	Percent Absolute Difference
	12.88	12.04	0.840	0.840	6.522
	8.43	8.61	-0.180	0.180	2.135
	9.20	8.31	0.890	0.890	9.674
	16.05	16.05	0.000	0.000	0.000
	6.83	7.85	-1.020	1.020	14.934
	9.92	10.20	-0.280	0.280	2.823
	7.07	7.17	-0.100	0.100	1.414
	8.77	8.07	0.700	0.700	7.982
	6.66	6.66	0.000	0.000	0.000
	4.18	4.85	-0.670	0.670	16.029
	13.75	12.89	0.860	0.860	6.255
	11.11	10.83	0.280	0.280	2.520
	4.98	4.83	0.150	0.150	3.012
	3.97	3.21	0.760	0.760	19.144
	8.53	7.66	0.870	0.870	10.199
	10.57	9.86	0.710	0.710	6.717
	5.59	5.78	-0.190	0.190	3.399
	33.01	32.52	0.488	0.488	1.479
	23.20	22.89	0.313	0.313	1.349
	36.40	35.50	0.898	0.898	2.468
Average	12.05492	11.78895192	0.26596808	0.50996808	5.902695859

Table 1: Error Calculation for Positioning

The sensor field of view (0.30 by 0.0095 m) was rectangular, and a grid of square pixels was created by averaging all the data within each 0.3 by 0.3 m element. It was critical to determine the best method for resampling the data within that 0.30 by 0.30 m area, so as not to induce a relatedness between pixels that did not truly exist. Nearest Neighbor, Inverse Distance to a Power, and Kriging (Golden Software, Inc., 1999) were investigated.

The objective was to calculate optimal NDVI values for each 0.3 by 0.3 m pixel that were not affected by surrounding NDVI values, yet would utilize all sensor readings within that area. To accomplish this, Inverse Distance to a Power was used. This method of resampling assigns a weight to each data point that is inversely proportional to the data point's distance from the center of the pixel of interest. That pixel value then becomes the weighted average of all values. With a power of one, and a search radius of 0.15 m, this method effectively approximates a weight of 1 for each data point within the 0.3 by 0.3 m square pixel. It assigns a weight of 0 to each data point outside 0.15 m of the pixel center.

The weighted averaging was performed using Surfer, a software package from Golden Software, Inc., which can be used to create grids and maps from point data. The values from each pixel center were then extracted from the grid and assigned a latitude and longitude. The resulting values were then used to perform semivariance analysis.

High-resolution optical sensor measurements were made on winter wheat during the months of January and February 2000. All plots scanned were 45.1 m by 45.7 m. Visual approximations of ground coverage and growth stage ranged from about 50% at Feekes stage 3 to about 80% at Feekes stage 5. Winter wheat at Feekes stage 3 often is beginning to lie prostrate on the ground, with several leaves formed. At Feekes stage 5, the sheaths of the leaves begin to form a strong "pseudo-stem" in an erect position (Feekes,1941).

The wheat cultivar Custer, planted with 0.19 m row spacing, at the Oklahoma State University Research Station at Perkins, Oklahoma was scanned on January 21, February 8, and February 29, 2000. This plot had previously been a research area and visually appeared highly variable. The plot was situated on level ground between two terraces that ran from east to west. Growth stages ranged from Feekes stage 4 to Feekes stage 5 during the scanning period. In the southeast corner of the plot, crabgrass suppressed the wheat.

The wheat cultivar '2137', planted with 0.15 m row spacing, in an OSU Plant and Soil Science field near Lake Carl Blackwell, Stillwater, Oklahoma was scanned on February 10 and February 29, 2000. The plot had a slight slope downhill to the east. The wheat was at the later part of Feekes stage 3 to the early part of Feekes stage 4. A strip that had been planted twice was evident through the center of the plot, running in the east-west direction.

Two plots in a field of wheat cultivar Coker near Lucien, Oklahoma were scanned on February 29, 2000. One of these plots was on level ground (Lucien Flat Plot), and the other had a terrace across the plot (Lucien Terrace Plot). The terrace ran through the plot from the North center toward the Southeast corner. The level plot had a strip that had been planted twice through the center of the plot, but was not as obvious at the strip at the Lake Carl Blackwell site.

The areas were scanned with the John Deere 4100 tractor equipped with the Patchen PhD 600 optical sensors. Sensing passes were made with a swath width of 1.22 m. Data were processed and analyzed using the methods described above. A detailed experiment and analysis procedure is listed in Appendix C.

A Microsoft Excel Visual Basic for Applications (Microsoft Corp., Redmond, WA) spreadsheet macro was written to calculate semivariance for all separation distances. Semivariance was calculated and semivariograms plotted. The expected sill was visually determined. The data were truncated to the expected sill, and semivariogram transition models fitted to the data using Table Curve, a software program from SPSS, Inc. (AISN Software, Inc., 1996). A spherical (Eq. 2) equation typically fit the data very well. In some cases, an exponential equation (Eq. 4) fit more accurately. The integral scale was calculated by numerically integrating the equation for data in the transition region.

Results and Analysis

Integral scales for the plots ranged from 0.55 m to 3.69 m (Table 1). This is consistent with the work of Solie et al., 1996 and Solie et al., 1999, who found variability at the sub-meter level.

Plot	Date	Direction	1st Sill	2nd Sill	3rd Sill	4th Sill	1st Range (m)	2nd Range (m)	3rd Range (m)	4th Range (m)	Nugget	Integral Scale (m)
Perkins	January 21	e-w	0.00291	0.00314	0.00338	0.00363	4.60	6.10	9.75	11.89	0	1.66
Perkins	January 21	n-s	0.00235	0.00354	--	--	2.81	11.28	--	--	0	1.15
Perkins	February 8	e-w	0.00298	0.00364	--	--	5.64	14.02	--	--	0	2.24
Perkins	February 8	n-s	0.00200	0.00304	0.00304	--	2.12	8.53	16.76	--	0.00006	1.15
Perkins	February 29	e-w	0.00504	0.00622	0.00683	--	5.00	9.14	12.80	--	0	2.06
Perkins	February 29	n-s	0.00276	0.00463	0.00623	--	2.33	7.92	14.94	--	0.00012	1.02
Lucien Terrace	February 29	e-w	0.00966	0.01106	--	--	7.47	11.89	--	--	0.00010	2.67
Lucien Terrace	February 29	n-s	0.00697	--	--	--	10.25	--	--	--	0	3.69
Lucien Flat	February 29	e-w	0.00189	0.00340	--	--	1.74	7.32	--	--	0.00012	0.80
Lucien Flat	February 29	n-s	0.00179	0.00373	--	--	1.55	11.89	--	--	0.00012	0.55
Lake	February 10	e-w	0.00316	0.00316	--	--	13.13	25.60	--	--	0.00020	3.29
Lake	February 10	n-s	0.00380	0.00458	--	--	2.19	3.35	--	--	0	0.83
Lake	February 29	e-w	0.00293	0.00399	--	--	5.70	11.58	--	--	0.00035	1.53
Lake	February 29	n-s	0.00454	0.00719	--	--	3.36	11.58	--	--	0.00012	0.91

Table 2: Relevant Semivariance Statistics for Winter Wheat Plots

Pseudocycling was evident in several semivariograms. The most obvious case was for the Lake Carl Blackwell Plot on February 10, 2000, where tillage or other implement effects induced pseudocycling in the North-South Direction (Figure 4). The effect of the twice-planted strip can be seen on the map of NDVI (Figure 5).

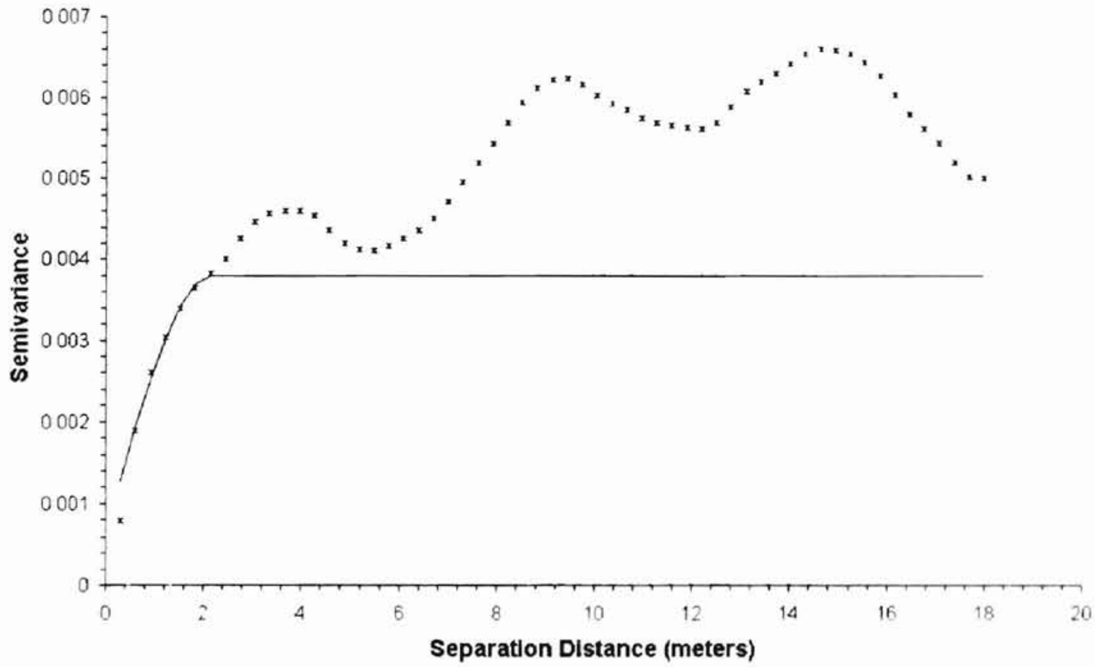

Figure 4: Lake Carl Blackwell Plot, Stillwater, OK, February 10, 2000, North-South Semivariance

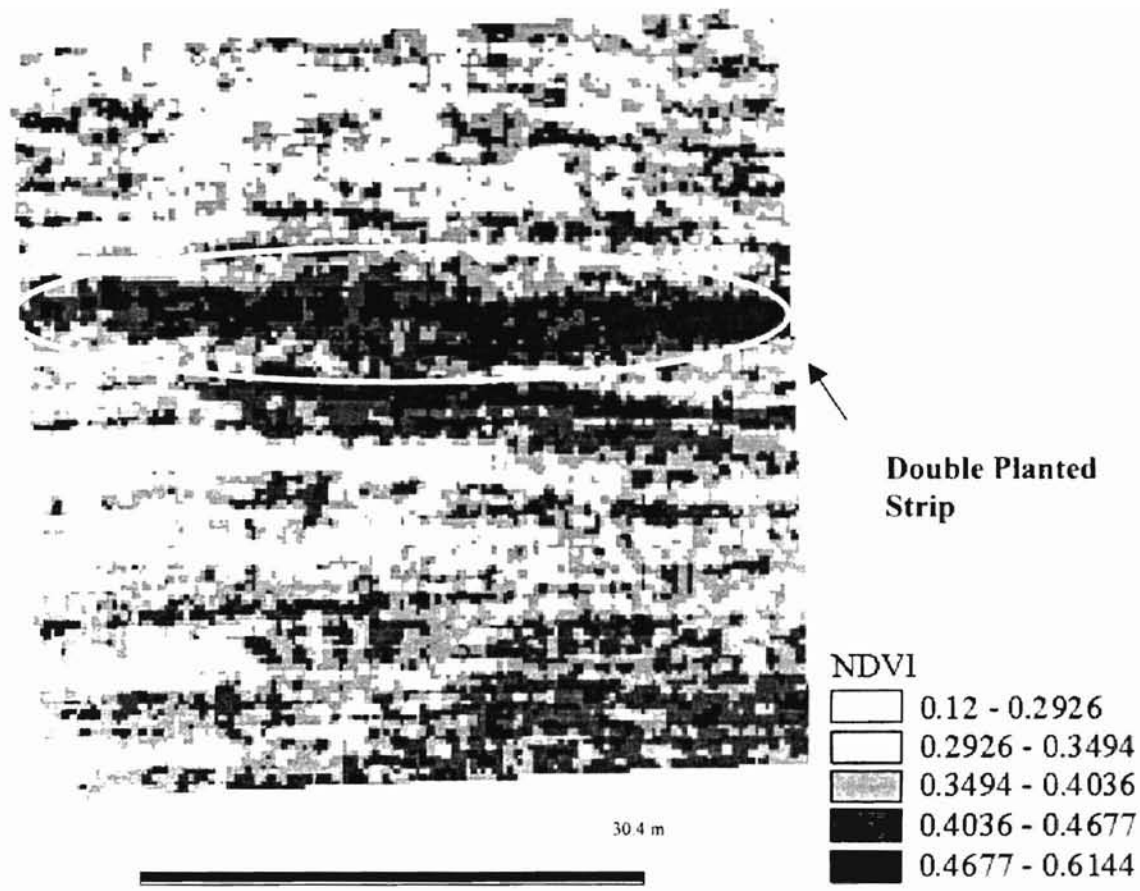

Figure 5: Map of Lake Carl Blackwell Plot, February 10, 2000

Individual plant differences (genetic variation), fertility, tillage and implement effects, and soil differences are the most likely causes of nested sills. Small-scale variation causes the sills at shorter separation distances (1 to 2 m). These are most likely caused by the genetic variation in the plants and fertility levels. The larger-scale variations at the larger separation distances (4 to 6 m) are typically caused by tillage and implement effects and even larger-scale sills by soil type variation. Nested sills were obvious in most of the semivariograms, but were most noticeable in the Flat Plot at Lucien (Figure 6) and some of the semivariograms from the Perkins plot.

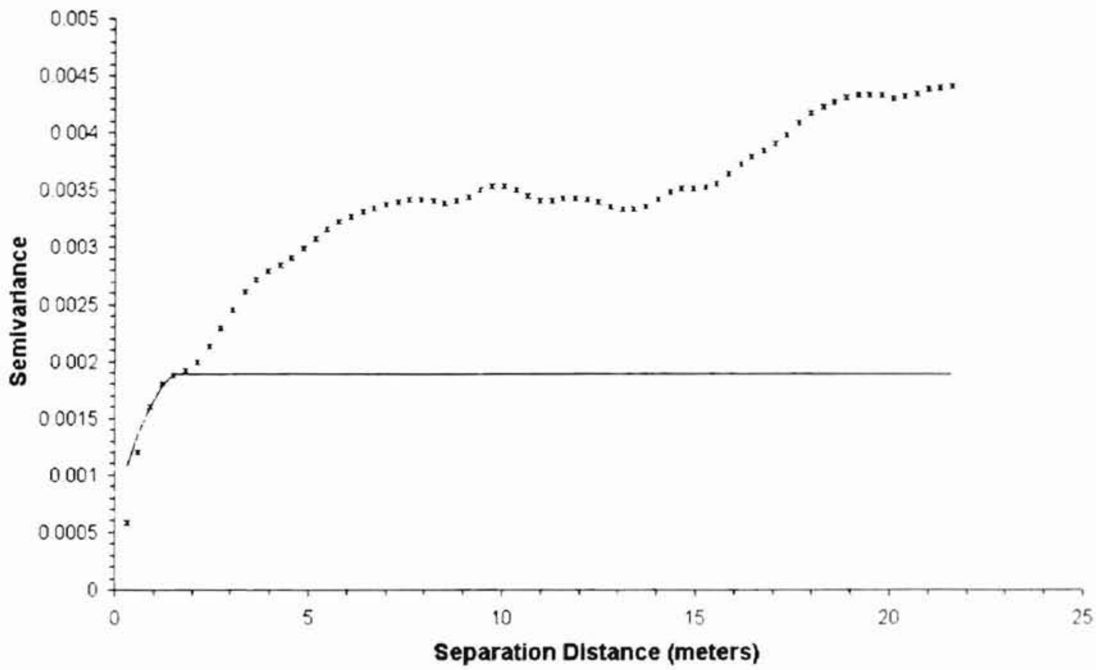

Figure 6: Lucien Field, Flat Plot, February 29, 2000, East-West Semivariance

Some of the semivariograms exhibited drift, a possible indication of a gradient throughout the plot. This gradient might be due to slope or other terrain effects, or soil type changes. The Lucien Flat Plot showed signs of drift in the East-West direction (Figure 6). One cause of this could be a drainage effect, since the ground directly to the west began to slope away from the plot.

Although the true semivariance at a separation distance of h=0 should be 0, the intercept of the curve fitted to the data (the nugget) will typically not be 0. This can be caused by factors such as sampling error, instrument error, and variability on a scale smaller than the sampling resolution (Isaaks and Srivastava, 1989). To determine the nugget, a straight line was fitted through the first three data points. The y-intercept of this line is the estimated nugget, as reported in Table 1. If the intercept was less than 0, it was reported as 0, since there cannot be a negative semivariance. The straight-line estimation of the nugget is shown in Figure 7. By using the data points, rather than the semivariance curve to estimate the true nugget, a better understanding of the small-scale variability can be obtained.

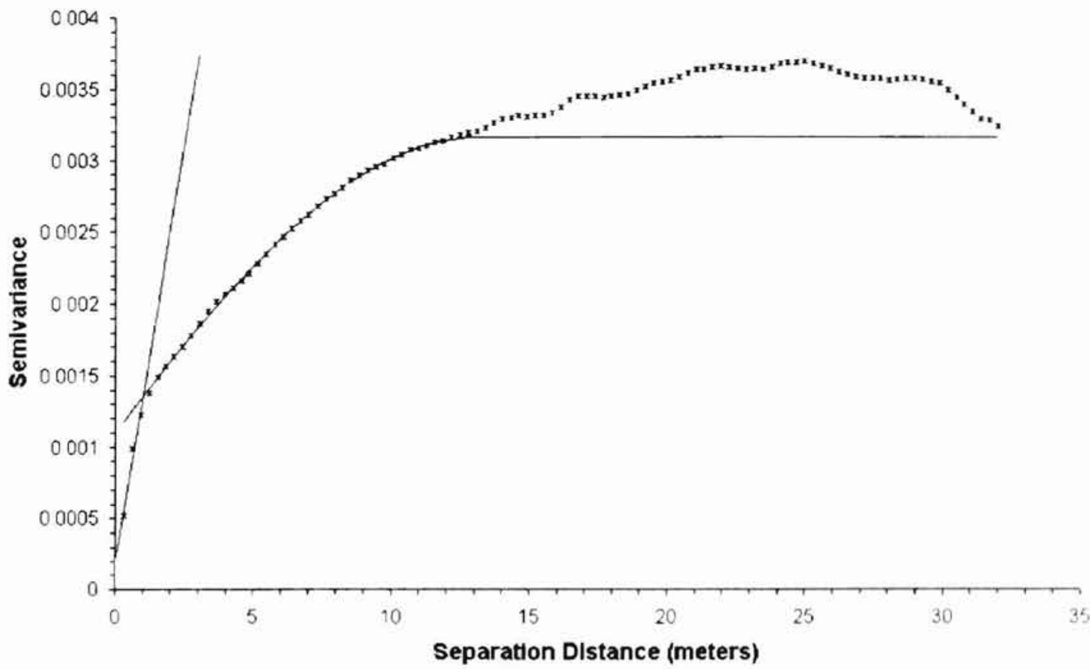

Figure 7: Lake Carl Blackwell Plot, Stillwater, OK, February 10, 2000 East-West Semivariance

Semivariance analysis of the effect of direction showed a trend with the direction of planting. In all cases, the value of integral scale was higher along the direction of planting than across the direction of planting. For the Lucien Flat Plot, the direction of planting was not parallel or perpendicular to north-south, but diagonal. The values of integral scale for that plot were similar in both north-south and east-west analyses. This suggests that a portion of the small-scale variability in winter wheat can be attributed to the variability between growing plants and bare soil. In order to sense and treat the variability of the plants, rather than trying to sense and treat the variability between plants and soil, the optimum field of view might be somewhat larger than the integral scale in the direction perpendicular to planting.

The integral scale values for the plots scanned for this study show that variability of winter wheat measured by the optical sensors exists at the sub-meter scale. The values of integral scale ranged from 0.55 m to 3.69 m. To maximize the opportunity for treating variability, a field element size of 1.0 m to 1.5 m would be appropriate for sensing small-scale variability.

Summary

The spatial variability of agronomic crops can be determined by using high resolution optical scanning of the canopy. Using NDVI, the characteristics of very small areas of crop can be detected, and using a geostatistical technique known as semivariance, the inherent spatial variability of the crop can be defined. From this, a field element size for scanning was established.

Results of high resolution optical sensing of winter wheat in Oklahoma show that variability of wheat exists below the one-meter level. Using high-resolution sensor data to construct grids of 0.3 by 0.3 m pixels in both the lateral and in-line directions allows determination of variability in all directions, and results from this study show sub-meter variability in all directions. In order to use variable rate treatment of winter wheat, sensing and treatment minimum resolutions should be made at a much higher resolution than many of the current practices. Based on the integral scales calculated, a field element size for detecting and treating the largest amount of variability could range from 1.0 m to 1.5 m.

Chapter III

Use of High-Resolution Optical Sensor Data for Mapping of Golf Course Fairways

Objective

The objective of this study was to establish the spatial variability of high-maintenance turfgrass using high-resolution optical sensor data. Also of interest was to determine the practicality of using that high-resolution optical sensor data to discover and locate areas of water stress, nutrient deficiency, and other problem areas.

Methods of Data Processing and Analysis

The methods of data collection, processing, and analysis were the same as the wheat study in Chapter II. Instead of 45.1 m by 45.7 m plots of winter wheat, two 'Meyer' zoysiagrass (zoysia japonica Willd.) fairways at Karsten Creek Golf Course, Stillwater, Oklahoma were scanned and analyzed. Scanning passes were made over the entire fairway, with adjacent pass locations determined by judgment, rather than on a measured distance. Scanning passes were driven diagonally across the fairway, rather than along the line of the fairway.

Due to restrictions in spreadsheet size, the two fairways, No. 12 and No. 13, were divided into smaller areas for semivariance analysis (Figures 8 and 9). Fairway 12 was divided into two areas, a south section and a north section. Fairway 13 was divided into three areas, a southeast, middle, and northwest section.

Figure 8: Map of Fairway 12 with semivariance sections indicated

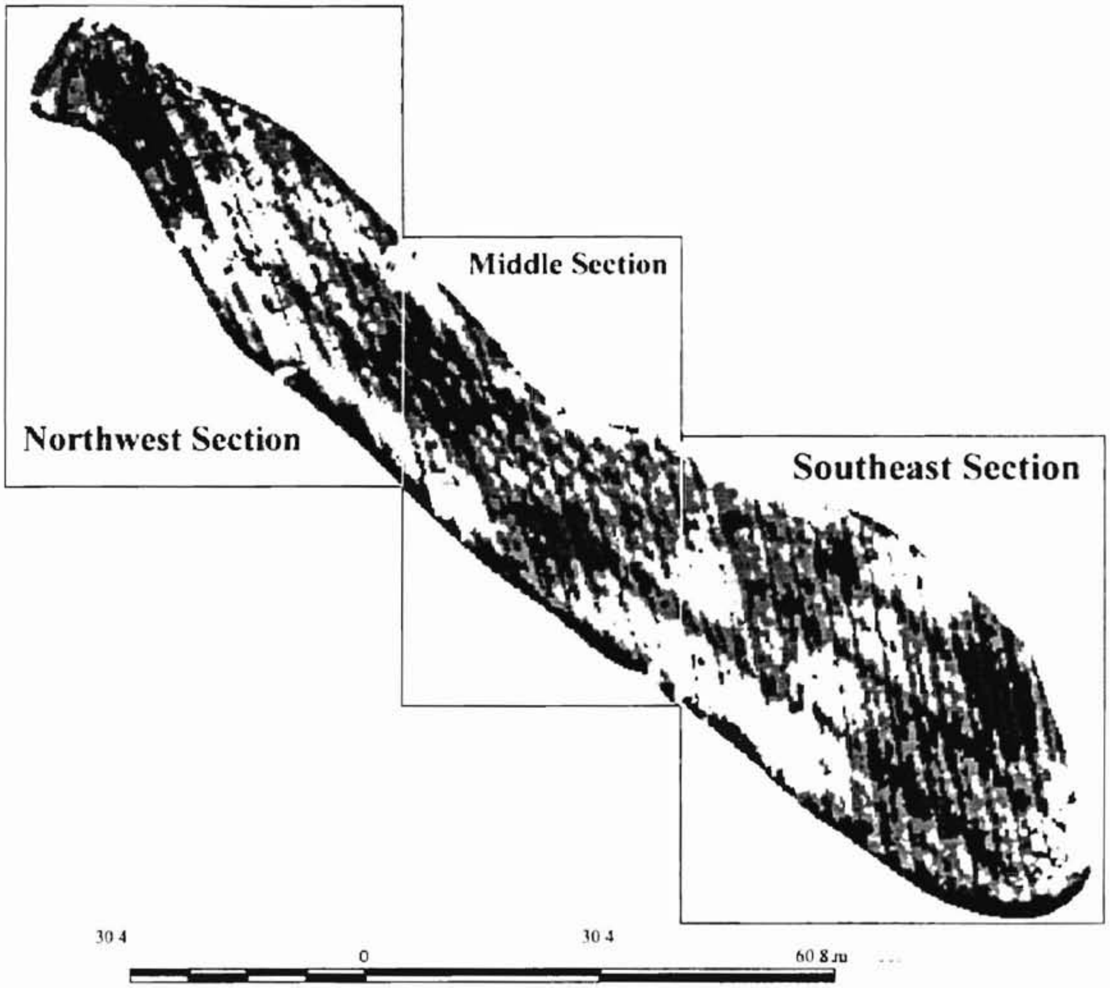

Figure 9: Map of Fairway 13 with semivariance sections indicated

Results and Analysis

Integral scales for the zoysia fairways ranged from 1.11 m to 4.98 m (Table 2). Integral scales for Fairway 13, scanned on June 13, 2000, were significantly smaller than those for Fairway 12, scanned on June 20, 2000.

Fairway Section	Date	Direction	1st Sill	2nd Sill	1st Range (m)	2nd Range (m)	Nugget	Integral Scale (m)
13 southeast	June 13, 2000	e-w	0.00058	0.000651	4.62	7.87	-0.0000317	1.93
13 southeast	June 13, 2000	n-s	0.00039	0.000583	3.35	5.79	-0.0000071	1.40
13 middle	June 13, 2000	e-w	0.00073	0.000831	4.51	8.71	-0.0000195	2.03
13 middle	June 13, 2000	n-s	0.00048	0.000656	3.19	5.38	-0.0000453	1.26
13 northwest	June 13, 2000	e-w	0.00078	0.000797	5.14	6.24	-0.0000435	2.22
13 northwest	June 13, 2000	n-s	0.00054	0.000800	2.87	6.44	-0.0000266	1.17
12 south	June 20, 2000	e-w	0.00055	0.000863	11.95	20.04	-0.0000109	4.98
12 south	June 20, 2000	n-s	0.00046	0.000632	7.80	22.99	-0.0000130	2.90
12 north	June 20, 2000	e-w	0.00082	0.000899	11.46	14.03	-0.0000201	4.33
12 north	June 20, 2000	n-s	0.00105	0.00120	10.25	17.28	-0.0000269	3.76

Table 3: Relevant Semivariance Statistics for Golf Course Fairways Scanned

Nested sills were evident in most of the semivariograms. The Middle (Figure 10) and Southeast (Figure 11) Sections of Fairway 13 showed very obvious nested sills, especially in the north-south directions.

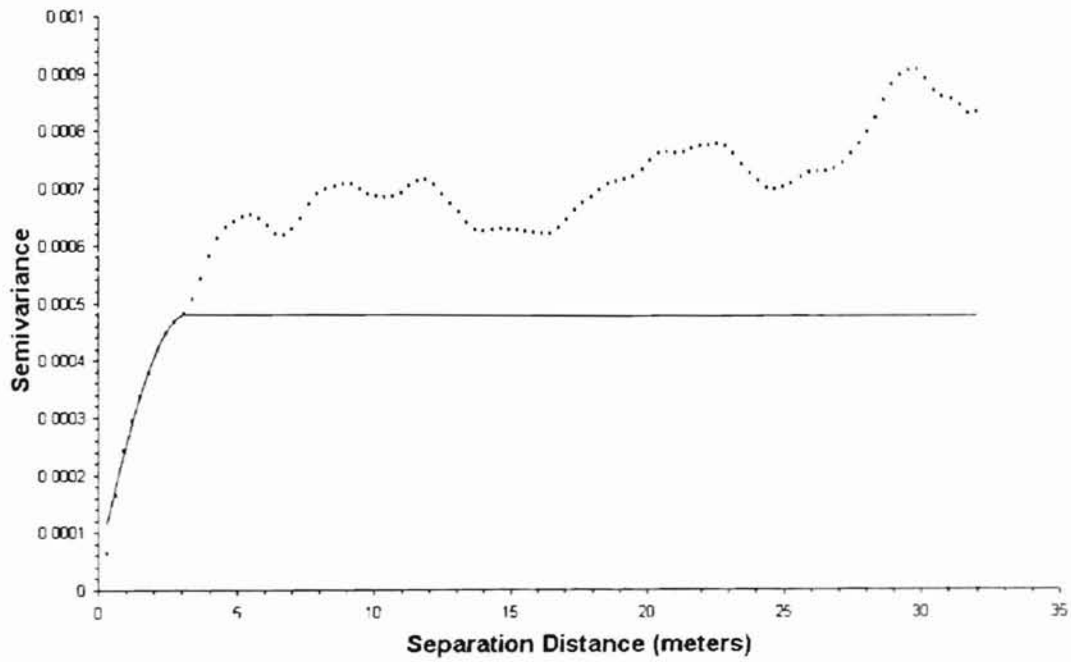

Figure 10: Fairway 13, Middle Section, Karsten Creek Golf Course, June 13, 2000, North-South Semivariance

Oklahoma State University Library

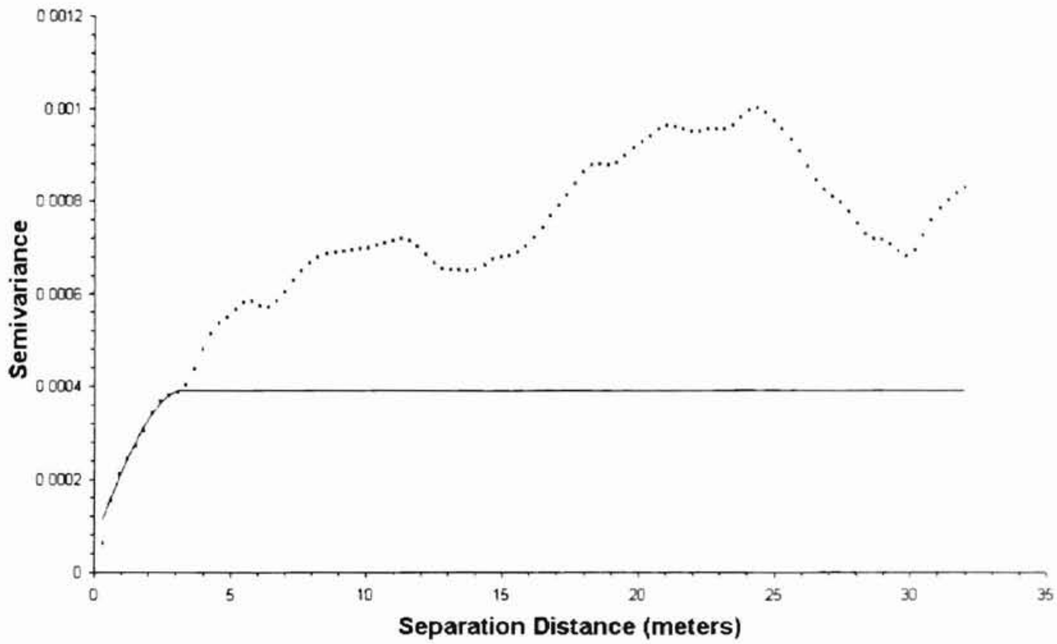

Figure 11: Fairway 13, Southeast Section, Karsten Creek Golf Course, June 13, 2000, North-South Semivariance

The small-scale variability is most likely due to nutrient differences from area to area. The likelihood of genetic variation causing small-scale variation is small, because the turf was all from the same sod farm, and all of the sod was from the same parent stock. Some of the large-scale variability is most likely due to variations in watering patterns, affected by sprinkler coverage and land slope. In The North Section of Fairway 12, nested sills were very obvious in the north-south direction (Figure 12), but less obvious in the east-west direction (Figure 13).

Oklahoma State University Library

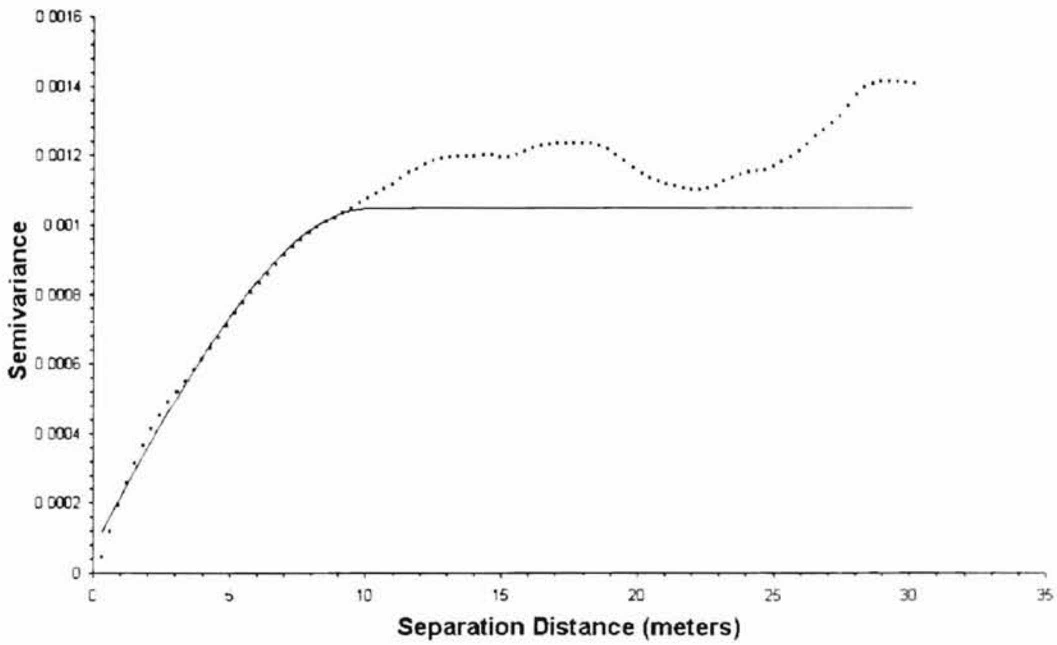

Figure 12: Fairway 12, North Section, Karsten Creek Golf Course, June 20, 2000, North-South Semivariance

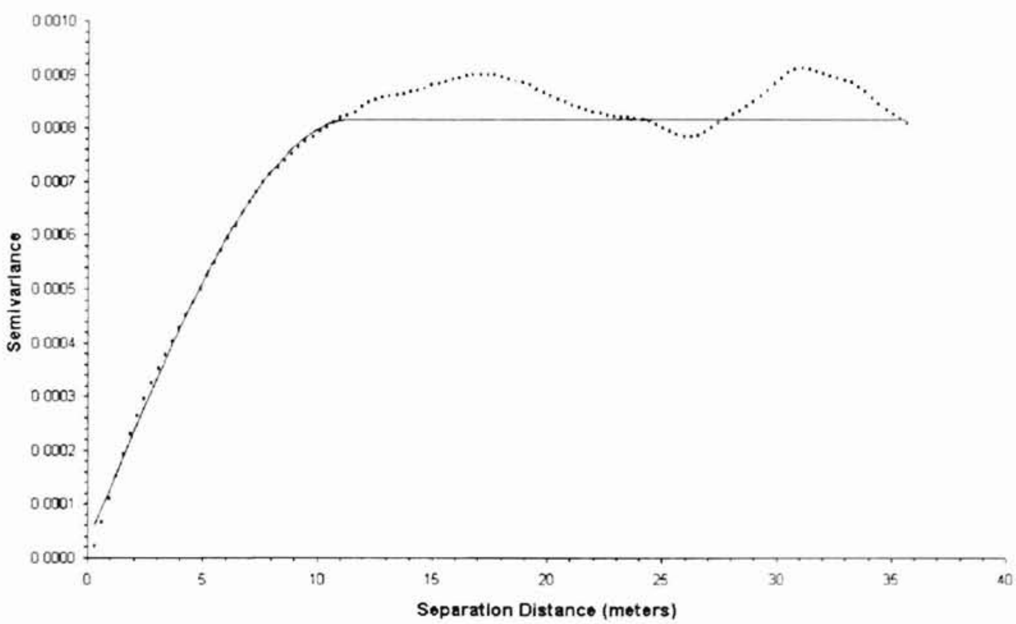

Figure 13: Fairway 12, North Section, Karsten Creek Golf Course, June 20, 2000, East-West Semivariance

Oklahoma State University Library

The nested sills in the north-south direction were possibly due to slope and maintenance practices, such as mowing and fertilizer application.

Pseudocycling in the semivariograms could be due to several factors. The driven path did not cover 100% of the fairway, so there are gaps between some sensor readings. When the resampling was done to create 0.30 by 0.30 m pixels, some of the pixels were influenced by neighboring sensor data. This "striping" effect caused some of the fluctuations in variance. It also is more noticeable on Fairway 13 (Figure 14) than on Fairway 12 (Figure 15), because the passes with the sensors were closer together for those readings. The pseudocycling of Fairway 13 occurs at about 2.5 m intervals, which is very close to twice the width of the sensor swath.

Oklahoma State University Library

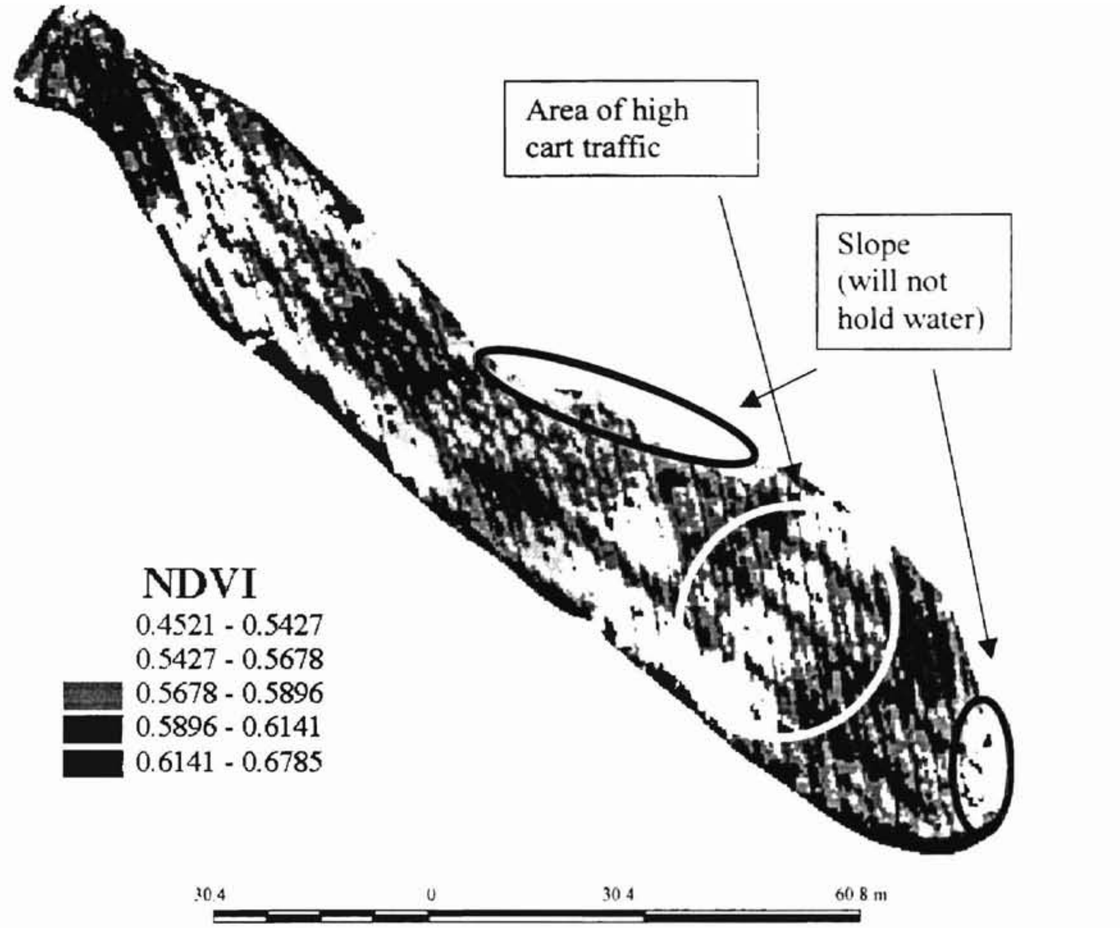

Figure 14: Fairway 13, Karsten Creek Golf Course, Stillwater, OK, Scanned on June 13, 2000

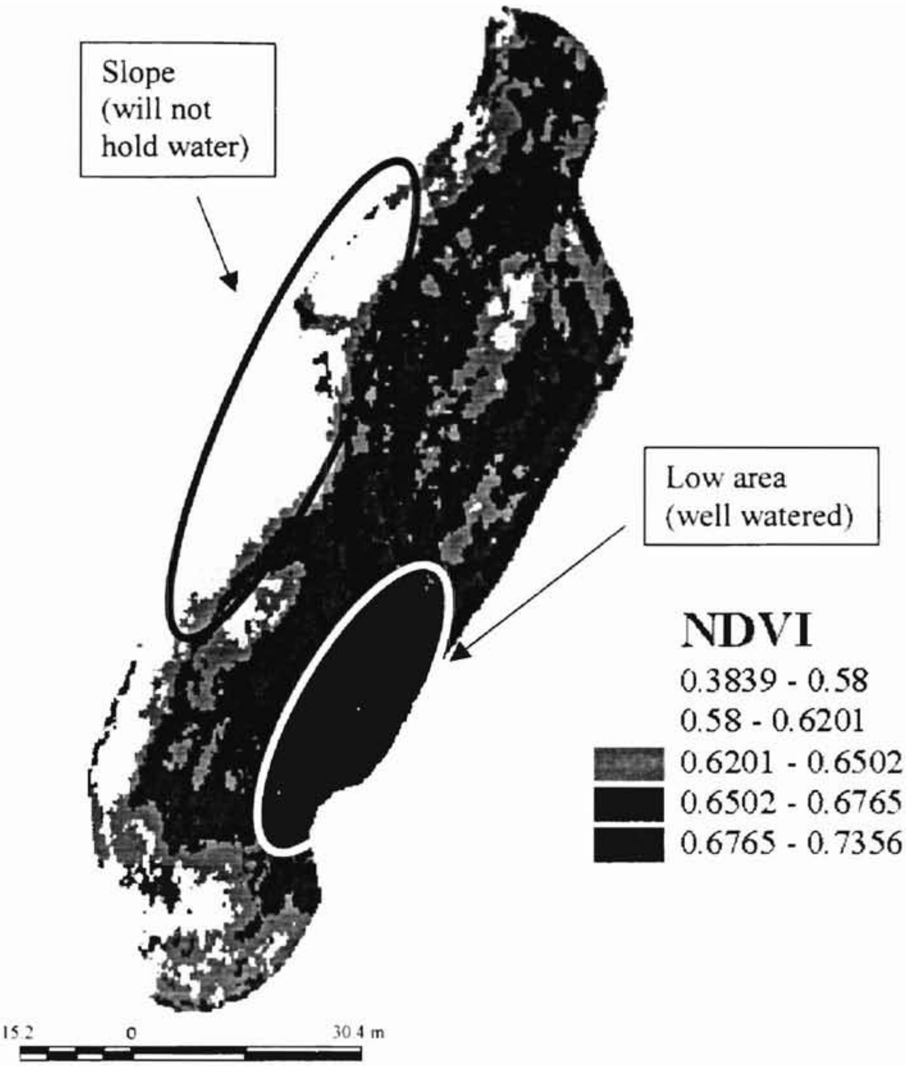

Figure 15: Fairway 12, Karsten Creek Golf Course, Stillwater, OK, Scanned on June 20, 2000

The maps of the fairways generated can be used to aid golf course maintenance. They distinctly show areas of water stress. Other effects are visible, but the causes are not immediately apparent. They could be due to cart and foot traffic, fertility variance, or other causes. Further knowledge of the causes of stress would be necessary for effective treatment.

Oklahoma State University Library

Semivariance analysis of the golf course fairways shows variability on small and large scales. An appropriate field element size as indicated by the integral scale would be 1.5 to 2.0 m. The field element size is larger than the integral scale because some of the small-scale variability found in the semivariance analyses is due to incomplete sensor coverage of the area of interest.

Summary

The spatial variability of two golf course fairways was determined using high resolution optical scanning of the canopy. The spectral characteristics of small areas of turf were detected, and using a geostatistical technique known as semivariance, the inherent spatial variability of the crop was determined. From this, a field element size for scanning was established.

Semivariance analysis of zoysia golf course fairways shows large and small-scale variability. Based on the semivariance analyses, an appropriate field element size would be around 1.0 to 1.5 m.

Mapping of golf course fairways can be a useful tool for golf course maintenance supervisors. It will not replace the supervisor, because areas of plant stress can be found by looking at a map of NDVI, but the cause may not be immediately apparent. Water stress, nutrient availability, and other sources of plant stress can cause variability and are apparent on maps of NDVI.

Oklahoma State University Library

Chapter IV

Overall Summary and Suggestions for Future Research

Overall Summary

Semivariance analysis of both winter wheat and zoysia golf course fairways showed evidence of both large and small-scale variability. The small-scale variability was used to determine a field element size for sensing and treating with variable rate technology. For winter wheat, the suggested field element size was 1.0 to 1.5 m. For zoysia fairways, presumably indicative of fairways with other types of grass, the suggested field element size was 1.5 to 2.0 m. The differences in scale are due to the fact that the grass of the fairway covered a greater percentage of the ground than the wheat plants did. The suggested field element sizes were not exactly the same as the integral scale because the wheat had small scale variability affected by the bare soil between rows of plants, and the turf had small scale variability affected by incomplete sensor coverage.

Mapping of golf course fairways can be a useful tool for finding areas that need attention. Water stress, nutrient variability and other sources of plant variability can cause these areas. A map of NDVI will locate these areas, but the golf course superintendent would still be needed for evaluation of causes and treatments.

Suggestions for Future Research

More research should be done to determine the relationship between field element size and return on investment. As field element size increases, initial costs decrease, but crop production and fertilizer and chemical costs may increase. With a smaller field element

Oklahoma State University Library

size, the cost of sensing and treating will increase, but the crop has a better chance of reaching its full yield potential.

Some improvements of the high-resolution sensor data processing program need to be made. Many of the position errors are due to changes in machine velocity. At this time, the user enters an offset distance from the GPS antenna to the middle of the toolbar that holds the sensors on the front of the tractor. If the offset distance entered is the actual distance between the antenna and the sensors, it will result in erroneous positioning of the sensors. Because of the speed of the GPS signal, the position of the tractor can be off by 0.2 m or more at a speed of 1.34 m/s. Atmospheric delays, satellite distance, and processing time increase this error. With the present method of using an offset distance that is greater than the actual distance, the position error can fluctuate more than 0.12 m for a change of only 0.45 m/s. Currently the program that records sensor voltages and geographic position does not record velocity or time. If time were recorded with position, velocity could be determined, and from that the amount of offset could be calculated for each reading, resulting in a more accurate position. An equation that relates offset distance to velocity could be used if the average signal delay was known. Offset distance would increase as velocity increases.

To get accurate data for turfgrass, the test needs to be repeated, with closer to 100% sensor coverage. Any method such as flagging routes or using a foam marker would be appropriate. The maps and semivariance generated from the repeated data should be analyzed carefully. The features of the fairway should be carefully observed for identifying those features on the NDVI map. Areas of concern such as drought conditions, low fertility, or heavy cart and foot traffic should be noted for comparison to

Oklahoma State University Library

the maps. The semivariance analyses done for this study were not adequate, because of incomplete sensor coverage. Optical scanning should be performed several times with care taken to have complete sensor coverage. Semivariance analysis should then be done to better determine variability of the turfgrass.

Several scans of the same area, either on turfgrass or wheat, should be done on the same day to establish the repeatability of the experiment. If the same results occur from one test to the next, it can be assumed that the variability seen by the sensors is actually in the field, not in the electronics of the sensor or in the software programs.

BIBLIOGRAPHY

Cahn, M. D., J. W. Hummel, and B. H. Brouer. 1994. *Spatial Analysis of Soil Fertility for Site-Specific Crop Management*. Soil Sci. Soc. Am. J. 58:1240-1248.

Dusek, D. A., R. A. Jackson, and J. T. Musick. 1985. *Winter Wheat Vegetative Indices Calculated From Combinations of Seven Spectral Bands*. Remote Sens. Environ. Vol 18: p. 255-267. Elsevier Science Publishing Co. Inc.

Feekes, W. 1941. De Tarwe en haar milieu. Vers. XVII Tech. Tarwe Comm. Groningen, 560-1.

Golden Sofware, Inc., 1999. Surfer 7 User's Guide. Golden, CO.

Guertal, E. A. *Geostatistical Analysis of Forms of Soil Nitrogen in a Continuously Cropped Wheat Field.* Unpubl. Ph.D. diss. Oklahoma State University, Stillwater, OK.

Han, S., J. W. Hummel, C. E. Goering and M. D. Cahn. 1994. *Cell Size Selection for Site-specific Crop Management*. Trans. Am. Soc. Agr. Engrs. 37(1):19-26.

Hornbeck, R. W. 1975. Numerical Methods. Prentice-Hall, Inc. Englewood Cliffs, NJ.

Isaaks, E. H., and R. M. Srivastava. 1989. *An Introductions to Applied Geostatistics*. Oxford University Press, New York, NY.

Klemen, J. and E. Fagerlund. 1987. *Influence of Different Nitrogen and Irrigation Treatments on the Spectral Reflectance of Barley*. Remote Sens. Environ. 21: 1-14.

Knighton, R. E. and R. J. Wagenet. 1987. *Geostatistical Estimation of Spatial Structure*. In Continuum, Vol. 1. Water Resources Inst., Center for Environ. Res., Ithaca, NY.

Lersten, N. R. 1987. *Morphology and Anatomy of the Wheat Plant*, Wheat and Wheat Improvement, Second Edition, E. G. Heyne, editor. American Society of Agronomy, Inc., Crop Science Society of America, Inc., Soil Science Society of America, Inc., Madison, WI.

Raun, W. R., J. B. Solie, G. V. Johnson, M. L. Stone, R. W. Whitney, H. L. Lees, H. Sembiring, and S. B. Phillips. 1998. *Microvariability in Soil Test, Plant Nutrient, and Yield Parameters in Bermudagrass*. Soil Sci. Soc. Am. J. 62:683-690.

Royle, A., I. Clark, P. I. Brooker, H. Parker, A. Journel, J. M. Rendu, R. I. Sandefur, and P. Mousset-Jones. 1980. *Geostatistics*. McGraw-Hill, Inc., New York, NY.

Oklahoma State University Library

Russo, D., and W. A. Jury. 1987. *A Theoretical Study of the Estimation of the Correlation Scale in Spatially Variable Fields 2. Nonstationary Fields*. Water Resources Research, Vol. 23, No. 7, p. 169-1279.

Sawyer, J. E. 1994. *Concepts of Variable Rate Technology With considerations for Fertilizer Application*. J. Prod. Agric. 7:195-201.

Solie, J. B., W. R. Raun, R. W. Whitney, M. L. Stone, and J. D. Ringer. 1996. *Optical Sensor Based Field Element Size and Sensing Strategy for Nitrogen Application*. Trans. Am. Soc. Agr. Engrs. 39(6): 1983-1992.

Solie, J. B., W. R. Raun, and M. L. Stone. 1999. *Submeter Spatial Variability of Selected Soil and Bermudagrass Production Variables*. Soil Sci. Soc. Am. J. 63:1724-1733.

Stone, M. L., J. B. Solie, R. W. Whitney, W. R. Raun, and H. L. Lees. 1996. *Sensors For Detection of Nitrogen in Winter Wheat*. Paper No. SAE 961757. Presented at the International Off-Highway & Powerplant Congress & Exposition, August 26-28, 1996. SAE. Warrendale, PA.

Tabor, J. A., A. W. Warrick, D. E. Meyers and D. A. Pennington. 1985. *Spatial Variability of NO_3-N in Irrigated Cotton: II. Soil NO_3-N and Correlated Variables*. Soil Sci. Soc. Am. J. 49:390-394.

Taylor, S. L. 1996. *Bermudagrass Forage Yield Response to High Rates of Applied Urea and Ammonium Nitrate and the Use of Spectral Radiance for Estimating Nitrogen Deficiencies and Soil Variability*. Unpubl. M. S. Thesis. Oklahoma State University, Stillwater, OK.

Wanjura, D. F. and J. L. Hatfield. 1987. *Sensitivity of Spectral Vegetative Indices to Crop Biomass*. Trans. Am. Soc. Agr. Engrs. 30(3): 810-816.

West, C. P., A. P. Mallarino, W. F. Wedin and D. B. Marx. 1989. *Spatial Variability of Soil Chemical Properties in Grazed Pastures*. Soil Sci. Soc. Am. J. 53:784-789.

Appendix A

Semivariograms

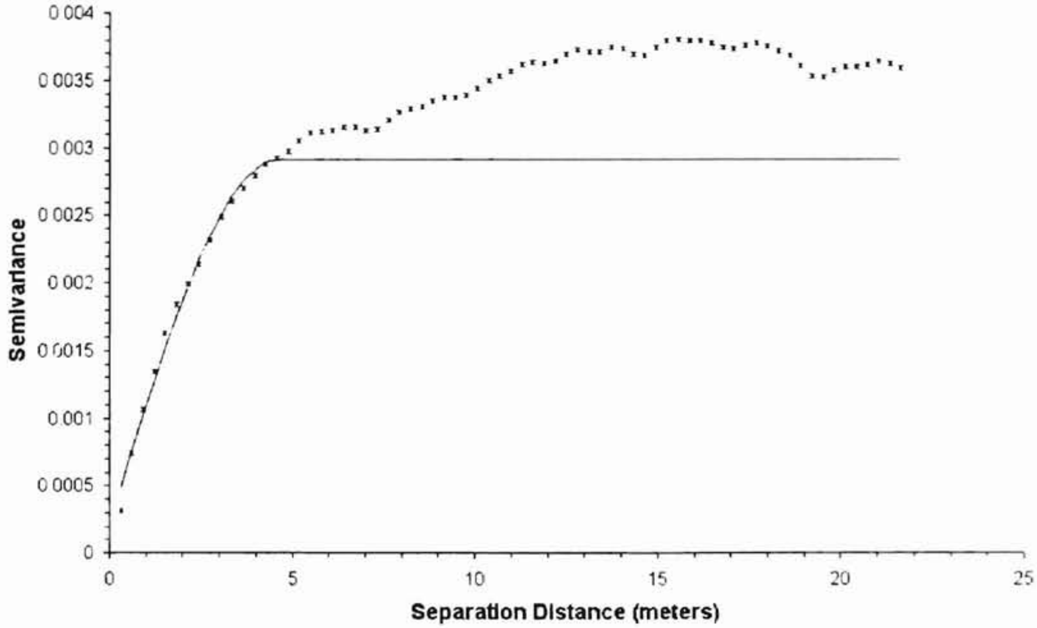

Figure 16: Perkins, OK, January 21, 2000 East-West

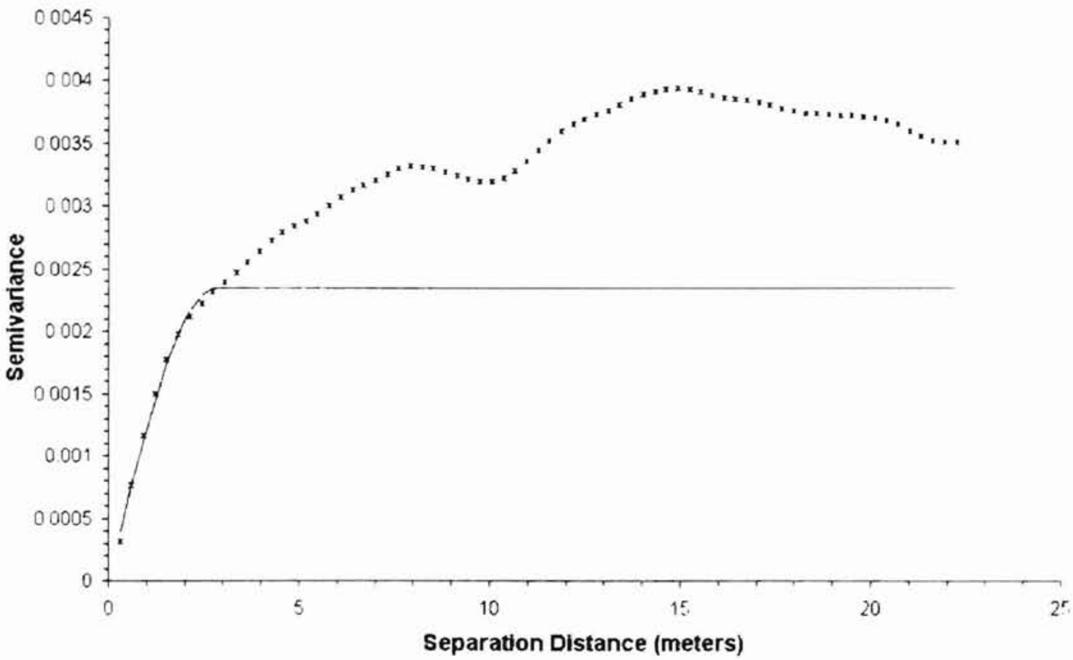

Figure A2: Perkins, OK, January 21, 2000 North-South

Appendix A, Continued

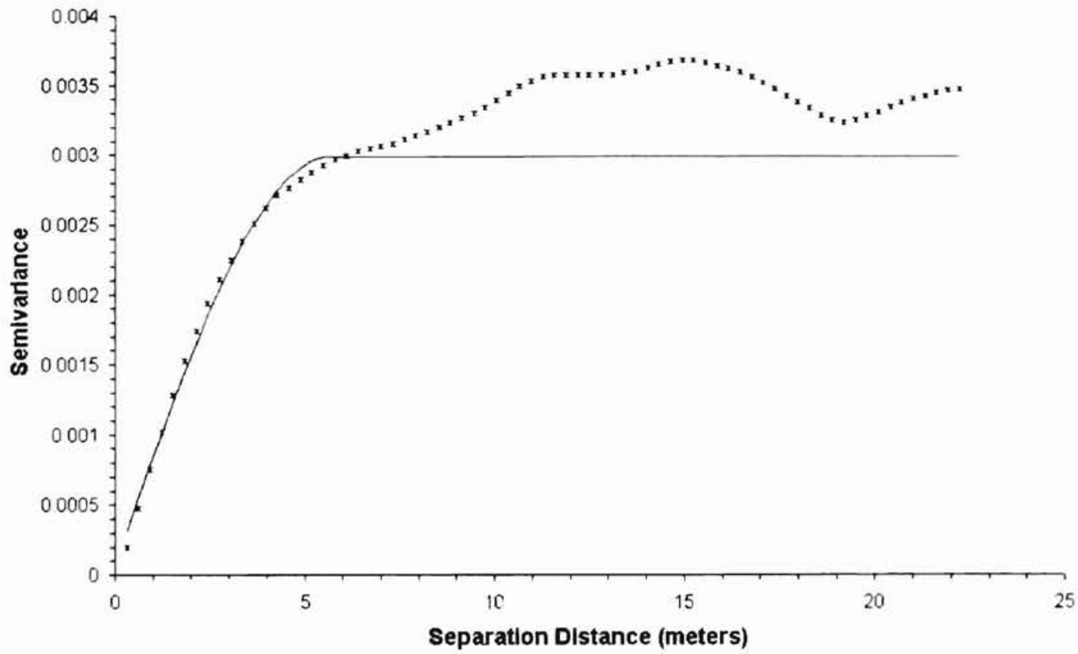

Figure A3: Perkins, OK, February 8, 2000, East-West

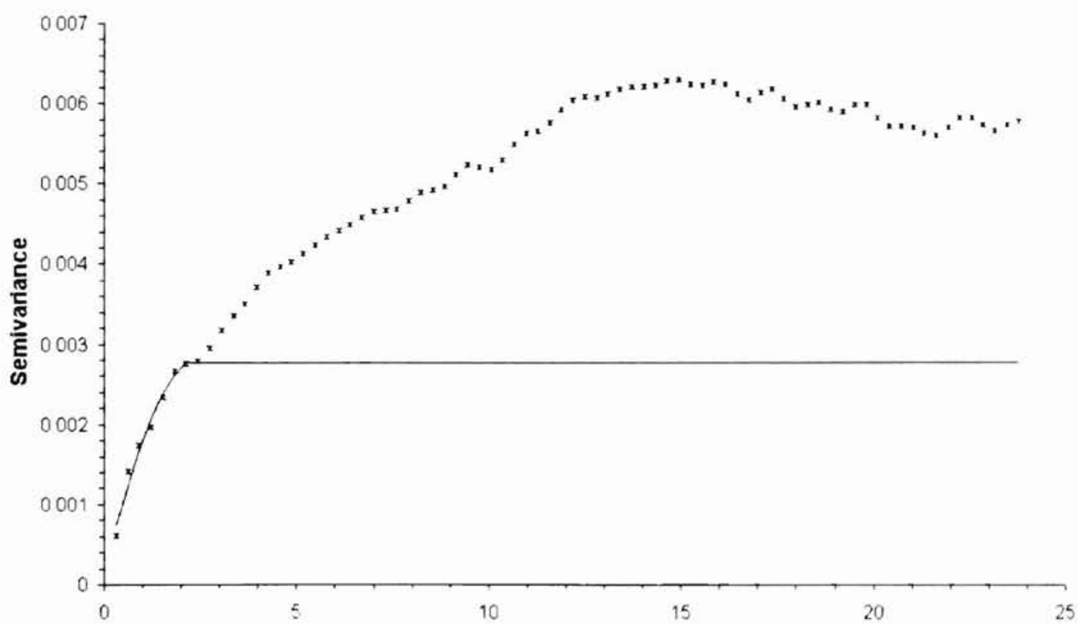

Figure A4: Perkins, OK, February 8, 2000, North-South

Appendix A, Continued

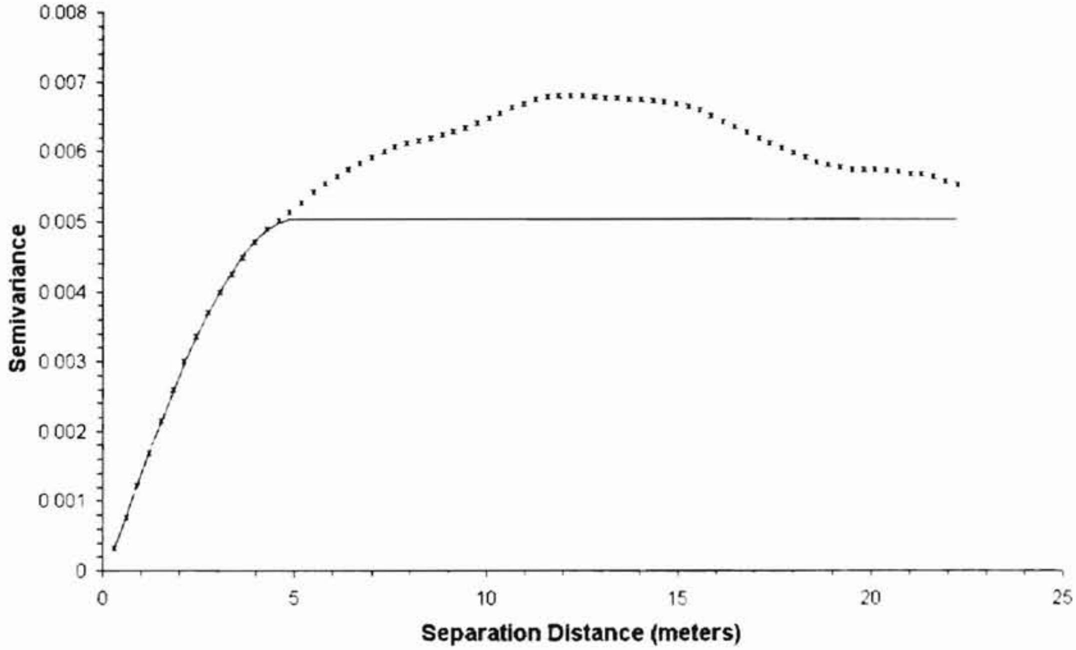

Figure A5: Perkins, OK, February 29, 2000, East-West

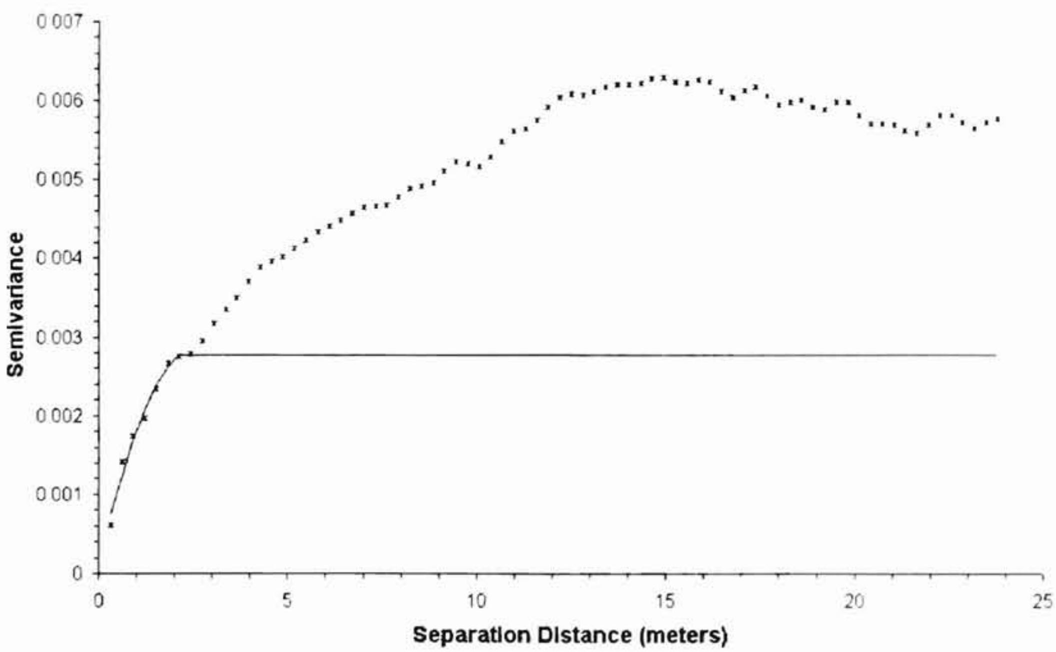

Figure A6: Perkins, OK, February 29, 2000, North-South

Appendix A, Continued

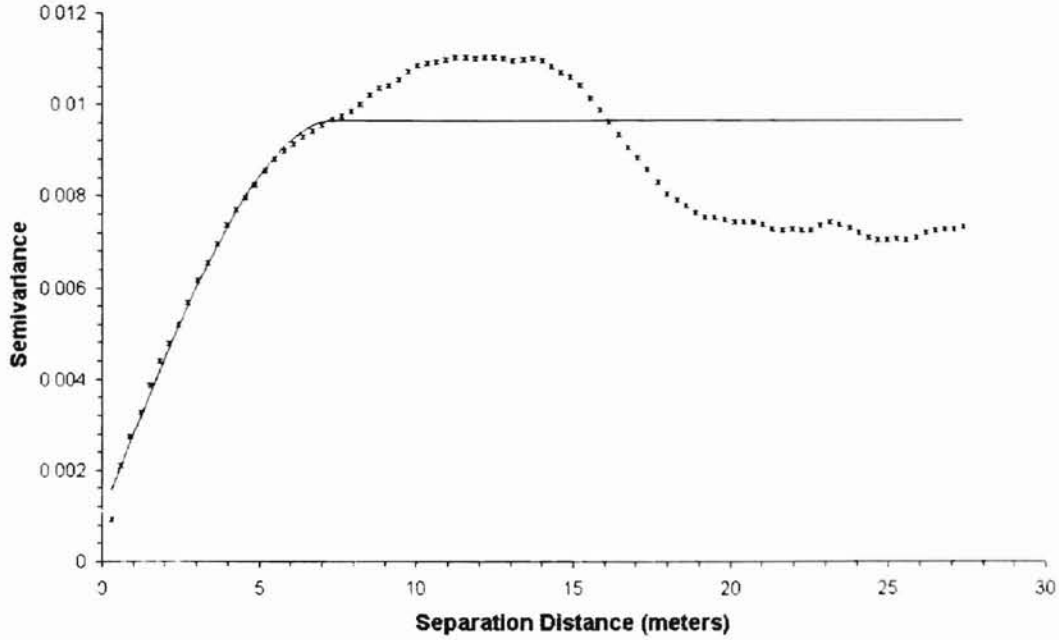

Figure A7: Terrace Plot, Lucien, OK, February 29, 2000, East-West

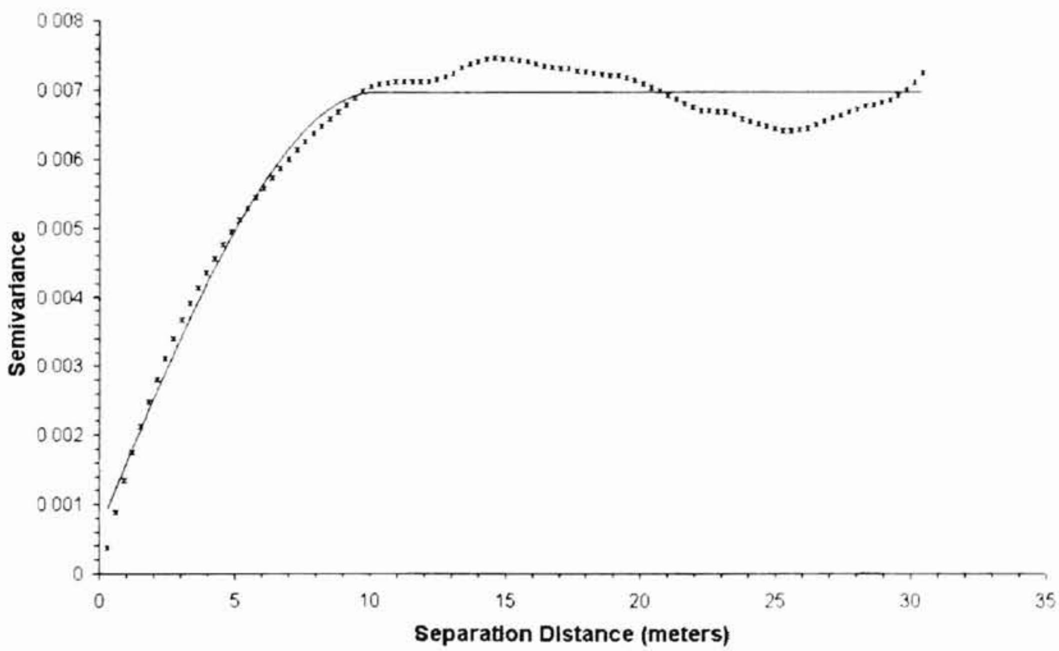

Figure A8: Terrace Plot, Lucien, OK, February 29, 2000, North-South

Appendix A, Continued

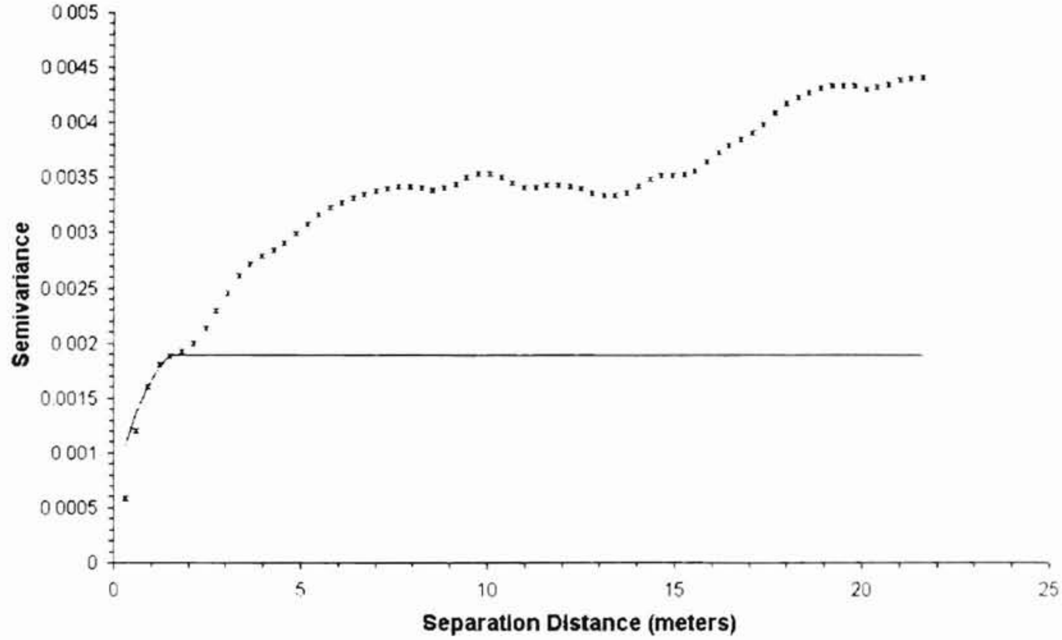

Figure A9: Flat Plot, Lucien, OK, February 29, 2000, East-West

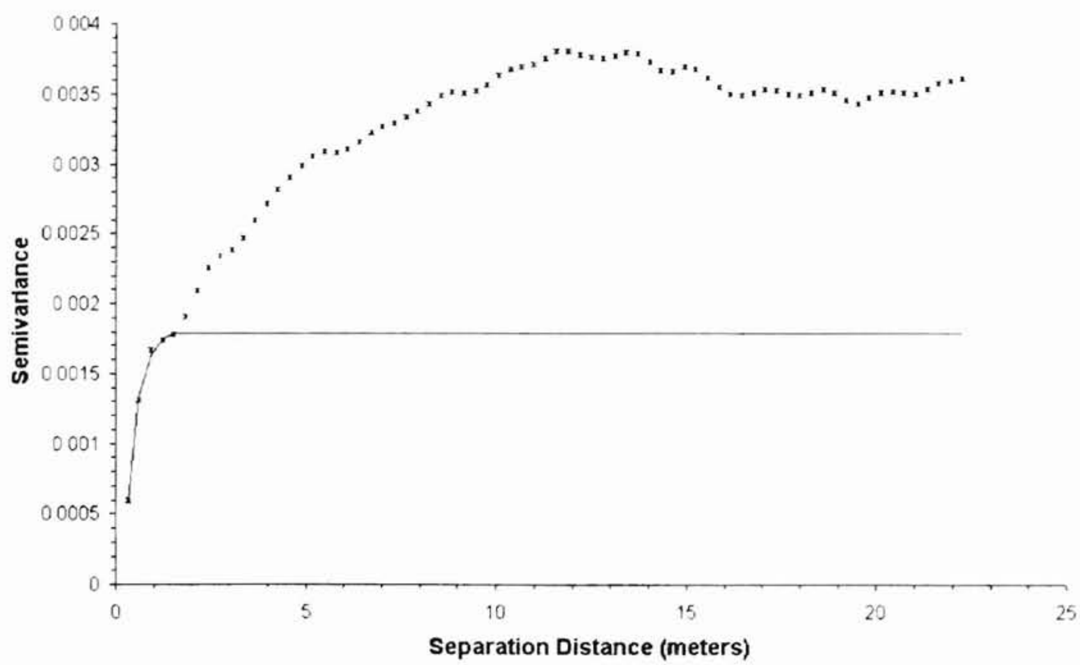

Figure A10: Flat Plot, Lucien, OK, February 29, 2000, North-South

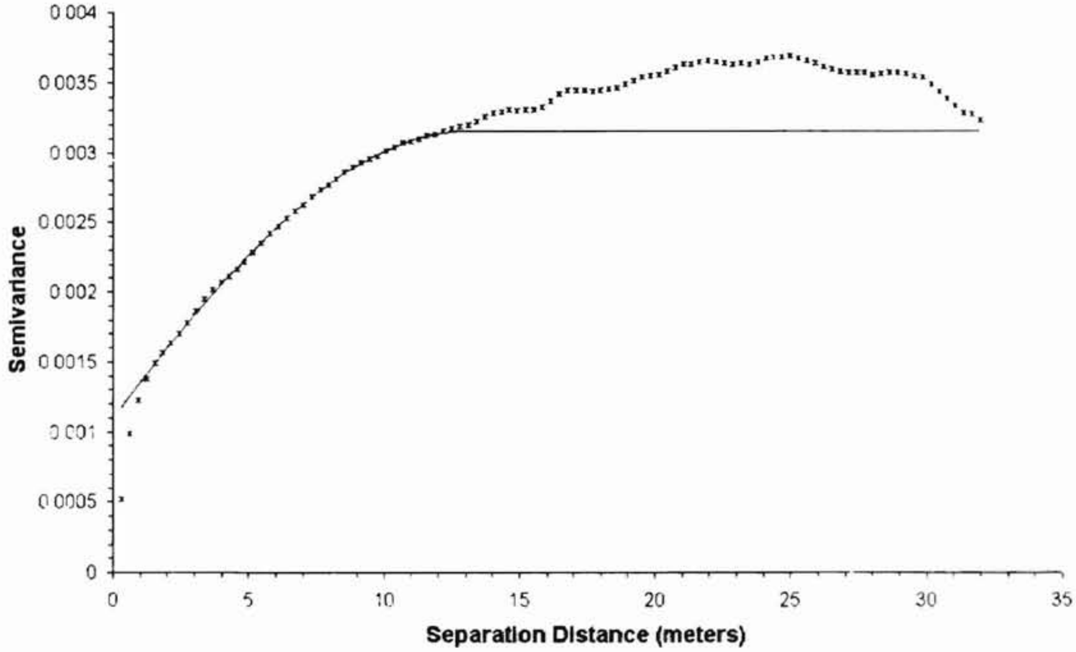

Figure A11: Lake Carl Blackwell Plot, Stillwater, OK, February 10, 2000, East-West

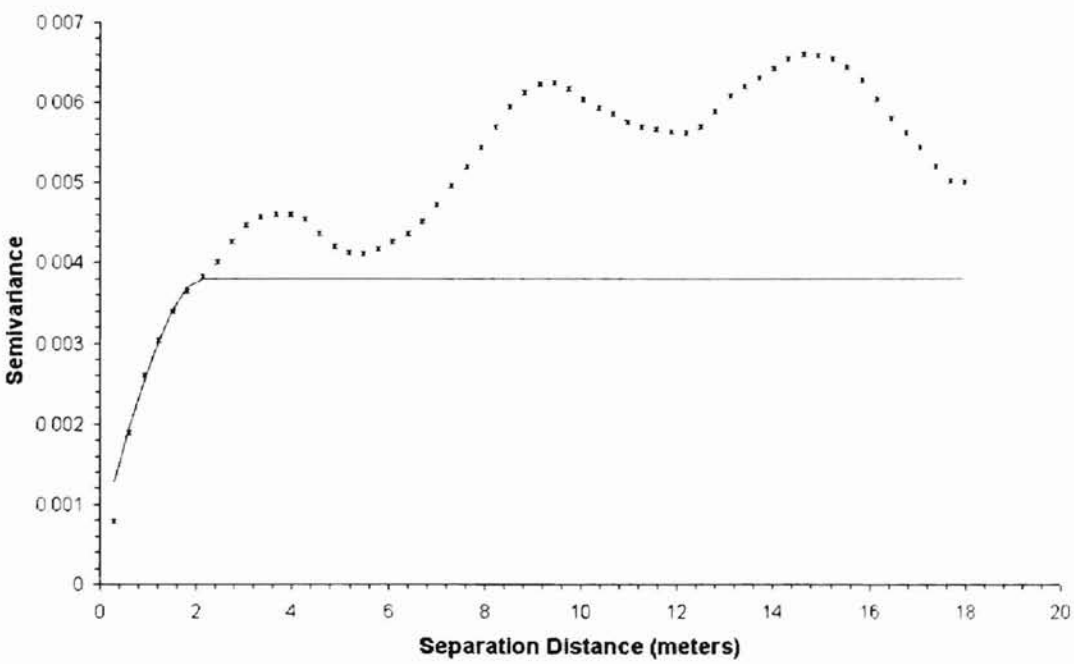

Figure A12: Lake Carl Blackwell Plot, Stillwater, OK, February 10, 2000, North-South

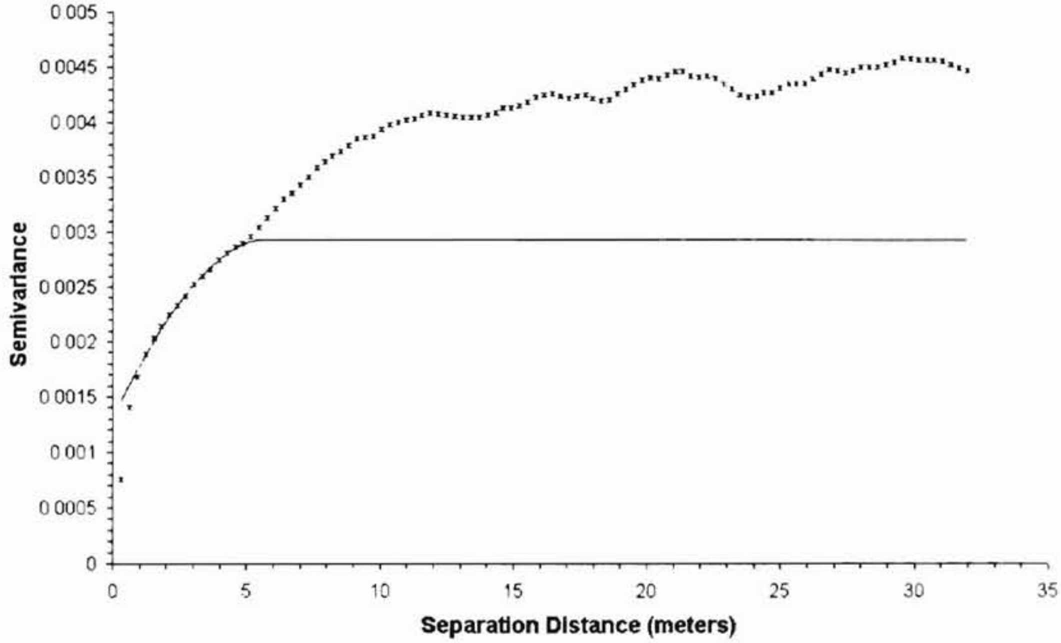

Figure A13: Lake Carl Blackwell Plot, Stillwater, OK, February 29, 2000, East-West

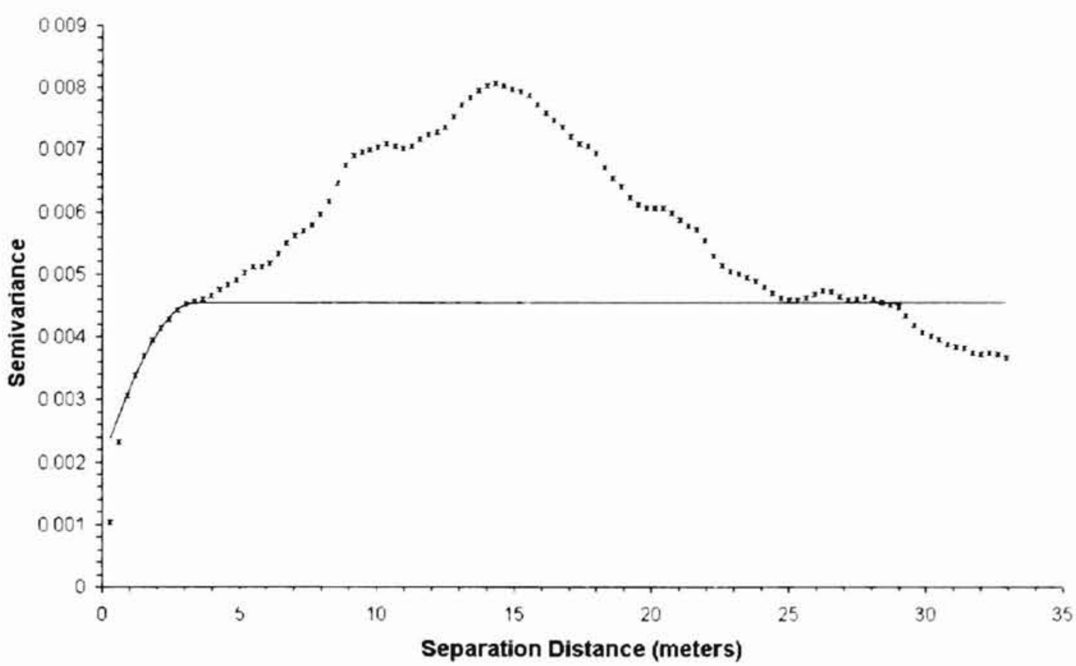

Figure A14: Lake Carl Blackwell Plot, Stillwater, OK, February 29, 2000, North-South

Appendix A, Continued

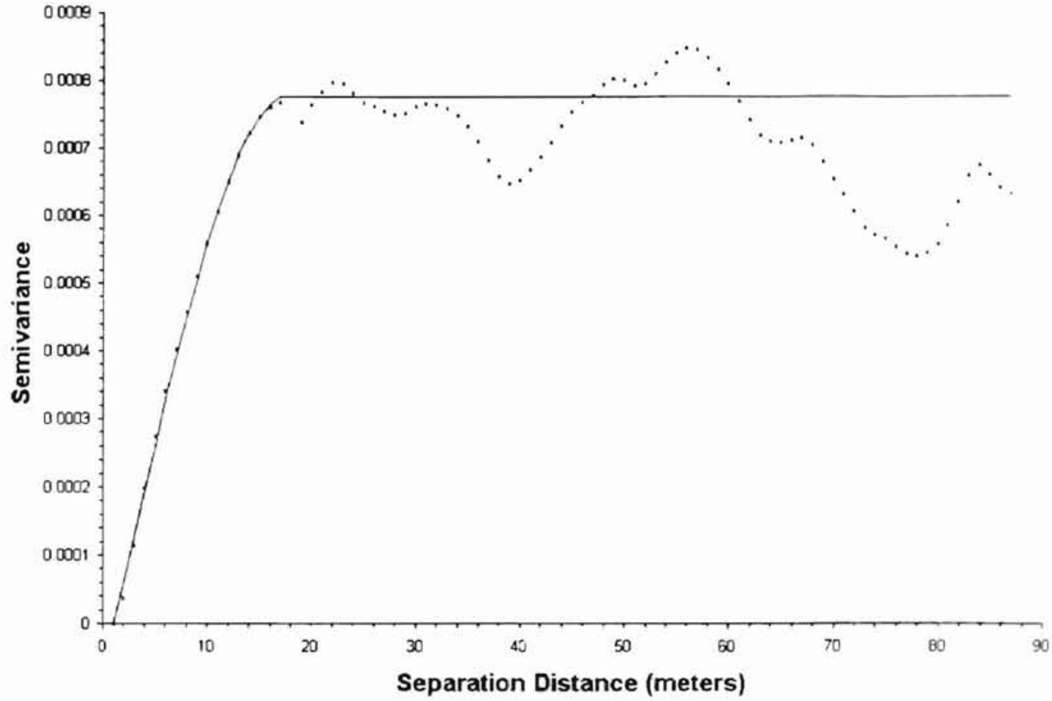

Figure A15: Karsten Creek Fairway 13, Northwest Section, Stillwater, OK, June 13, 2000, East-West

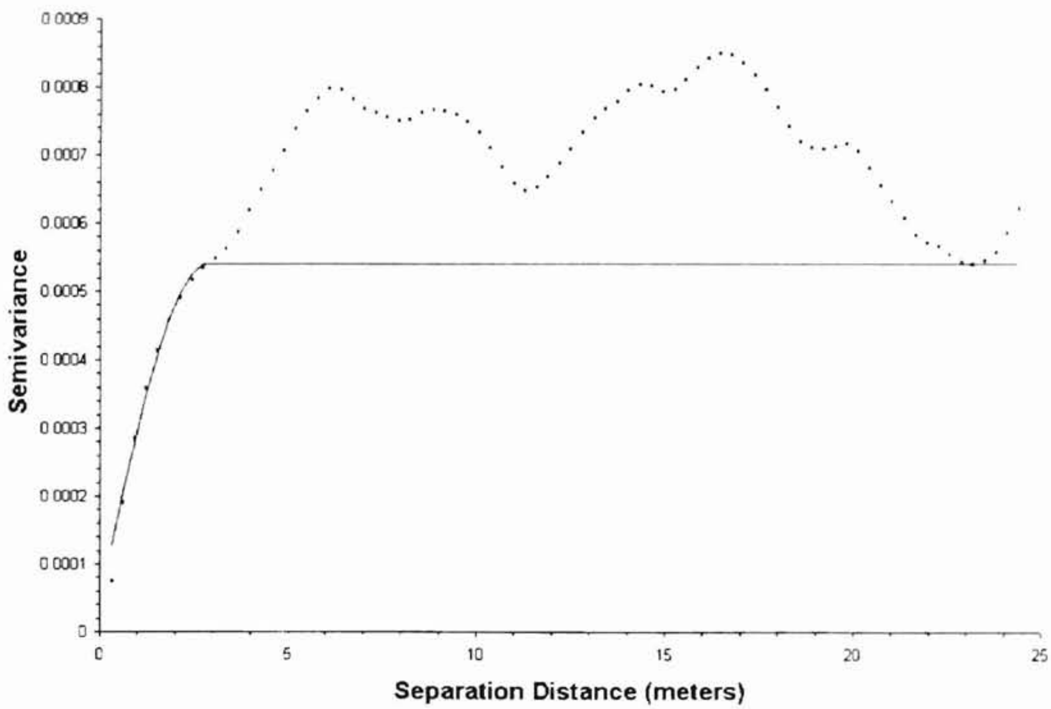

Figure A16: Karsten Creek Fairway 13, Northwest Section, Stillwater, OK, June 13, 2000, North-South

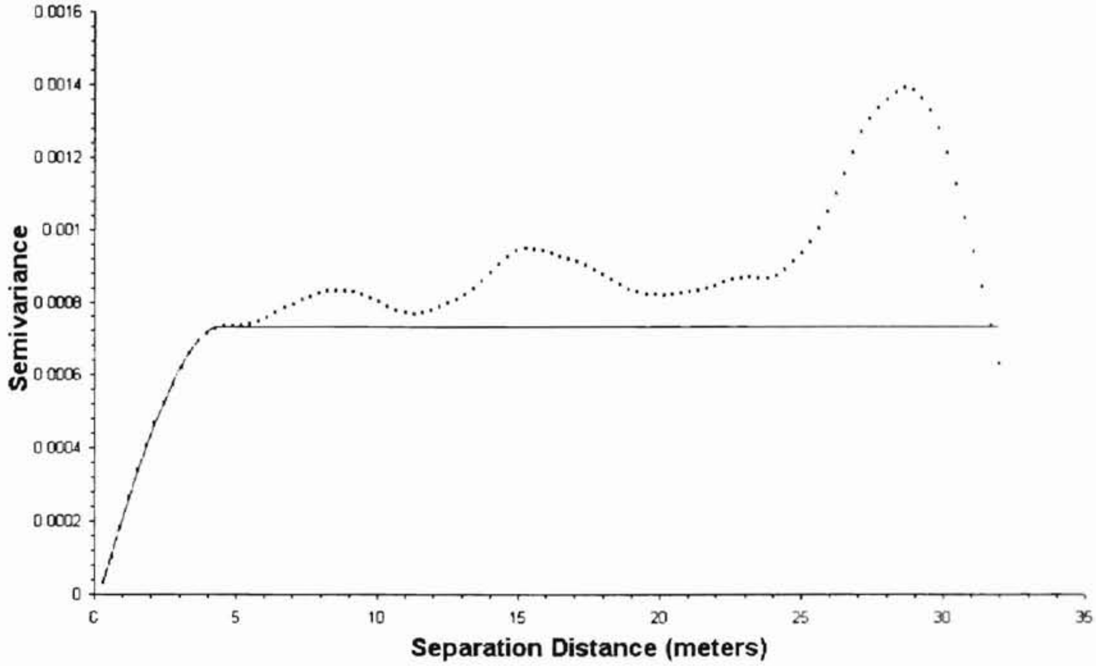

Figure A17: Karsten Creek Fairway 13, Middle Section, Stillwater, OK, June 13, 2000, East-West

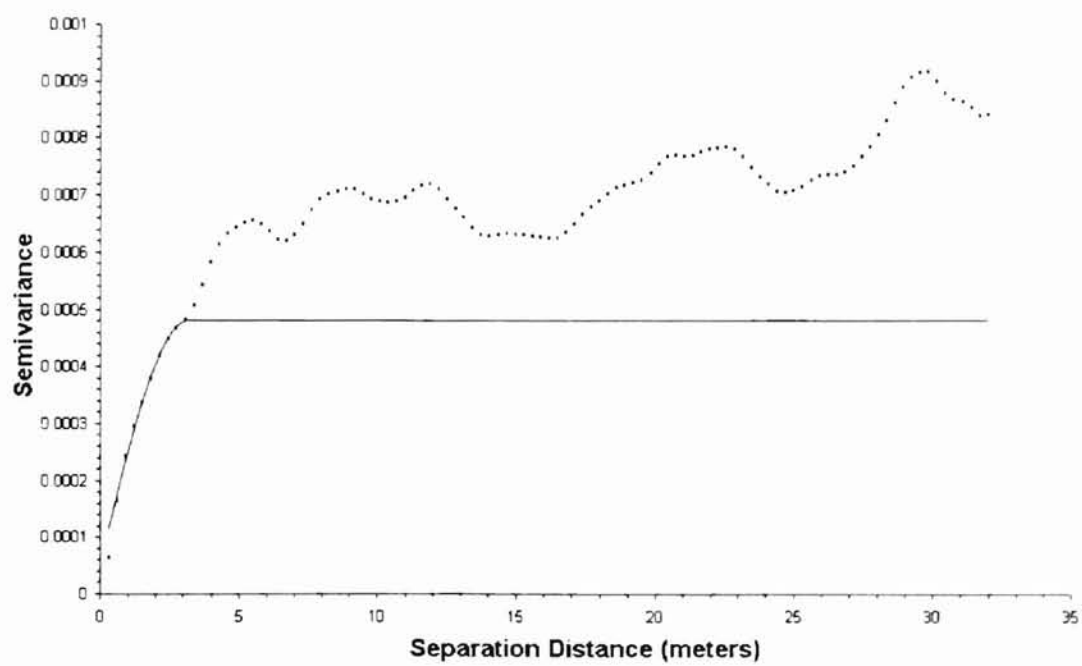

Figure A18: Karsten Creek Fairway 13, Middle Section, Stillwater, OK, June 13, 2000, North-South

Appendix A, Continued

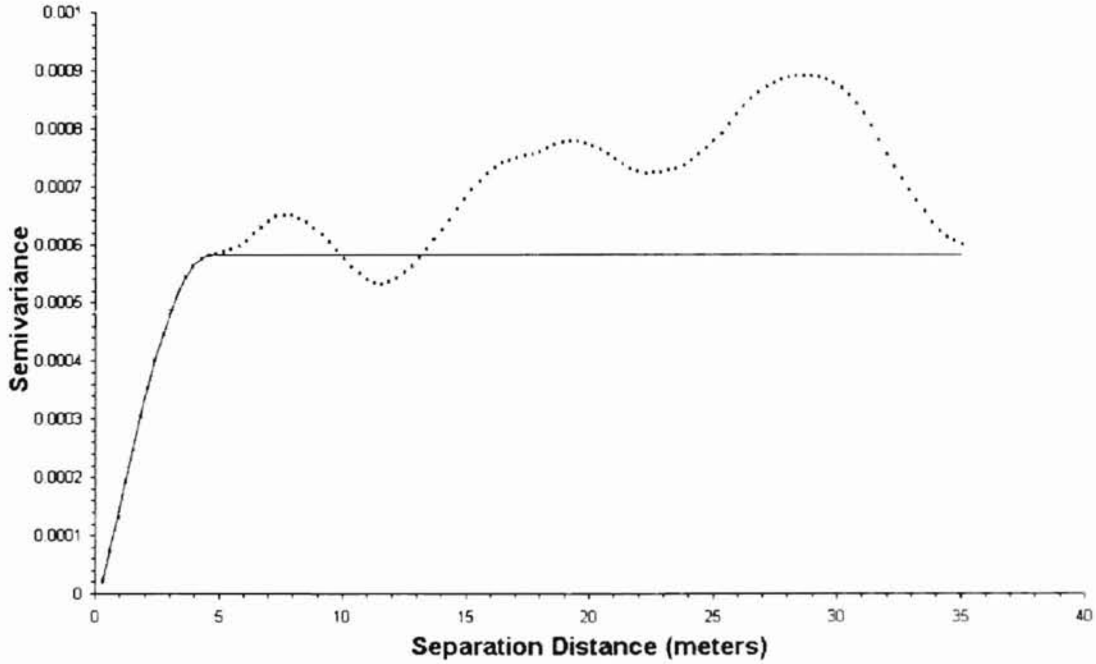

Figure A19: Karsten Creek Fairway 13, Southeast Section, Stillwater, OK, June 13, 2000, East-West

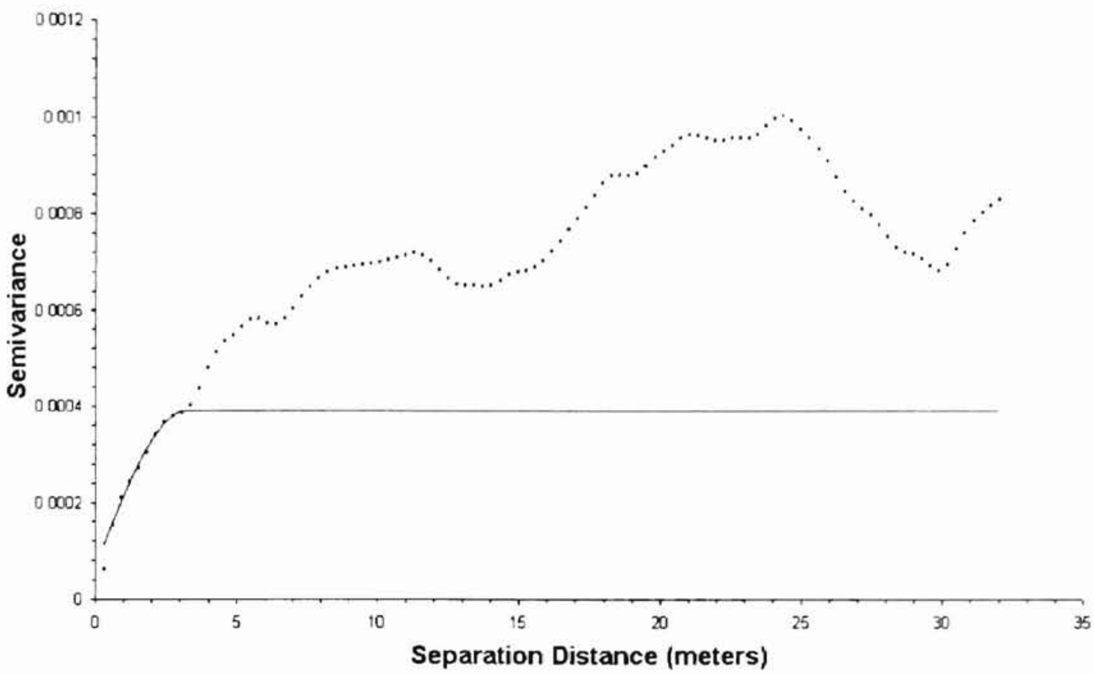

Figure A20: Karsten Creek Fairway 13, Southeast Section, Stillwater, OK, June 13, 2000, North-South

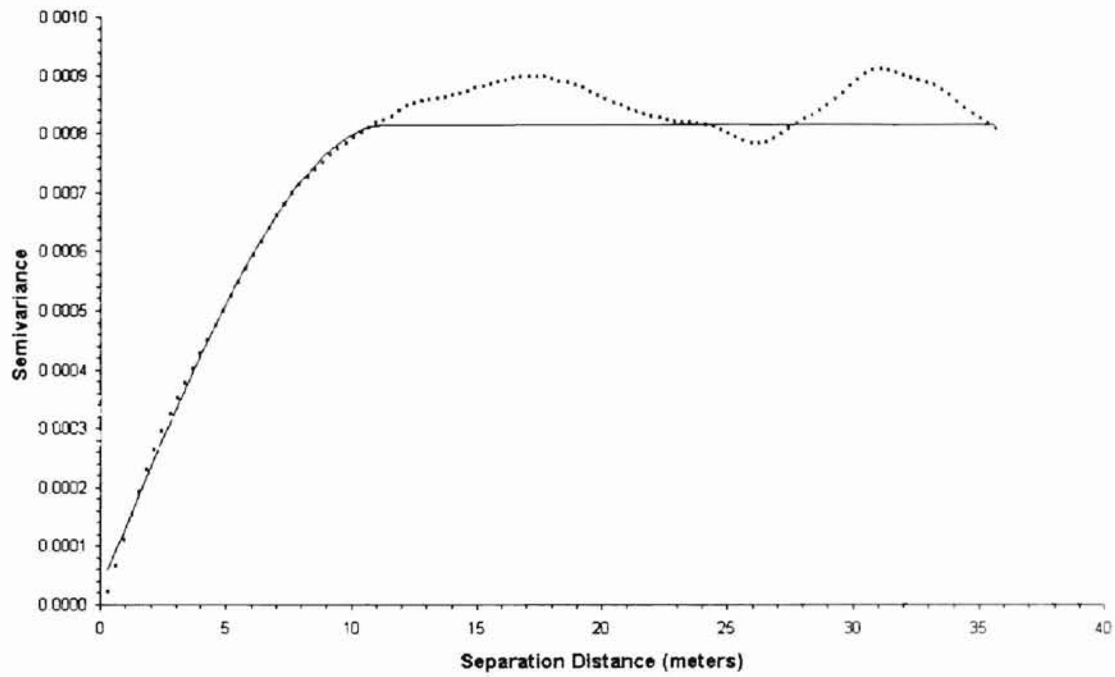

Figure A21: Karsten Creek Fairway 12, North Section, Stillwater, OK, June 20, 2000, East-West

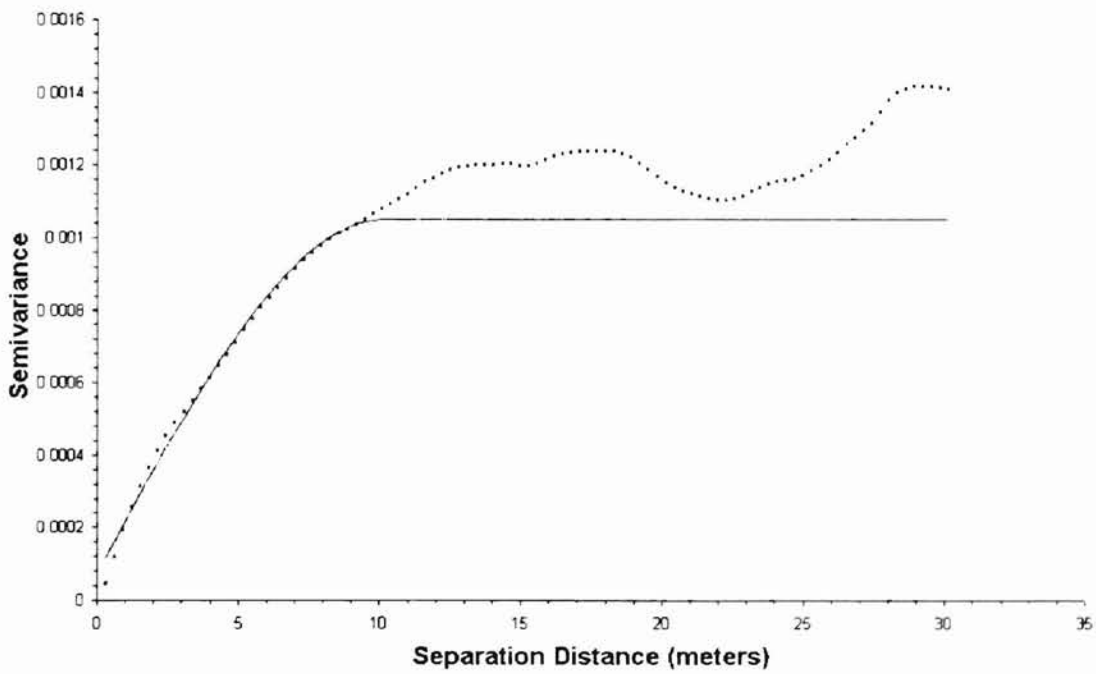

Figure A22: Karsten Creek Fairway 12, North Section, Stillwater, OK, June 20, 2000, North-South

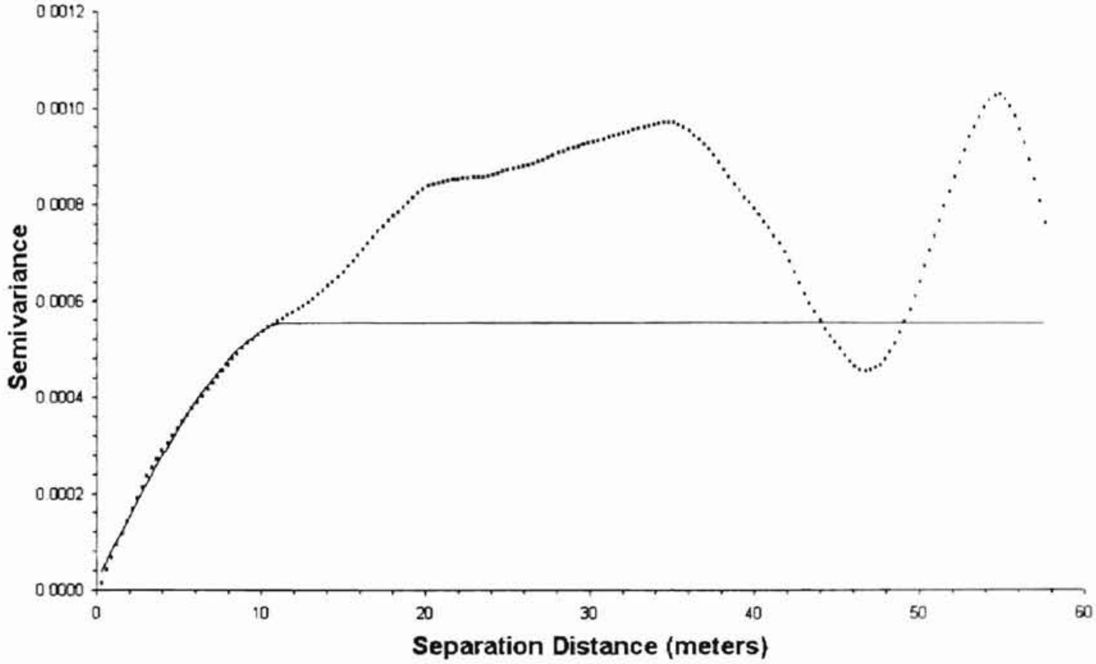

Figure A23: Karsten Creek Fairway 12, South Section, Stillwater, OK, June 20, 2000, East-West

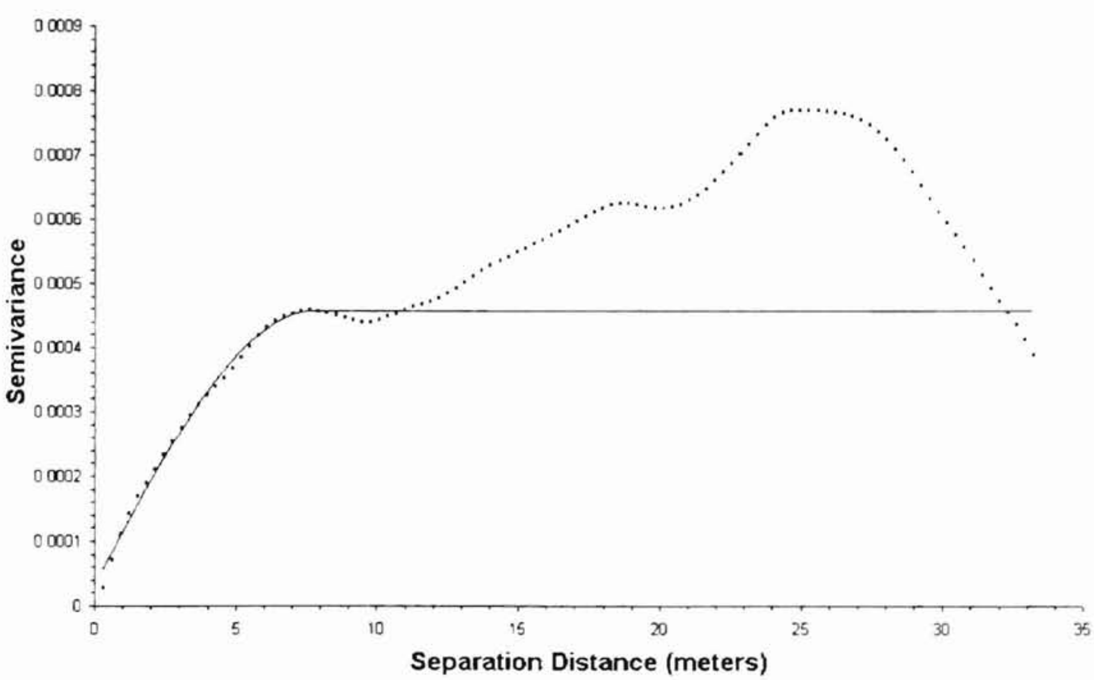

Figure A24: Karsten Creek Fairway 12, South Section, Stillwater, OK, June 20, 2000, North-South

Appendix B

Surface Maps

All surface maps are at 0.30 by 0.30 m resolution. Note that all scales of NDVI are **NOT** equal. The data that are used to generate these maps in SSToolbox are the same data used to perform the semivariance analysis.

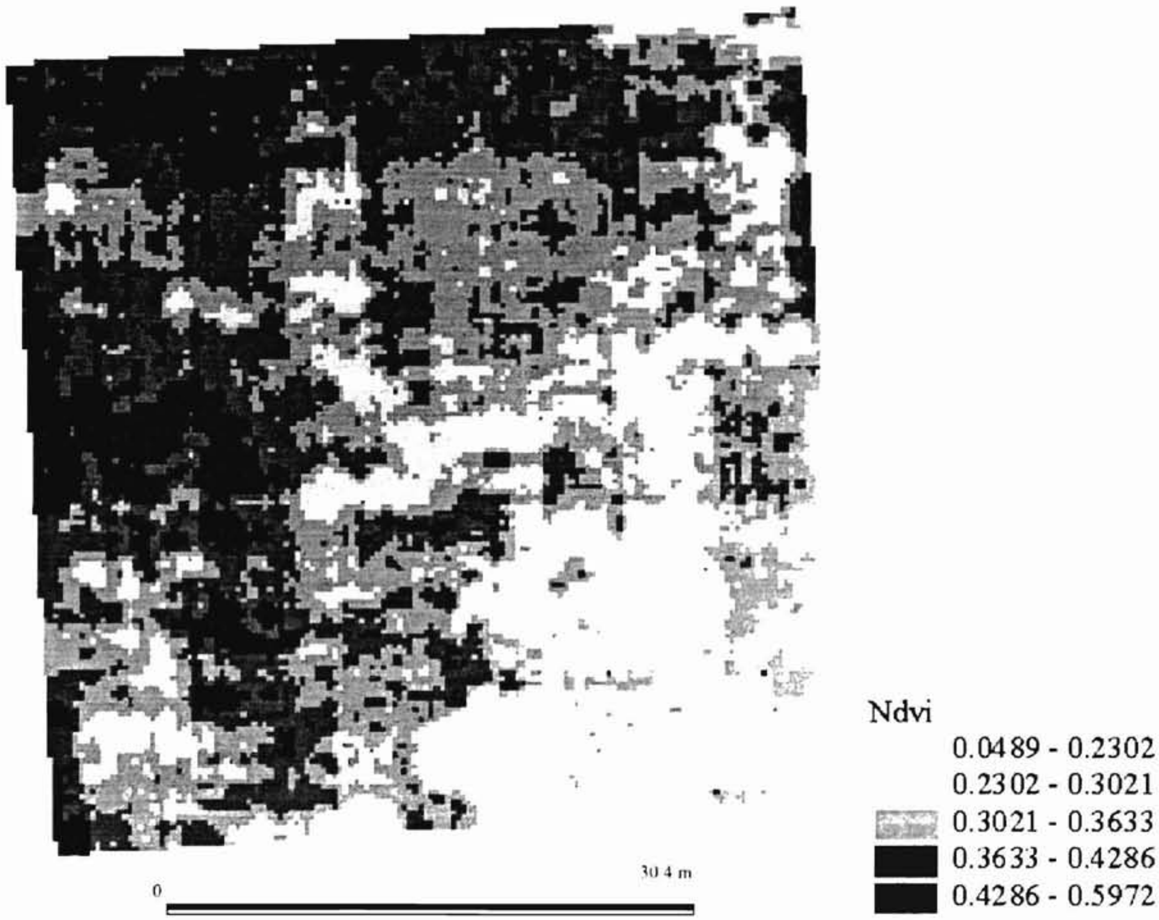

Figure B1: Perkins, OK, January 21, 2000

Appendix B, Continued

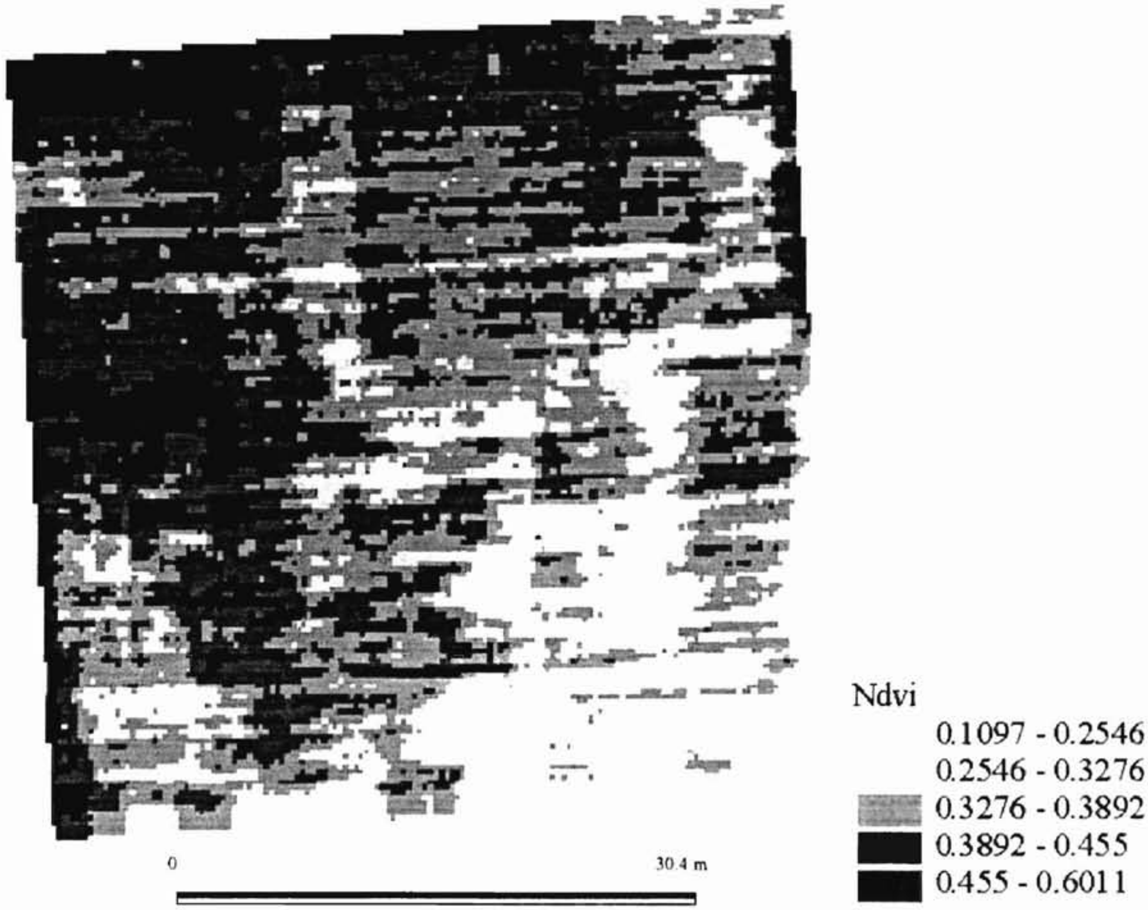

Figure B2: Perkins, OK, February 8, 2000

Appendix B, Continued

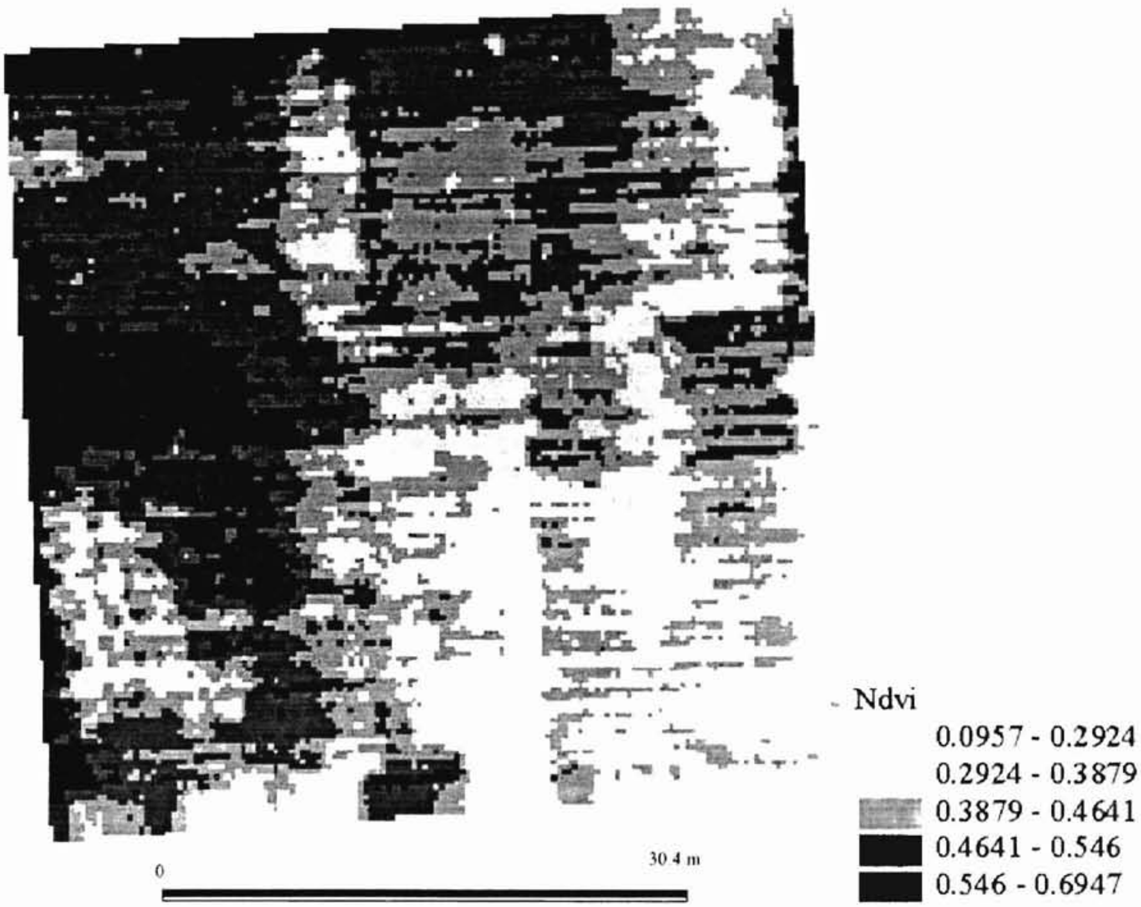

Figure 17: Perkins, OK, February 29, 2000

Appendix B, Continued

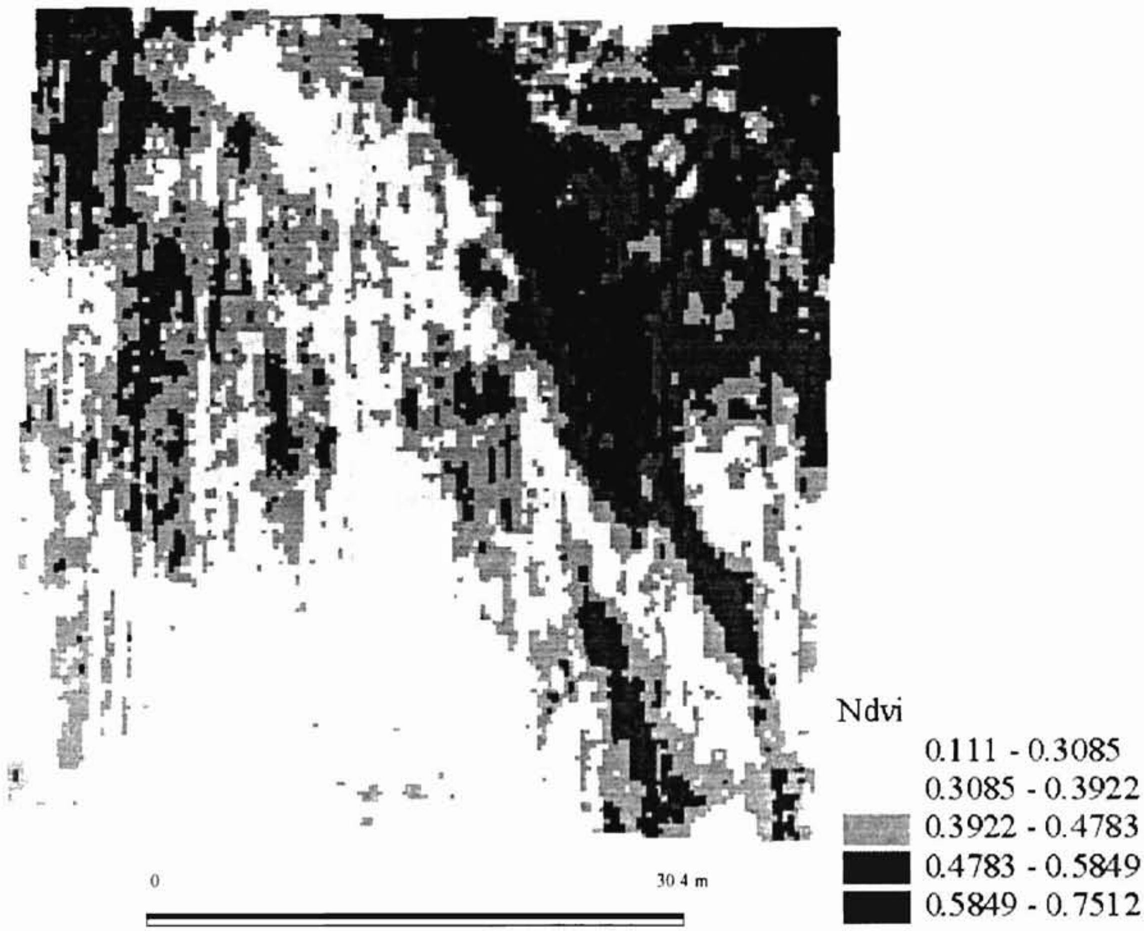

Figure 18: Terrace Plot, Lucien, OK, February 29, 2000

Appendix B, Continued

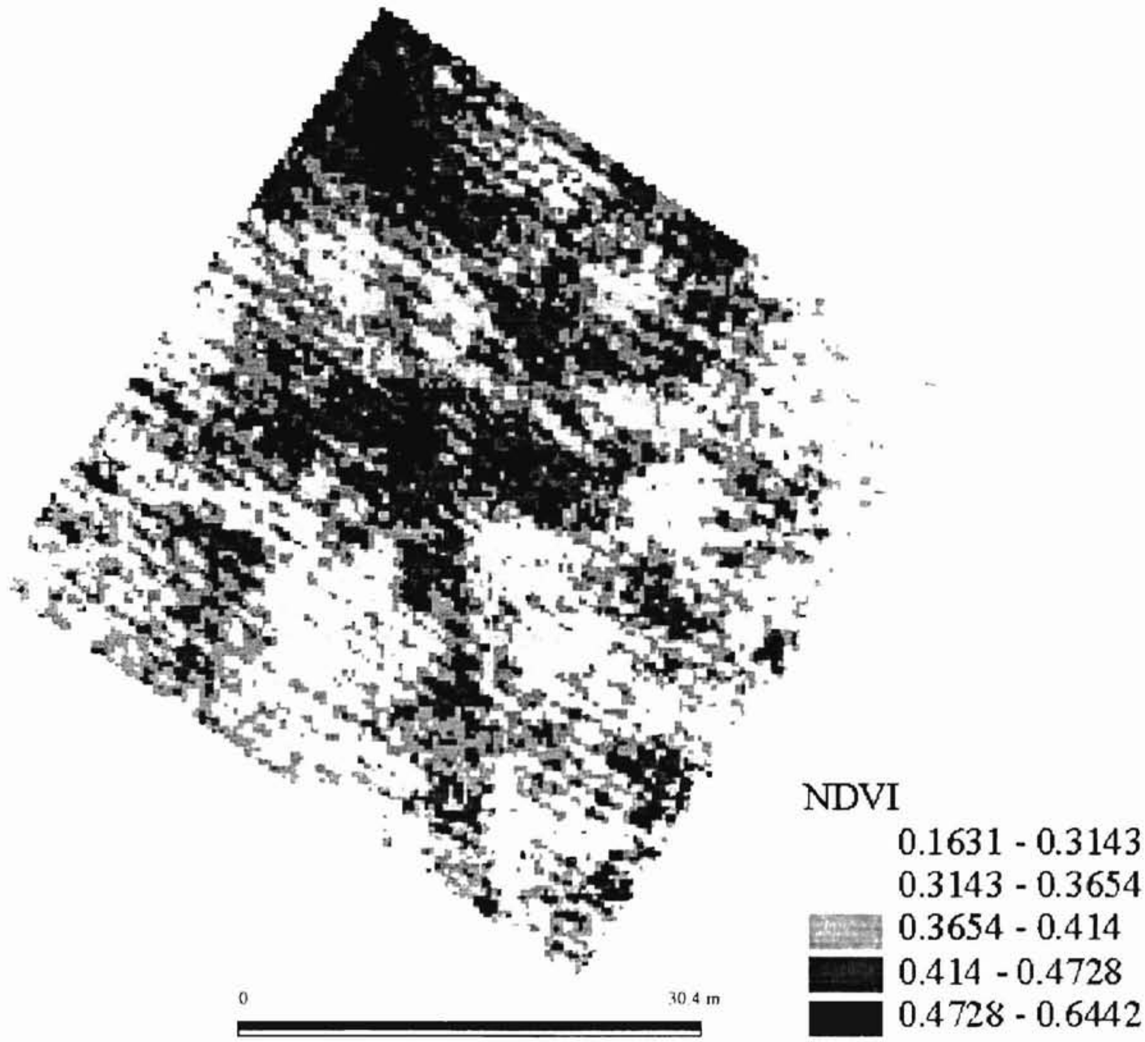

Figure 19: Flat Plot, Lucien, OK, February 29, 2000

Appendix B, Continued

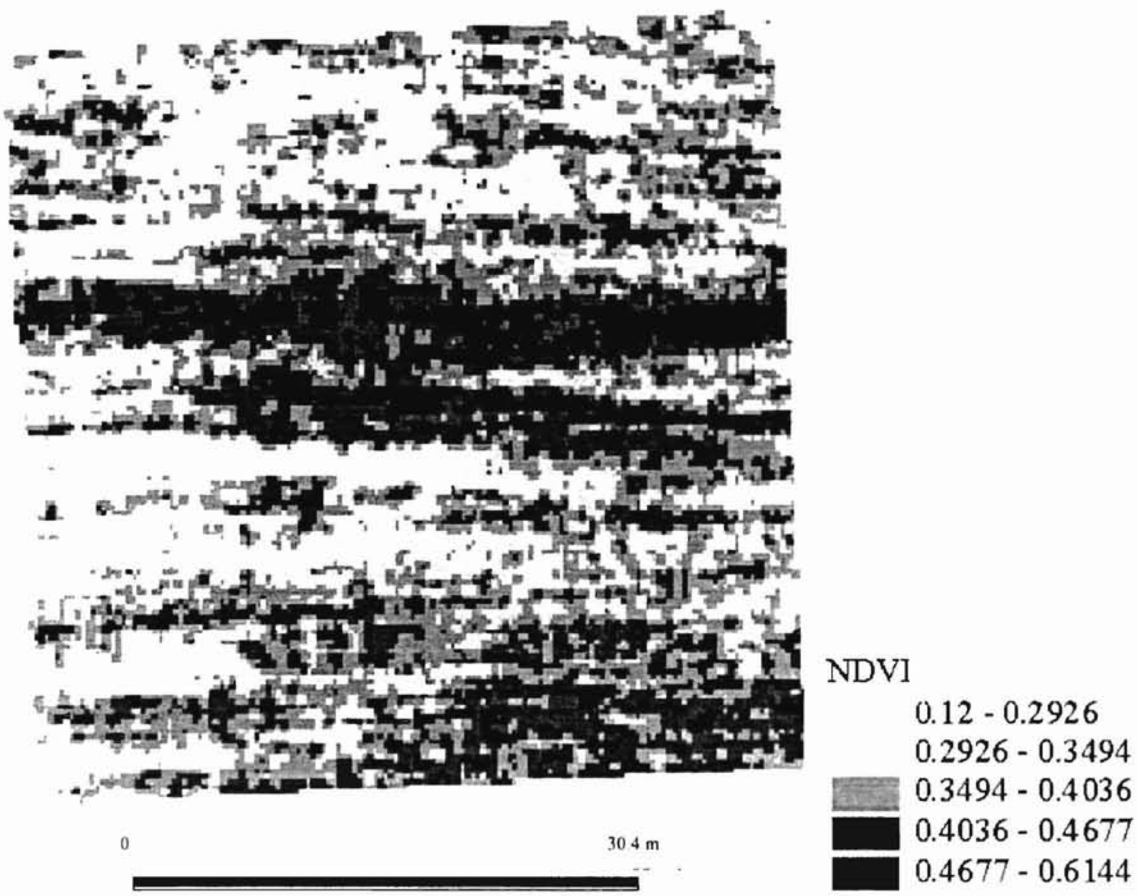

Figure 20: Lake Carl Blackwell Plot, Stillwater, OK, February 10, 2000

Appendix B, Continued

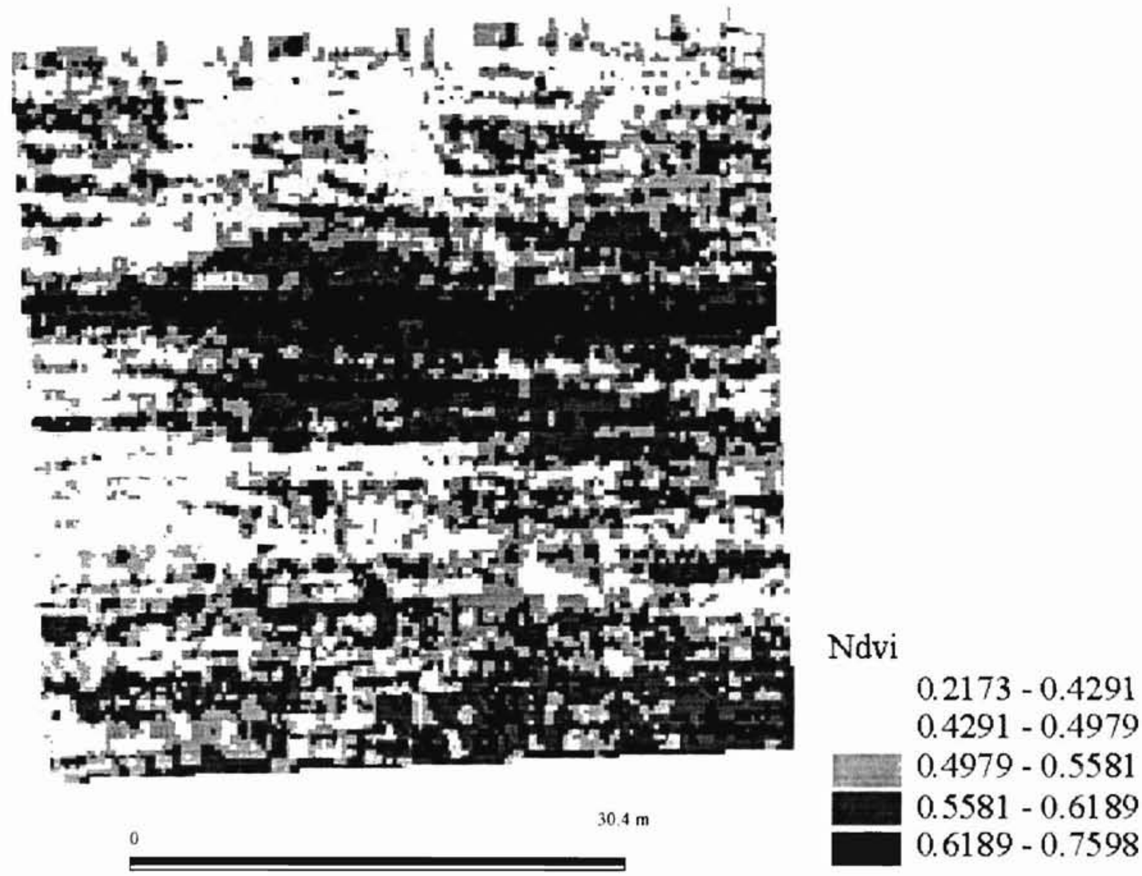

Figure B7: Lake Carl Blackwell Plot, Stillwater, OK, February 29, 2000

Appendix B, Continued

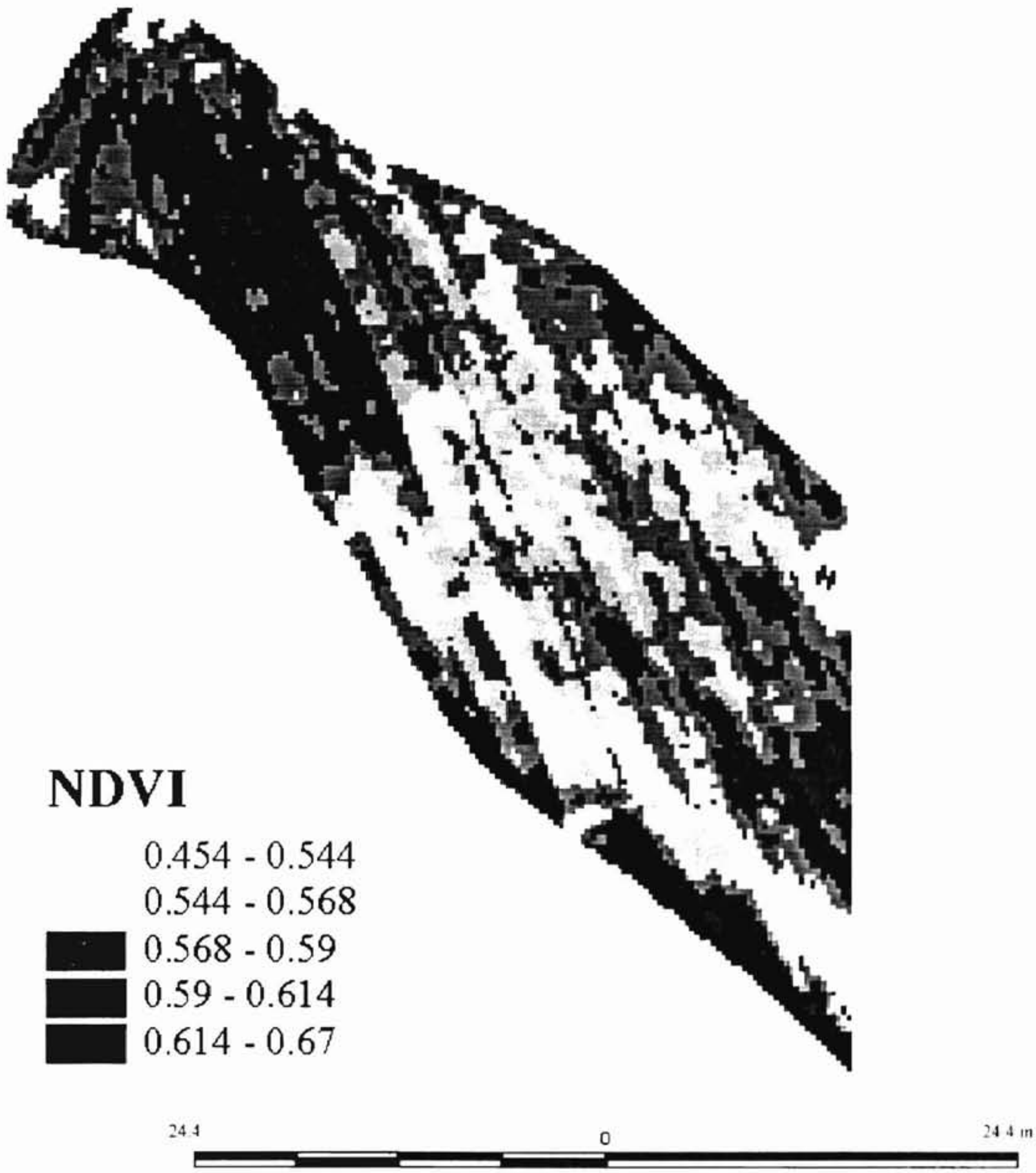

Figure B8: Karsten Creek Fairway 13, Northwest Section, Stillwater, OK, June 13, 2000

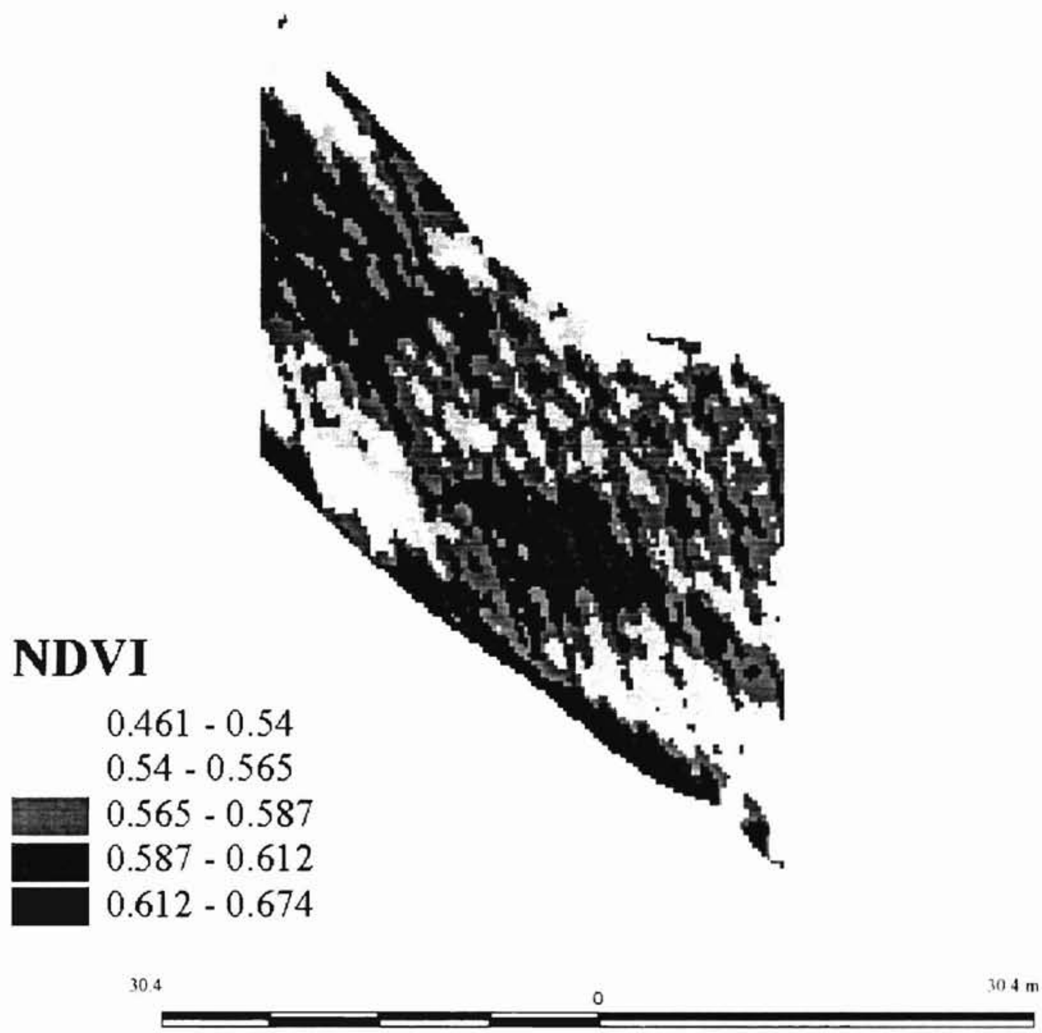

Figure B9: Karsten Creek Fairway 13, Middle Section, Stillwater, OK, June 13, 2000

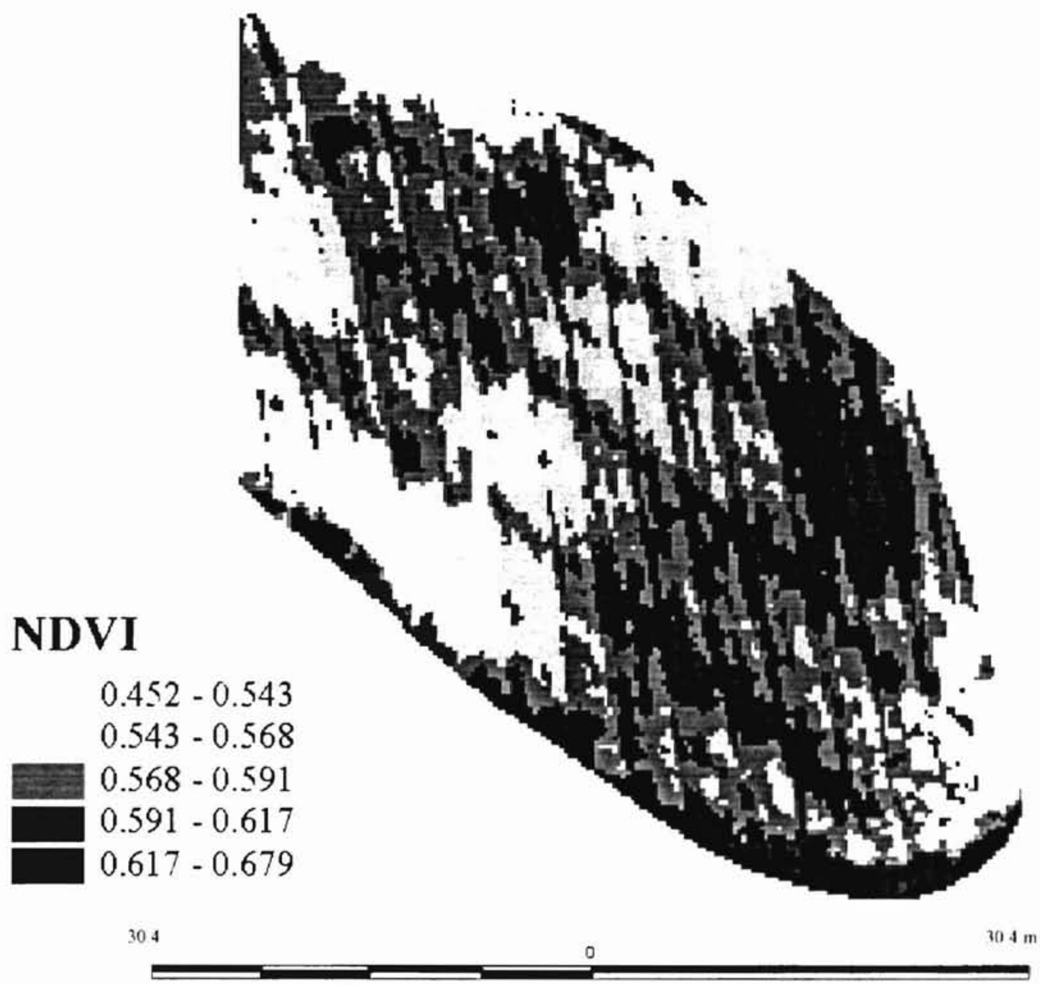

Figure B10: Karsten Creek Fairway 13, Southeast Section, Stillwater, OK, June 13, 2000

Appendix B, Continued

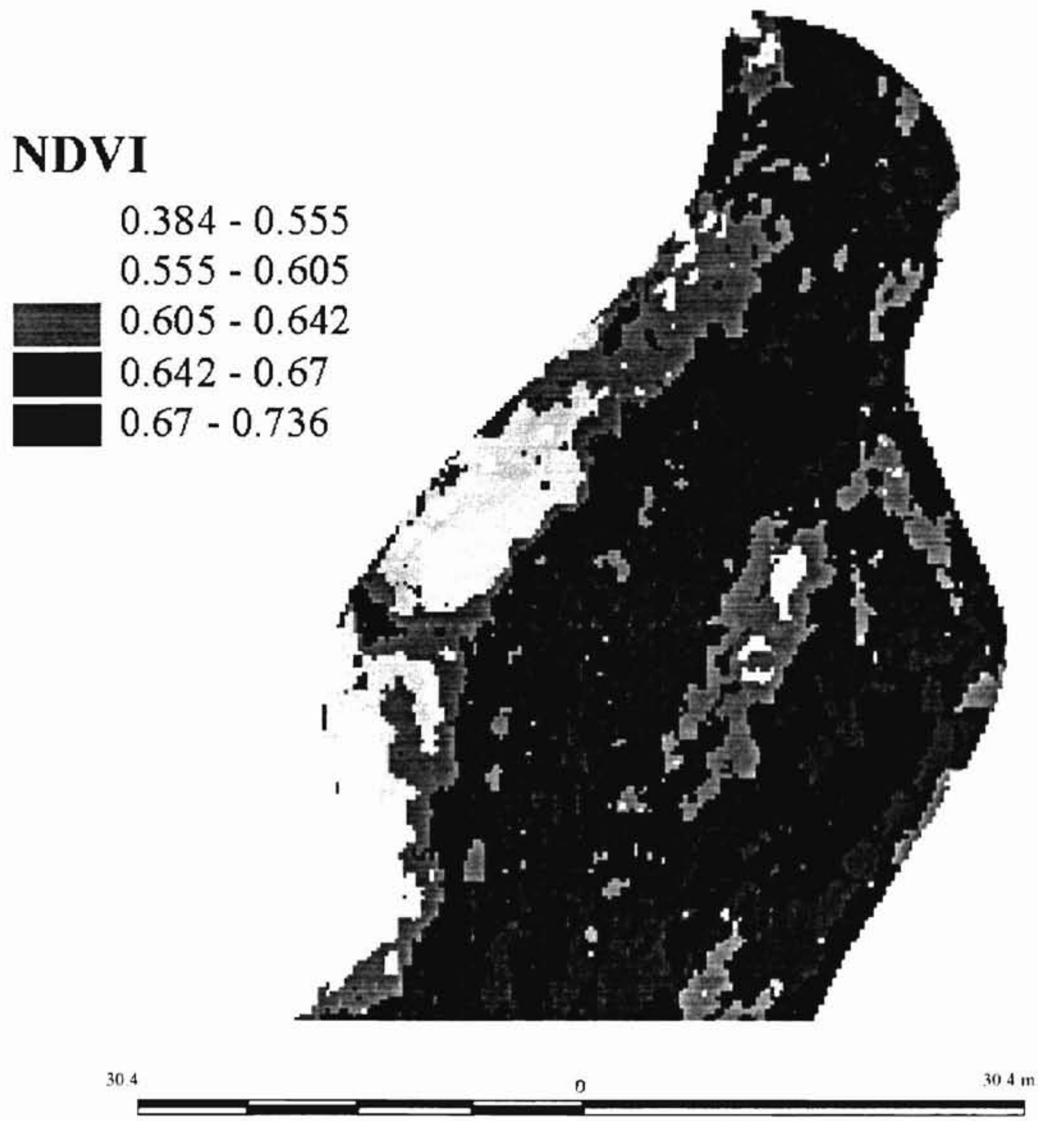

Figure B11: Karsten Creek Fairway 12, North Section, Stillwater, OK, June 13, 2000

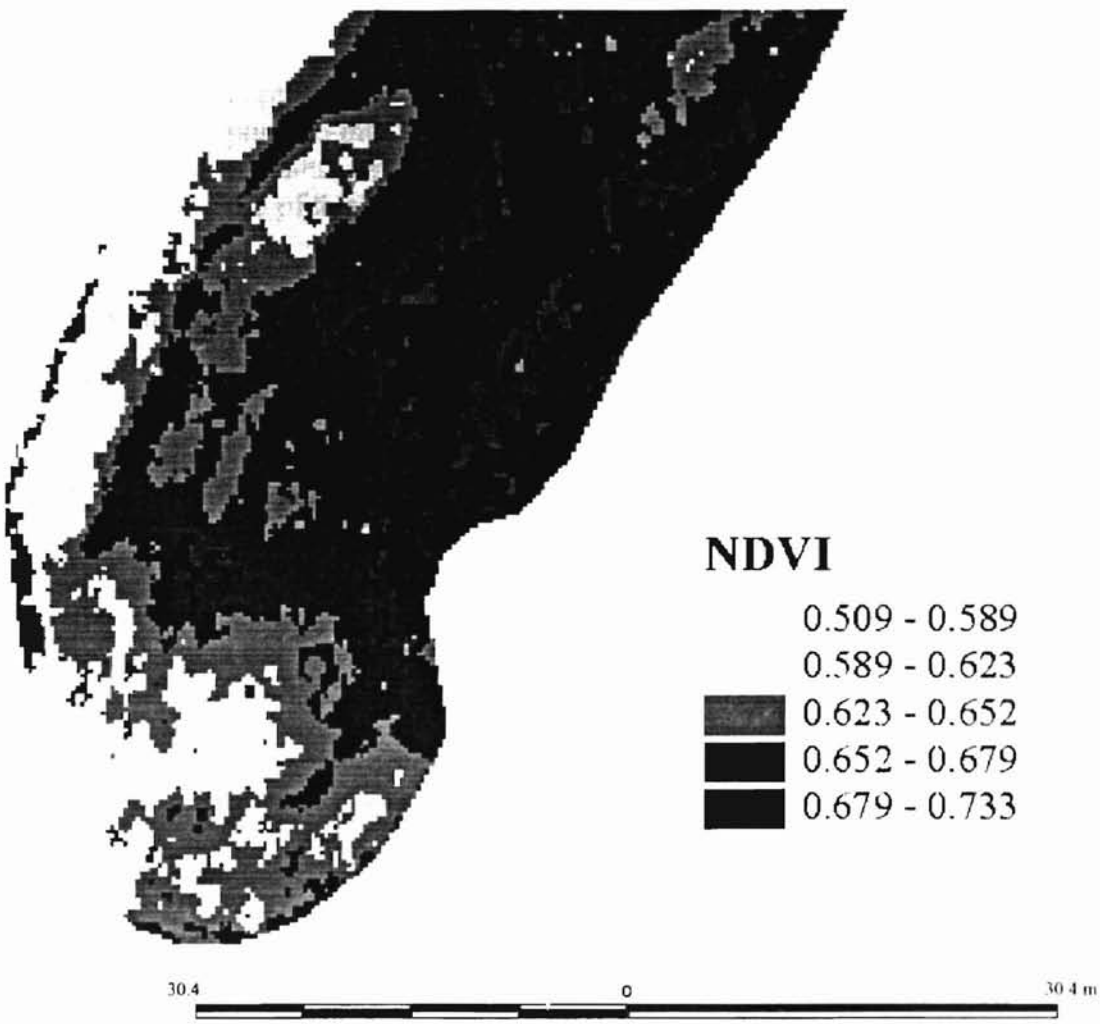

Figure B12: Karsten Creek Fairway 12, South Section, Stillwater, OK, June 13, 2000

Appendix C

Procedure for Taking and Processing Data

Data Acquisition -- C1

Make certain correct cable is plugged into the output from the Patchen sensors. The output normally is plugged into the gray sensor box on the front of the 4100. In order to scan with GPS coordinates, it must be attached to the cable that runs to the rear of the tractor. The other end of this cable must be attached to the end coming off the DAQBOOK.

The cables on the DAQBOOK should be hooked up as follows:

Channel	Wire	Description
37	1r*	Red wire from Patchen sensor 1
36	2r	Red wire from Patchen sensor 2
35	3r	Red wire from Patchen sensor 3
34	4r	Red wire from Patchen sensor 4
19-18	black	Wires should connect these channels so that all Patchen signals are properly grounded
18-17		
17-16		
16-15		
18	1b**	Black wire from Patchen sensor 1
17	2b	Black wire from Patchen sensor 2
16	3b	Black wire from Patchen sensor 3
15	4b	Black wire from Patchen sensor 4
1		Red wire from shaft encoder-reducer
25		White wire from shaft encoder-reducer
7		Black and shielding wires from shaft encoder-reducer

* red wire

** black and shielding wires combined, or just black wire

Make sure the shaft encoder cable is plugged into the reducer box

Parallel cable should be hooked to the port that says "To computer" and the other end hooked to the laptop parallel port

Ribbon cable must be properly hooked up between the DAQBOOK/100 and the case containing the DBK11

Serial cable from GPS should be hooked to the serial port of the laptop

After making certain all wires and cables are properly hooked up, turn on power to the sensors, GPS, DAQBOOK and boot the laptop. Let the Patchen sensors warm up at least a couple of minutes (the warm up light should begin flashing, and an annoying beeping will begin to sound)

Open the program "tracker.exe" under the directory c:\chi\nsmntr (or whatever drive it's installed on

Open the GPS menu and choose "Settings"

Change the baud rate to match the GPS (for the 132, it is adjustable, currently set at 9600). Click OK. The "light" for the GPS signal should appear. If not, check cables, power, and settings.

Open the Acquire menu, and choose "Options"

Change the trigger to TTL if you want to use the shaft encoder to trigger a response. If not, use the internal clock and set your desired frequency. Click OK.

Open the Acquire menu again. Choose "Begin".

The Acquire "light" should be on. If not, check power to DAQBOOK, cable connections, and settings. If you are using TTL, it may require some forward travel to begin acquiring data.

To begin logging data, move to the position where you wish to begin, and open the File menu. Choose "Open". A dialogue box will appear, prompting you to choose a file name to save the data. After you choose the file name and click Open, or hit **Enter**, the "light" for the File should be on.

After you choose the file, it will begin logging data to that file. When you are done, Choose File, "Close". The file will be saved in the directory you chose.

Note: There are also buttons for Begin, Stop, File Open , and File Close
The voltage values that appear on the screen should be somewhere around +1.8 to +2.8 for grass, wheat, etc. If negative voltages appear, there is a problem. First look to make sure all connections are properly done, and check to make sure all needed units are powered and turned on.

Make sure to shut off all power switches after you are finished.

File Analysis -- C2

Determine the air temperature (°F) from the time you took your readings.

Run "4100 for grass 1.exe" to convert "funky" lat/lon to decimal degrees and convert voltages to NDVI

Click the Input Temperature button, and type in the temperature (°F) from when you took your readings.

If you have previously run the program and the calibration coefficients are all correct, you can click Retrieve Last Calibration Curves, if not, click Input Sensor Calibration Curves.

Currently the curves are as follows:

Sensor 1 Slope	1.636866
Sensor 1 3rd Order Temp	0.0021
Sensor 1 2nd Order Temp	-0.1254
Sensor 1 1st Order Temp	-37.888
Sensor 1 Temp Intercept	15652
Sensor 1 Intercept	-1.313
Sensor 2 Slope	1.854818
Sensor 2 3rd Order Temp	-0.015
Sensor 2 2nd Order Temp	3.7638
Sensor 2 1st Order Temp	-317.63
Sensor 2 Temp Intercept	22925
Sensor 2 Intercept	-1.4299
Sensor 3 Slope	1.59175
Sensor 3 3rd Order Temp	-0.0103
Sensor 3 2nd Order Temp	2.6614
Sensor 3 1st Order Temp	-235.46
Sensor 3 Temp Intercept	19768
Sensor 3 Intercept	-1.2584
Sensor 4 Slope	1.764511
Sensor 4 3rd Order Temp	0.0016
Sensor 4 2nd Order Temp	0.0189
Sensor 4 1st Order Temp	-46.108
Sensor 4 Temp Intercept	15944
Sensor 4 Intercept	-1.4063

Choose File, Open

A dialogue box to choose the input file should appear. The file you want is "yourfile.dat", which is the raw data from HongYu Chi's program. Currently this data is in the following format:

xxxxDD.mmMMMMMMxxSDDD.mmMMMMMMxxVVVVVVxxxV.VVVxxxV.VVVxxxV.VVVxxxV.VVV

where "x" is a space, the "DD" is degrees latitude, the first "mm" is minutes latitude, the first "MMMMMM" is the decimal minutes latitude, the "S" is the sign for longitude, the "DDD" is degrees longitude, the second "mm" is minutes longitude, the second "MMMMMM" is decimal minutes longitude, the "VVVVVV" is a system voltage check, the "V.VVV"s are the raw voltages for Patchen sensor 1, 2, 3, and 4, respectively, if the Patchen sensors are hooked into the DAQBOOK correctly.

A second dialogue box will appear after you choose your input file. You will need to type in a file name. If you just hit **Enter**, it will put it in the same directory as your input file. I would suggest that you put most of these files in a temporary directory, and only save the original *.dat file and the final *.csv file in a permanent directory. This program will save a tab delimited file with Latitude (decimal degrees), Longitude (decimal degrees), NDVI1, NDVI2, NDVI3, and NDVI4, as a file called "yourfile.pts". If you just hit **Enter**, and do not pay attention to what is in the dialogue box, your file will be named "yourfile.dat.pts".

Choose File, Exit

Open BlacklandGrass, a GIS program from Texas A&M Blackland Research Center, Temple TX..

Choose Options, Start Shell Window.

Type "mll2u" and hit **Enter**. This will convert lat/lon to UTM. This will not affect the NDVI values, just the first two columns of numbers.

When asked for the spheroid, type "wgs84" and hit **Enter**.
Hit **Enter** again to confirm.

When asked for the zone, type "14" and hit **Enter**, or choose the correct UTM zone if it is not 14.
Hit **Enter** again to confirm.

When asked for the path, type the full path of your file (e.g. "d:\temp\yourfile.pts") and hit **Enter**.
Hit **Enter** again to confirm.

When asked for the path of the output file, type the path for your new file (e.g. "d:\temp\yourfile.out) and hit **Enter**. Make certain to name the file *.out, or it will not appear in the dialogue box for the next part of the program. You can call it whatever you want, but you will have to type in the name of the file next time, if it is not a *.out file.
Hit **Enter** again to confirm.

When asked if the input is reversed, type "y" for yes and hit **Enter**.

When asked if you want to flag invalid UTM points, hit **Enter** to accept no.

When asked if you want to flag other errors, hit **Enter** to accept no.

When asked if you want to suppress printing the UTM zone, **TYPE "Y" FOR YES!!!!!!!!!!! and hit *Enter*.**

Let the program run, and when the mll2u window disappears, it is done.

Run "4100 for grass 2.exe" to extract geographic position for each individual sensor in UTM coordinates

Click the "Input Sensor Offsets" button and type the distance (feet) from the GPS antenna to the sensors. ***THIS IS NOT ACTUALLY THE DISTANCE FROM THE ANTENNA TO THE SENSORS!!! Due to the delays in the signal travelling from the satellites to the GPS, there is a slight offset.*** Right now the correct distance to type in is 11.5 ft.

Select File, Open. A dialogue box should appear, asking for an input file (*.out). Choose "yourfile.out" from the available files, or type in the path and name of your file. Hit **Enter**.

Another dialogue box should appear, and you will need to input your output file name ("yourfile.pt2") and hit **Enter**. This will output a file with the following: UTMy (meters), UTMx (meters), NDVI, Sensor #.

From BlacklandGrass shell window, type "mu2ll". This will convert back from UTM to lat/lon (in decimal minutes if you so specify).

From each prompt, type the following:

Spheroid: wgs84, **Enter**, **Enter**
UTM zone: 14, **Enter**, **Enter**
Input: "path\yourfile.pt2", **Enter**, **Enter**
Output: "path\yourfile.ot2", **Enter**, **Enter**
Southern Hemisphere: **Enter** (for no)
Points reversed: YES, **Enter**

Flag input errors: **Enter** (for no)
Flag other errors: **Enter** (for no)
Decimal degrees: YES, **Enter**

When mu2ll is done running, you can exit from the shell window and close BlacklandGrass.

Run "4100 for grass 3.exe" to convert the "*.ot2" file to a comma delimited file "*.csv" that can be imported to SSToolbox

File, Open

Choose "yourfile.ot2" or type the path and filename for your file.

For output, choose the directory where you want to store the output (I would suggest that you put this file somewhere a little more permanent than where you've been storing these other files. Type your file name, and it will append a csv to the name. ("yourfile.csv")

You can now import the comma delimited (csv) file into SSToolbox. It will appear with the following:

Lat Lon NDVI Sensor#

To generate a grid file for use in mapping and semivariance analysis

Open SSToolbox, and create a field as usual (this is not a Toolbox training manual, so you'll have to learn that somewhere else).

Import the data points as a comma delimited text file (*.csv) into that field.

Use the table editing feature to delete any outliers (negative or very large NDVI, or any points with a Sensor# of "0").

Choose Input/Convert, Create Surface Using Interpolation

Change the Cell Size to 1 ft

Choose the correct variable (in this case, NDVI)

Change the Interpolation Method to "Inverse Distance"

Click the "Advanced" button

Make certain the following parameters are correct:

Data Treatment: Duplicates = Average

Ignore Data Outside Grid = check marked on

Search Option = Simple

Search Ellipse: Radius 1 = 0.5
Radius 2 = 0.5

Click OK for the Advanced options

Click OK for Creating the surface

Toolbox will "shell" to Surfer and create the grid, then import it into Toolbox as a surface

Export the surface as a shapefile and import the "*.dbf" file into Excel. It will have the following format:

ID Longitude Latitude NDVI (or other variable)

To put this file into a form that can be easily used to calculate semivariance, run the Excel VBA macro "matrixconv.bas". This will convert the columnar data into a matrix.

After conversion, place the cursor on the cell B2 (the first cell of the data [i.e. not a position coordinate, but an NDVI value]) and run the Excel VBA macro "semivariance.bas". This will place two columns of data to the right of the matrix. The first column is Separation Index (to get Separation Distance, multiply the Separation Index times the incremental separation distance). The second column is Semivariance.

This will give you the semivariance for the east-west direction. To get semivariance in the north-south direction, select the entire matrix (including lon/lat data) and copy to the clipboard. Go to a new sheet, place the cursor in cell A1 and Paste Special: Transpose. Then run the "semivariance.bas" macro just as you did for east-west.

To fit a curve to the semivariance data, graph the data, and visually identify the sills. Extract the data that makes up the initial part of the curve, and use Table Curve to fit a "spherical" curve to the data. The results will be a curve with three constants, "a," "b," and "c." The maximum occurs at the X-value "a + b". This means that if you add the values of "a" and "b" you will be computing the range. The value of Y at the range (a+b) from Eq. 3 will be the sill.

The integral scale is computed as described in Eq. 8

Appendix D

Visual Basic Programs for Post Processing of Data

4100 for grass 1.exe

"4100 for grass 1.exe" converts the GPS input from the raw data file to decimal degrees latitude and longitude and changes sensor voltages to NDVI.

```
Private Sub Command3_Click()

  Open "d:\temp\coeff.txt" For Input As #4

    Input #4, Slope1, T1Coeff3, T1Coeff2, T1Coeff1, T1Inter, Inter1
    Input #4, Slope2, T2Coeff3, T2Coeff2, T2Coeff1, T2Inter, Inter2
    Input #4, Slope3, T3Coeff3, T3Coeff2, T3Coeff1, T3Inter, Inter3
    Input #4, Slope4, T4Coeff3, T4Coeff2, T4Coeff1, T4Inter, Inter4

    Label1.Caption = Slope1
    Label3.Caption = T1Coeff3
    Label4.Caption = T1Coeff2
    Label5.Caption = T1Coeff1
    Label6.Caption = T1Inter
    Label7.Caption = Inter1
    Label8.Caption = Slope2
    Label10.Caption = T2Coeff3
    Label11.Caption = T2Coeff2
    Label12.Caption = T2Coeff1
    Label13.Caption = T2Inter
    Label14.Caption = Inter2
    Label15.Caption = Slope3
    Label17.Caption = T3Coeff3
    Label18.Caption = T3Coeff2
    Label19.Caption = T3Coeff1
    Label20.Caption = T3Inter
    Label21.Caption = Inter3
    Label22.Caption = Slope4
    Label24.Caption = T4Coeff3
    Label25.Caption = T4Coeff2
    Label26.Caption = T4Coeff1
    Label27.Caption = T4Inter
```

```
      Label28.Caption = Inter4


    Close #4

End Sub
Private Sub Command1_Click()          'input temperature
    Dim prompt
    prompt = "Type the air temperature during sensor readings."
    Temp = Val(InputBox$(prompt))
    Label29.Caption = Temp


End Sub

Private Sub Command2_Click()          'input calibration constants
    Dim prompt
    prompt = "Input Sensor 1 Slope."
    Slope1 = Val(InputBox$(prompt))
    Label1.Caption = Slope1
    prompt = "Input Sensor 1 3rd order Temperature Coefficient."
    T1Coeff3 = Val(InputBox$(prompt))
    Label3.Caption = T1Coeff3
    prompt = "Input Sensor 1 2nd order Temperature Coefficient."
    T1Coeff2 = Val(InputBox$(prompt))
    Label4.Caption = T1Coeff2
    prompt = "Input Sensor 1 1st order Temperature Coefficient."
    T1Coeff1 = Val(InputBox$(prompt))
    Label5.Caption = T1Coeff1
    prompt = "Input Sensor 1 Temperature Intercept."
    T1Inter = Val(InputBox$(prompt))
    Label6.Caption = T1Inter
    prompt = "Input Sensor 1 Intercept."
    Inter1 = Val(InputBox$(prompt))
    Label7.Caption = Inter1

    prompt = "Input Sensor 2 Slope."
    Slope2 = Val(InputBox$(prompt))
    Label8.Caption = Slope2
    prompt = "Input Sensor 2 3rd order Temperature Coefficient."
    T2Coeff3 = Val(InputBox$(prompt))
    Label10.Caption = T2Coeff3
    prompt = "Input Sensor 2 2nd order Temperature Coefficient."
    T2Coeff2 = Val(InputBox$(prompt))
    Label11.Caption = T2Coeff2
    prompt = "Input Sensor 2 1st order Temperature Coefficient."
```

```
T2Coeff1 = Val(InputBox$(prompt))
Label12.Caption = T2Coeff1
prompt = "Input Sensor 2 Temperature Intercept."
T2Inter = Val(InputBox$(prompt))
Label13.Caption = T2Inter
prompt = "Input Sensor 2 Intercept."
Inter2 = Val(InputBox$(prompt))
Label14.Caption = Inter2

prompt = "Input Sensor 3 Slope."
Slope3 = Val(InputBox$(prompt))
Label15.Caption = Slope3
prompt = "Input Sensor 3 3rd order Temperature Coefficient."
T3Coeff3 = Val(InputBox$(prompt))
Label17.Caption = T3Coeff3
prompt = "Input Sensor 3 2nd order Temperature Coefficient."
T3Coeff2 = Val(InputBox$(prompt))
Label18.Caption = T3Coeff2
prompt = "Input Sensor 3 1st order Temperature Coefficient."
T3Coeff1 = Val(InputBox$(prompt))
Label19.Caption = T3Coeff1
prompt = "Input Sensor 3 Temperature Intercept."
T3Inter = Val(InputBox$(prompt))
Label20.Caption = T3Inter
prompt = "Input Sensor 3 Intercept."
Inter3 = Val(InputBox$(prompt))
Label21.Caption = Inter3

prompt = "Input Sensor 4 Slope."
Slope4 = Val(InputBox$(prompt))
Label22.Caption = Slope4
prompt = "Input Sensor 4 3rd order Temperature Coefficient."
T4Coeff3 = Val(InputBox$(prompt))
Label24.Caption = T4Coeff3
prompt = "Input Sensor 4 2nd order Temperature Coefficient."
T4Coeff2 = Val(InputBox$(prompt))
Label25.Caption = T4Coeff2
prompt = "Input Sensor 4 1st order Temperature Coefficient."
T4Coeff1 = Val(InputBox$(prompt))
Label26.Caption = T4Coeff1
prompt = "Input Sensor 4 Temperature Intercept."
T4Inter = Val(InputBox$(prompt))
Label27.Caption = T4Inter
prompt = "Input Sensor 4 Intercept."
Inter4 = Val(InputBox$(prompt))
Label28.Caption = Inter4
```

```
   Open "d:\temp\coeff.txt" For Output As #3
   Write #3, Slope1, T1Coeff3, T1Coeff2, T1Coeff1, T1Inter, Inter1
   Write #3, Slope2, T2Coeff3, T2Coeff2, T2Coeff1, T2Inter, Inter2
   Write #3, Slope3, T3Coeff3, T3Coeff2, T3Coeff1, T3Inter, Inter3
   Write #3, Slope4, T4Coeff3, T4Coeff2, T4Coeff1, T4Inter, Inter4

   Close #3

End Sub


Private Sub mnuItemExit_Click()
   End                    'quit program
End Sub

Private Sub mnuItemOpen_Click()  'specify input file

   Wrap$ = Chr$(13) + Chr$(10)  'create wrap character
   CommonDialog1.Filter = "Data files (*.DAT)|*.DAT"
   CommonDialog1.ShowOpen       'display Open dialog box
   If CommonDialog1.FileName <> "" Then
      Form1.MousePointer = 11  'display hourglass
      Open CommonDialog1.FileName For Input As #1


   CommonDialog1.Filter = "Output files (*.pts)|*.pts"
   CommonDialog1.DialogTitle = "Specify Output File"
   CommonDialog1.ShowOpen      'specify output file

   If CommonDialog1.FileName <> "" Then
      Form1.MousePointer = 11
      Open CommonDialog1.FileName For Output As #2

      Dim Latdeg As Double        'Latitude         'dimension arrays as variable size
      Dim Latmin As Double         'Longitude
      Dim Latdecm As Double
      Dim Londeg As Double
      Dim Lonmin As Double
      Dim Londecm As Double
      Dim NDVI1 As Double        'NDVI for sensor #1
      Dim NDVI2 As Double        'NDVI for sensor #2
      Dim NDVI3 As Double        'NDVI for sensor #3
      Dim NDVI4 As Double        'NDVI for sensor #4
      Dim v1 As Double
      Dim v2 As Double
```

```
    Dim v3 As Double
    Dim v4 As Double

    Do Until EOF(1)         'then read lines from file and assign the correct values to Lat.
Lon, and Voltages 1-4
        Line Input #1, lineoftext$
        a = a + 1
        Latdeg = Val(Mid(lineoftext, 5, 2))
        Latmin = Val(Mid(lineoftext, 8, 2))
        Latdecm = Val("." + Mid(lineoftext, 10, 6))
        Londeg = Val(Mid(lineoftext, 18, 4))
        Lonmin = Val(Mid(lineoftext, 23, 2))
        Londecm = Val("." + Mid(lineoftext, 25, 6))
        Lat = Latdeg + (Latmin + Latdecm) / 60
        Lon = Londeg - (Lonmin + Londecm) / 60
        v1 = Val(Mid(lineoftext, 42, 5))
        v2 = Val(Mid(lineoftext, 50, 5))
        v3 = Val(Mid(lineoftext, 58, 5))
        v4 = Val(Mid(lineoftext, 66, 5))

        NDVI1 = Slope1 * v1 * 6240# / (T1Coeff3 * Temp ^ 3# + T1Coeff2 * Temp ^ 2#
+ T1Coeff1 * Temp + T1Inter) + Inter1 ' calculate NDVI values and write to array
locations
        NDVI2 = Slope2 * v2 * 6240# / (T2Coeff3 * Temp ^ 3# + T2Coeff2 * Temp ^ 2#
+ T2Coeff1 * Temp + T2Inter) + Inter2
        NDVI3 = Slope3 * v3 * 6240# / (T3Coeff3 * Temp ^ 3# + T3Coeff2 * Temp ^ 2#
+ T3Coeff1 * Temp + T3Inter) + Inter3
        NDVI4 = Slope4 * v4 * 6240# / (T4Coeff3 * Temp ^ 3# + T4Coeff2 * Temp ^ 2#
+ T4Coeff1 * Temp + T4Inter) + Inter4


        Print #2, Format(Lat, "00.0000000000"); Chr(9); Format(Lon,
"000.0000000000"); Chr(9); Format(NDVI1, "0.00000"); Chr(9); Format(NDVI2,
"0.00000"); Chr(9); Format(NDVI3, "0.00000"); Chr(9); Format(NDVI4, "0.00000")


    Loop

CleanUp:
    Form1.MousePointer = 0   'reset mouse
    Close #1                 'close file
    Close #2
    Close #10

  End If
  End If
```

```
   Exit Sub
End Sub
```

Below this line is the module for declaring public variables for 4100 for grass 1

```
Public Temp As Double
Public Slope1 As Double
Public Slope2 As Double
Public Slope3 As Double
Public Slope4 As Double
Public Inter1 As Double
Public Inter2 As Double
Public Inter3 As Double
Public Inter4 As Double
Public T1Slope As Double
Public T2Slope As Double
Public T3Slope As Double
Public T4Slope As Double
Public T1Inter As Double
Public T1Coeff3 As Double
Public T1Coeff2 As Double
Public T1Coeff1 As Double
Public T2Inter As Double
Public T2Coeff3 As Double
Public T2Coeff2 As Double
Public T2Coeff1 As Double
Public T3Inter As Double
Public T3Coeff3 As Double
Public T3Coeff2 As Double
Public T3Coeff1 As Double
Public T4Inter As Double
Public T4Coeff3 As Double
Public T4Coeff2 As Double
Public T4Coeff1 As Double
Public length As Double
Public slopex As Double
Public intx As Double
Public slopey As Double
Public inty As Double
```

4100 for grass 2.exe

"4100 for grass 2.exe" interpolates between known data points and assigns a UTM coordinate to each sensor reading.

```
Private Sub Command4_Click()

  Dim prompt

  prompt = "Input Offset Length (feet) from GPS antenna to sensor bar (along vehicle
long axis)."
  length = Val(InputBox$(prompt))

End Sub

Private Sub mnuItemExit_Click()
  End                    'quit program
End Sub

Private Sub mnuItemOpen_Click()  'specify input file

  Wrap$ = Chr$(13) + Chr$(10)  'create wrap character
  CommonDialog1.Filter = "Data files (*.out)|*.out"
  CommonDialog1.ShowOpen      'display Open dialog box
  If CommonDialog1.FileName <> "" Then
    Form1.MousePointer = 11  'display hourglass
    Open CommonDialog1.FileName For Input As #1


  CommonDialog1.Filter = "Output files (*.pt2)|*.pt2"
  CommonDialog1.DialogTitle = "Specify Output File"
  CommonDialog1.ShowOpen      'specify output file

  If CommonDialog1.FileName <> "" Then
    Form1.MousePointer = 11
    Open CommonDialog1.FileName For Output As #2

    Print #2, "Lat"; Chr(9); "Lon"; Chr(9); "NDVI"; Chr(9); "Sensor#"

    Dim UTMyM() As Double      'Latitude       'dimension arrays as variable size
    Dim UTMxM() As Double      'Longitude
```

```
    Dim NDVI1()       'NDVI for sensor #1
    Dim NDVI2()       'NDVI for sensor #2
    Dim NDVI3()       'NDVI for sensor #3
    Dim NDVI4()       'NDVI for sensor #4

    ReDim Preserve UTMyM(100)       'set initial dimensions of arrays to 100
    ReDim Preserve UTMxM(100)
    ReDim Preserve NDVI1(100)
    ReDim Preserve NDVI2(100)
    ReDim Preserve NDVI3(100)
    ReDim Preserve NDVI4(100)

    Dim strval As String

    Do Until EOF(1)         'then read lines from file and assign the correct values to Lat,
Lon, and Voltages 1-4
      Line Input #1, lineoftext$
      a = a + 1

      If Mid(lineoftext, 1, 1) = "L" Then
      GoTo 10
      End If

      b = 0
      b = b + 1
      Do Until Mid(lineoftext, b, 1) = Chr(9)
        strval = strval & Mid(lineoftext, b, 1)
        b = b + 1
      Loop

      UTMyM(a - 1) = Val(strval)
      b = b + 1
      strval = ""

      Do Until Mid(lineoftext, b, 1) = Chr(9)
        strval = strval & Mid(lineoftext, b, 1)
        b = b + 1
      Loop

      UTMxM(a - 1) = Val(strval)
      b = b + 1
      strval = ""

      NDVI1(a - 1) = Val(Mid(lineoftext, b, 7))
        b = b + 8
```

```
        NDVI2(a - 1) = Val(Mid(lineoftext, b, 7))
          b = b + 8

        NDVI3(a - 1) = Val(Mid(lineoftext, b, 7))
          b = b + 8

        NDVI4(a - 1) = Val(Mid(lineoftext, b, 7))
          b = b + 8

        counts = counts + 1

        If counts = 100 Then              'enlarge arrays if needed
          ReDim Preserve UTMyM(a + 100)
          ReDim Preserve UTMxM(a + 100)
          ReDim Preserve NDVI1(a + 100)
          ReDim Preserve NDVI2(a + 100)
          ReDim Preserve NDVI3(a + 100)
          ReDim Preserve NDVI4(a + 100)
          counts = 0
        End If

10      Loop


      'this part of the program will interpolate GPS points for each repeated lat and lon by
fitting a straight line between points

      Dim fcount As Double
      Dim points(2) As Double


      b = 1
      numpoints1 = b
      Py1 = UTMyM(b)
      Px1 = UTMxM(b)

      Do Until b = a

        If UTMyM(b) <> UTMyM(b + 1) Or UTMxM(b) <> UTMxM(b + 1) Then 'if the
GPS coordinate changes, then consider that a point for interpolation

          If UTMyM(b + 1) = UTMyM(b + 2) And UTMxM(b + 2) = UTMxM(b + 2)
Then

            py2 = UTMyM(b + 1)
            Px2 = UTMxM(b + 1)
```

```
            If Px2 = 0 Or py2 = 0 Then
               GoTo 13
            End If
            numpoints2 = b + 1

            Do Until c = numpoints2 - numpoints1
               c = c + 1
               UTMyM(numpoints1 + c) = c * (py2 - Py1) / (numpoints2 - numpoints1)
+ Py1
               UTMxM(numpoints1 + c) = c * (Px2 - Px1) / (numpoints2 - numpoints1)
+ Px1

            Loop
            Py1 = py2
            Px1 = Px2
            numpoints1 = numpoints2
            c = 0
         End If

      End If
      b = b + 1
   Loop


   'this part of the program assigns a latitude and longitude to each sensor at each point.
   'and associates the correct NDVI reading with each sensor
13

   Dim UTMy() As Double
   Dim UTMx() As Double
   Dim ys1() As Double
   Dim ys2() As Double
   Dim ys3() As Double
   Dim ys4() As Double
   Dim xs1() As Double
   Dim xs2() As Double
   Dim xs3() As Double
   Dim xs4() As Double

   ReDim UTMy(a)
   ReDim UTMx(a)
   ReDim ys1(a)
   ReDim ys2(a)
   ReDim ys3(a)
   ReDim ys4(a)
   ReDim xs1(a)
```

```
ReDim xs2(a)
ReDim xs3(a)
ReDim xs4(a)


Const sensor1 As Double = 1.5
Const sensor2 As Double = 0.5
Const sensor4 As Double = -1.5
Const sensor3 As Double = -0.5
Const numpoints As Double = 5
Const pi As Double = 3.14159
Const piover2 As Double = pi / 2
Const threepiover2 As Double = 3 * pi / 2
Const twopi As Double = 2 * pi
Dim theta1 As Double
Dim theta2 As Double
Dim theta3 As Double
Dim theta4 As Double
Dim l1 As Double
Dim l2 As Double
Dim l3 As Double
Dim l4 As Double
Dim phi() As Double
ReDim phi(a)
Dim phi1 As Double
Dim phi2 As Double
Dim phi3 As Double
Dim phi4 As Double
Dim ysens1 As Double
Dim ysens2 As Double
Dim ysens3 As Double
Dim ysens4 As Double
Dim xsens1 As Double
Dim xsens2 As Double
Dim xsens3 As Double
Dim xsens4 As Double
Dim deltax() As Double
Dim deltay() As Double
ReDim deltax(a)
ReDim deltay(a)
Dim slopeone() As Double
ReDim slopeone(a)
Dim slope As Double
Dim newUTMx() As Double
ReDim newUTMx(a)
Dim newUTMy() As Double
```

```
    ReDim newUTMy(a)

    theta1 = Atn(sensor1 / length)
    theta2 = Atn(sensor2 / length)          ' tractor and the line between the GPS
    theta3 = Atn(sensor3 / length)          ' antenna and the sensor is theta
    theta4 = Atn(sensor4 / length)
    l1 = Sqr(sensor1 ^ 2 + length ^ 2)
    l2 = Sqr(sensor2 ^ 2 + length ^ 2)
    l3 = Sqr(sensor3 ^ 2 + length ^ 2)
    l4 = Sqr(sensor4 ^ 2 + length ^ 2)


    h = 0

    Do Until h = b

      h = h + 1


      UTMy(h) = UTMyM(h) * 3.28083989501  'transform UTM(meters) into
UTM(feet)
      UTMx(h) = UTMxM(h) * 3.28083989501
    Loop




    j = 50

    Do Until j = b - 50

      j = j + 1
      newUTMy(j) = (UTMy(j - 50) + UTMy(j - 49) + UTMy(j - 48) + UTMy(j - 47) +
UTMy(j - 46) + UTMy(j - 45) _
      + UTMy(j - 44) + UTMy(j - 43) + UTMy(j - 42) + UTMy(j - 41) + UTMy(j - 40)
+ UTMy(j - 39) + UTMy(j - 38) _
      + UTMy(j - 37) + UTMy(j - 36) + UTMy(j - 35) + UTMy(j - 34) + UTMy(j - 33)
+ UTMy(j - 32) + _
      UTMy(j - 31) + UTMy(j - 30) + UTMy(j - 29) + UTMy(j - 28) + UTMy(j - 27) +
UTMy(j - 26) + _
      UTMy(j - 25) + UTMy(j - 24) + UTMy(j - 23) + UTMy(j - 22) + UTMy(j - 21) +
UTMy(j - 20) + _
      UTMy(j - 19) + UTMy(j - 18) + UTMy(j - 17) + UTMy(j - 16) + UTMy(j - 15) +
UTMy(j - 14) _
      + UTMy(j - 13) + UTMy(j - 12) + UTMy(j - 11) + UTMy(j - 10) + UTMy(j - 9) +
UTMy(j - 8) + UTMy(j - 7) _
```

```
        + UTMy(j - 6) + UTMy(j - 5) + UTMy(j - 4) + UTMy(j - 3) + UTMy(j - 2) +
UTMy(j - 1) + UTMy(j) + UTMy(j + 10) _
        + UTMy(j + 9) + UTMy(j + 8) + UTMy(j + 7) + UTMy(j + 6) + UTMy(j + 5) +
UTMy(j + 4) + UTMy(j + 3) _
        + UTMy(j + 2) + UTMy(j + 1) + UTMy(j + 11) + UTMy(j + 12) + UTMy(j + 13)
+ UTMy(j + 14) + UTMy(j + 15) _
        + UTMy(j + 16) + UTMy(j + 17) + UTMy(j + 18) + UTMy(j + 19) + UTMy(j +
20) + UTMy(j + 21) + UTMy(j + 22) _
        + UTMy(j + 23) + UTMy(j + 24) + UTMy(j + 25) + UTMy(j + 26) + UTMy(j +
27) + UTMy(j + 28) + UTMy(j + 29) _
        + UTMy(j + 30) + UTMy(j + 31) + UTMy(j + 32) + UTMy(j + 33) + UTMy(j +
34) + UTMy(j + 35) + UTMy(j + 36) _
        + UTMy(j + 37) + UTMy(j + 38) + UTMy(j + 39) + UTMy(j + 40) + UTMy(j +
41) + UTMy(j + 42) + UTMy(j + 43) _
        + UTMy(j + 44) + UTMy(j + 45) + UTMy(j + 46) + UTMy(j + 47) + UTMy(j +
48) + UTMy(j + 49) + UTMy(j + 50)) / 101#

        newUTMx(j) = (UTMx(j - 50) + UTMx(j - 49) + UTMx(j - 48) + UTMx(j - 47) +
UTMx(j - 46) + UTMx(j - 45) _
        + UTMx(j - 44) + UTMx(j - 43) + UTMx(j - 42) + UTMx(j - 41) + UTMx(j - 40)
+ UTMx(j - 39) + UTMx(j - 38) _
        + UTMx(j - 37) + UTMx(j - 36) + UTMx(j - 35) + UTMx(j - 34) + UTMx(j - 33)
+ UTMx(j - 32) + _
        UTMx(j - 31) + UTMx(j - 30) + UTMx(j - 29) + UTMx(j - 28) + UTMx(j - 27) +
UTMx(j - 26) + _
        UTMx(j - 25) + UTMx(j - 24) + UTMx(j - 23) + UTMx(j - 22) + UTMx(j - 21) +
UTMx(j - 20) + _
        UTMx(j - 19) + UTMx(j - 18) + UTMx(j - 17) + UTMx(j - 16) + UTMx(j - 15) +
UTMx(j - 14) _
        + UTMx(j - 13) + UTMx(j - 12) + UTMx(j - 11) + UTMx(j - 10) + UTMx(j - 9) +
UTMx(j - 8) + UTMx(j - 7) _
        + UTMx(j - 6) + UTMx(j - 5) + UTMx(j - 4) + UTMx(j - 3) + UTMx(j - 2) +
UTMx(j - 1) + UTMx(j) + UTMx(j + 10) _
        + UTMx(j + 9) + UTMx(j + 8) + UTMx(j + 7) + UTMx(j + 6) + UTMx(j + 5) +
UTMx(j + 4) + UTMx(j + 3) _
        + UTMx(j + 2) + UTMx(j + 1) + UTMx(j + 11) + UTMx(j + 12) + UTMx(j + 13)
+ UTMx(j + 14) + UTMx(j + 15) _
        + UTMx(j + 16) + UTMx(j + 17) + UTMx(j + 18) + UTMx(j + 19) + UTMx(j +
20) + UTMx(j + 21) + UTMx(j + 22) _
        + UTMx(j + 23) + UTMx(j + 24) + UTMx(j + 25) + UTMx(j + 26) + UTMx(j +
27) + UTMx(j + 28) + UTMx(j + 29) _
        + UTMx(j + 30) + UTMx(j + 31) + UTMx(j + 32) + UTMx(j + 33) + UTMx(j +
34) + UTMx(j + 35) + UTMx(j + 36) _
        + UTMx(j + 37) + UTMx(j + 38) + UTMx(j + 39) + UTMx(j + 40) + UTMx(j +
41) + UTMx(j + 42) + UTMx(j + 43) _
```

```
    + UTMx(j + 44) + UTMx(j + 45) + UTMx(j + 46) + UTMx(j + 47) + UTMx(j +
48) + UTMx(j + 49) + UTMx(j + 50)) / 101#


   Loop

   k = 52
   Do Until k = b - 52
     k = k + 1

     deltax(k) = newUTMx(k + 1) - newUTMx(k)
     deltay(k) = newUTMy(k + 1) - newUTMy(k)
     If deltax(k) = 0 Then
        If deltay(k) > 0 Then
           slopeone(k) = 1000000000000#
        ElseIf deltay(k) < 0 Then
           slopeone(k) = -1000000000000#
        End If

     Else: slopeone(k) = deltay(k) / deltax(k)
     End If

  Loop

     i = 52
     Do Until i = b - 52
        i = i + 1
        slope = slopeone(i)
        If deltay(i) > 0 And deltax(i) > 0 Then
           phi(i) = Atn(slope)
        ElseIf deltay(i) > 0 And deltax(i) < 0 Then
           phi(i) = pi - Atn(-slope)
        ElseIf deltay(i) < 0 And deltax(i) < 0 Then
           phi(i) = pi + Atn(slope)
        ElseIf deltay(i) < 0 And deltax(i) > 0 Then
           phi(i) = -Atn(-slope) + twopi
        ElseIf deltay(i) = 0 Then
           If deltax(i) > 0 Then
              phi(i) = Atn(slope)
           ElseIf deltax(i) < 0 Then
              phi(i) = -Atn(slope)
           End If
        End If

        If deltax(i) = 0 Then
           If deltay(i) > 0 Then
```

```
            phi(i) = pi / 2#
         ElseIf deltay(i) < 0 Then
            phi(i) = 3# * pi / 2#
         End If

      End If


2000        phi1 = phi(i) + theta1                    'phi1 is the angle of orientation
of line between GPS antenna and sensor1
      phi2 = phi(i) + theta2                          'phi2 between GPS and sensor2,
etc.
      phi3 = phi(i) + theta3
      phi4 = phi(i) + theta4

      ysens1 = (newUTMy(i) + l1 * Sin(phi1)) / 3.28083989501        'add/subtract
the necessary number of feet
      xsens1 = (newUTMx(i) + l1 * Cos(phi1)) / 3.28083989501        'to the UTM
coordinate of the GPS to adjust
      ysens2 = (newUTMy(i) + l2 * Sin(phi2)) / 3.28083989501        'for the sensor
location
      xsens2 = (newUTMx(i) + l2 * Cos(phi2)) / 3.28083989501
      ysens3 = (newUTMy(i) + l3 * Sin(phi3)) / 3.28083989501
      xsens3 = (newUTMx(i) + l3 * Cos(phi3)) / 3.28083989501
      ysens4 = (newUTMy(i) + l4 * Sin(phi4)) / 3.28083989501
      xsens4 = (newUTMx(i) + l4 * Cos(phi4)) / 3.28083989501



      ys1(i) = ysens1
      xs1(i) = xsens1
      ys2(i) = ysens2
      xs2(i) = xsens2
      ys3(i) = ysens3
      xs3(i) = xsens3
      ys4(i) = ysens4
      xs4(i) = xsens4



    Loop


  f = 52
  Do Until f = b - 52
```

```
        f = f + 1
        Print #2, ys1(f); xs1(f); NDVI1(f); 1
        Print #2, ys2(f); xs2(f); NDVI2(f); 2
        Print #2, ys3(f); xs3(f); NDVI3(f); 3
        Print #2, ys4(f); xs4(f); NDVI4(f); 4   'write UTMy, UTMx, NDVI values, and
Sensor Number to a file
      Loop


CleanUp:
      Form1.MousePointer = 0   'reset mouse
      Close #1              'close file
      Close #2
      Close #10

   End If
   End If


   Exit Sub

End Sub
```

Below this line is the module for declaring public variables for 4100 for grass 2

Module

```
Public Temp As Double
Public Slope1 As Double
Public Slope2 As Double
Public Slope3 As Double
Public Slope4 As Double
Public Inter1 As Double
Public Inter2 As Double
Public Inter3 As Double
Public Inter4 As Double
Public T1Slope As Double
Public T2Slope As Double
Public T3Slope As Double
Public T4Slope As Double
```

```
Public T1Inter As Double
Public T1Coeff3 As Double
Public T1Coeff2 As Double
Public T1Coeff1 As Double
Public T2Inter As Double
Public T2Coeff3 As Double
Public T2Coeff2 As Double
Public T2Coeff1 As Double
Public T3Inter As Double
Public T3Coeff3 As Double
Public T3Coeff2 As Double
Public T3Coeff1 As Double
Public T4Inter As Double
Public T4Coeff3 As Double
Public T4Coeff2 As Double
Public T4Coeff1 As Double
Public length As Double
Public slopex As Double
Public intx As Double
Public slopey As Double
Public inty As Double
```

4100 for grass 3.exe

"4100 for grass 3.exe" formats the data into a comma delimited text file that can be imported to SSToolbox.

```
Private Sub mnuItemExit_Click()
   End                    'quit program
End Sub

Private Sub mnuItemOpen_Click() 'specify input file

   Wrap$ = Chr$(13) + Chr$(10) 'create wrap character
   CommonDialog1.Filter = "Data files (*.ot2)|*.ot2"
   CommonDialog1.ShowOpen       'display Open dialog box
   If CommonDialog1.FileName <> "" Then
      Form1.MousePointer = 11 'display hourglass
      Open CommonDialog1.FileName For Input As #1


   CommonDialog1.Filter = "Output files (*.csv)|*.csv"
```

```
    CommonDialog1.DialogTitle = "Specify Output File"
    CommonDialog1.ShowOpen     'specify output file

    If CommonDialog1.FileName <> "" Then
       Form1.MousePointer = 11
       Open CommonDialog1.FileName For Output As #2

       Dim strval As String

       Do Until EOF(1)        'then read lines from file and assign the correct values to Lat.
Lon, and Voltages 1-4

          Line Input #1, lineoftext$
          a = a + 1

          If Mid(lineoftext, 1, 1) = "L" Then
             Col1 = "Lat"
             Col2 = "Lon"
             Col3 = "NDVI"
             Col4 = "Sensor#"
             GoTo 10
          End If

          b = 1
          b = b + 1
          Do Until Mid(lineoftext, b, 1) = Chr(9) Or Mid(lineoftext, b, 1) = " "
             strval = strval & Mid(lineoftext, b, 1)
             b = b + 1
          Loop

          Col1 = Val(strval)
          strval = ""

          b = b + 2
          Do Until Mid(lineoftext, b, 1) = Chr(9) Or Mid(lineoftext, b, 1) = " "
             strval = strval & Mid(lineoftext, b, 1)
             b = b + 1
          Loop

          Col2 = Val(strval)
          strval = ""

          b = b + 1
          Col3 = Val(Mid(lineoftext, b, 7))
          b = b + 8
```

```
        Col4 = Val(Mid(lineoftext, b, 7))
        b = b + 8

10        Write #2, Col1, Col2, Col3, Col4

      Loop


CleanUp:
     Form1.MousePointer = 0   'reset mouse
     Close #1                 'close file
     Close #2


   End If
   End If


   Exit Sub

End Sub
```

matrixconv.bas

```
Sub Macro1()
'
' Macro1 Macro
' Macro recorded 2/16/2000 by Travis Tsunemori
'

'
   namesht = ActiveSheet.Name
   Range("c2").Select
   y = ActiveCell.Value
   Do Until junk = 100
      If ActiveCell.Value = y Then
         ActiveCell.Offset(1, 0).Select
         counters = counters + 1
      Else
         ActiveCell.Offset(-counters, -1).Select
         Range(ActiveCell, ActiveCell.Offset(counters - 1)).Select
         Selection.Copy
         Sheets.Add
         namesht2 = ActiveSheet.Name
```

```
        Range("b1").Select
        Selection.PasteSpecial Paste:=xlAll, Operation:=xlNone, SkipBlanks:=False,
Transpose:=True
        ActiveCell.Offset(1, -1).Select
        ActiveCell.Value = y
        Sheets(namesht).Select
        Range(ActiveCell.Offset(0, 2), ActiveCell.Offset(counters - 1, 2)).Select
        Selection.Copy
        Sheets(namesht2).Select
        ActiveCell.Offset(0, 1).Select
        Selection.PasteSpecial Paste:=xlAll, Operation:=xlNone, SkipBlanks:=False,
Transpose:=True
        junk = 100
        Sheets(namesht).Select
      End If
    Loop
    ActiveCell.Offset(counters, -1).Select
    y = ActiveCell.Value
    Do Until ActiveCell.Value = ""
      If ActiveCell.Value = y Then
        counts = counts + 1
        ActiveCell.Offset(1, 0).Select
      Else
        Sheets(namesht2).Select
        ActiveCell.Offset(1, -1).Select
        ActiveCell.Value = y
        Sheets(namesht).Select
        y = ActiveCell.Value
        Range(ActiveCell.Offset(-counters, 1), ActiveCell.Offset(-1, 1)).Select
        Selection.Copy
        Sheets(namesht2).Select
        ActiveCell.Offset(0, 1).Select
        Selection.PasteSpecial Paste:=xlAll, Operation:=xlNone, SkipBlanks:=False,
Transpose:=True
        Sheets(namesht).Select
        ActiveCell.Offset(counters, -1).Select
      End If
    Loop
End Sub
```

semivariance.bas

```
Sub Array_Semivariance()
' This Macro calculates semivariorgrams for array data in an Excel Spread Sheet
```

```
' To run this Macro Semivariogram, activate (click on) the top left cell.
' Do Not Click on (activate) the title of the column, if a title is included.
' There should not be more than 1000 data points.  The two columns to the immediate
' right of the data must be empty to receive the distance and semivariance data.
' Use CNTRL-z for quick key activating the Macro Semivariance_2.  This macro skips
cell containing
' "." .

Dim Origdata(1 To 1000, 1 To 1000)

xcells = Val(InputBox("Enter number of sample cells in the X direction."))
ycells = Val(InputBox("Enter number of sample cells in the Y direction."))
xoffset = ActiveCell.Row
yoffset = ActiveCell.Column

'xcells = 160
'ycells = 155
del = 100 / xcells
Semiincr = ycells - 1 - del 'number of increments in semivariogram
outrow = 1

'Read in sample array
cnt = 0

 For i = 1 To ycells
  For j = 1 To xcells
    Origdata(i, j) = Cells(i + xoffset - 1, j + yoffset - 1)
  Next j
 Next i

'  dist = Val(InputBox("Enter transect distance."))
  dist = ycells
  Rownum = ycells
  colnum = xcells



  delta = dist / Rownum
  Cycle = 0
  calc = 0
  dif = 0
  sumdif = 0
  sumdifsq = 0

For j = 1 To Semiincr
  Cycle = 1 + Cycle
```

```
   calc = Int(Rownum - Cycle)
   Count = 0
   For i = 1 To calc
    For k = 1 To xcells
      If Origdata(i + Cycle, k) = "." Or Origdata(i, k) = "." Then GoTo 1
      dif = Origdata(i + Cycle, k) - Origdata(i, k)
      sumdif = dif + sumdif
      sumdifsq = dif * dif + sumdifsq
      Count = Count + 1
1    Next k
   Next i

   Semivar = ((sumdifsq - (sumdif * sumdif) / Count) / (Count - 1)) / 2
   sepdist = j * delta
   Cells(j + outrow, xcells + 3) = sepdist
   Cells(j + outrow, xcells + 4) = Semivar
   sumdif = 0
   sumdifsq = 0

Next j

End Sub
```

VITA

Travis S. Tsunemori

Candidate for the Degree of

Master of Science

Thesis: MAPPING AND DETERMINATION OF SPATIAL VARIABILITY OF WINTER WHEAT AND TURFGRASS USING HIGH-RESOLUTION SENSOR DATA

Major Field: Biosystems Engineering

Biographical:

Personal Data: Born in Scottsbluff, Nebraska, December 11, 1973, the son of John and Peggy Tsunemori.

Education: Graduated from Mitchell High School, Mitchell, Nebraska, June 1992; Bachelor of Science in Biosystems Engineering, from Oklahoma State University, July, 1997. Completed the requirements for the Master of Science degree with a major in Biosystems Engineering at Oklahoma State University in December, 2000.

Experience: Raised on farms near Meriden, Wyoming and Mitchell, Nebraska; employed as a farm laborer during childhood years; employed by Lashley Auto Sales during summer of 1992; employed by Scenic Knolls Golf Course during summers of 1993 and 1994; employed by Wal-Mart during summer of 1995; employed by Oklahoma State University Biosystems and Agricultural Engineering from fall of 1995 through spring of 1997 as a work-study and part-time undergraduate research technician; currently employed by Oklahoma State University Biosystems and Agricultural Engineering since summer of 1997 as a Research Engineer.

Professional Memberships: American Society of Agricultural Engineers, Alpha Epsilon, the National Honor Society of Biological and Agricultural Engineers.